The Role of Logic in AGI Systems:
Towards a Lingua Franca for General Intelligence

Helmar Gust and **Ulf Krumnack** and **Angela Schwering** and **Kai-Uwe Kühnberger**

Institute of Cognitive Science, University of Osnabrück, Germany

{hgust|krumnack|aschweri|kkuehnbe}@uos.de

Abstract

Systems for general intelligence require a significant potential to model a variety of different cognitive abilities. It is often claimed that logic-based systems – although rather successful for modeling specialized tasks – lack the ability to be useful as a universal modeling framework due to the fact that particular logics can often be used only for special purposes (and do not cover the whole breadth of reasoning abilities) and show significant weaknesses for tasks like learning, pattern matching, or controlling behavior. This paper argues against this thesis by exemplifying that logic-based frameworks can be used to integrate different reasoning types and can function as a coding scheme for the integration of symbolic and subsymbolic approaches. In particular, AGI systems can be based on logic frameworks.

Introduction

As a matter of fact artificial intelligence currently departs significantly from its origins. The first decades of AI can be characterized as the attempt to use mostly logic-based methods for the development of frameworks and implementations of higher cognitive abilities. A few prominent examples are knowledge representation formalisms like Minsky's frames (Minsky, 1975), semantic networks (Sowa, 1987), McCarthy's situation calculus (McCarthy, 1963), Brachnan's KL-ONE (Brachman and Schmolze, 1985). With the rise and partial success of neurosciences, neuroinformatics, dynamic system theory, and other nature-inspired computing paradigms, logic-based frameworks seem to lose their importance in artificial intelligence. The current success of these new methodologies are at least partially based on the fact that several non-trivial problems are connected with logic-based approaches. Examples of such problems are the profusion of knowledge, the variety of reasoning formalisms, the problem of noisy data, and the lack of cognitive plausibility with respect to learning from sparse data, or reasoning in non-tractable domains.

A number of new research paradigms were proposed in order to overcome some limitations of logical computational frameworks. Many of them are inspired by biological or psychological findings. The following list summarizes some of these new methodologies:

- *Natural Computation:* This cluster contains methods like learning with neural networks and support vector machines, evolutionary computing, genetic algorithms, dynamic system theory etc.

- *Cognitive Robotics:* Embedded and embodied models of agents focus on perception aspects, motor control, and the idea that the world itself is the best of all possible models. This tradition moves the interest from higher to lower cognitive abilities.

- *Cognitive Architectures:* Integrated frameworks for modeling cognitive abilities, often use many different methodologies in order to model cognitive abilities. There seems to be a tendency for a method pluralism including also ideas from natural computation.

- *Further Approaches:* Further approaches for AI systems are, for example, fuzzy and probabilistic approaches (sometimes combined with logic), semantic networks that are enriched with activation potentials, the modeling of social aspects in multi-agent networks etc.

The mentioned methodologies are usually considered as alternatives to specialized logic-based AI systems. Although methods of, for example, natural computation may have strengths in particular domains, it is clearly far from being obvious that they can be used for modeling higher cognitive abilities. In particular, for AGI systems, which attempt to model the whole spectrum of higher and lower cognitive abilities, the difficulty to find a basis for its underlying methodology seems to be crucial: due to the fact that there are no good ideas how logical frameworks can be translated into such new computational paradigms, a starting point for AGI is the clarification of its methodological basis.

This paper argues for the usage of a non-standard logic-based framework in order to model different types of reasoning and learning in a uniform framework as well as the integration of symbolic and subsymbolic approaches. Analogical reasoning has the potential for an integration of a variety of reasoning formalisms and neural-symbolic integration is a promising research endeavor to allow the learning of logical knowledge with neural networks. The remainder of the paper has the following structure: first, we will discuss the variety of logic and learning formalisms. Second, we will show possible steps towards a logic-based theory that comprises different types of reasoning and learning. Third, we will argue for using logic as the lingua franca for AGI systems based on neural-symbolic integration.

Types of Reasoning	Corresponding Formalisms
Deductions	Classical Logic
Inductions	Inductive Logic Programming
Abductions	Extensions of Logic Programming
Analogical Reasoning	SME, LISA, AMBR, HDTP
Similarity-Based Reasoning	Case-Based Reasoning
Non-Monotonic Reasoning	Answer Set Programming
Frequency-Based Reasoning	Bayesian Reasoning
Vague and Uncertain Rasoning	Fuzzy, Probabilistic Logic
Reasoning in Ontologies	Description Logics
Etc.	Etc.

Table 1: Some types of reasoning and some formalisms

	Learning Types	Learning Approaches
Unsupervised Learning	Clustering	Neural Networks, SOMs, ART, RBF network
Supervised Learning	Classification, Learning a Function	Case-based reasoning, k-Nearest Neighbor, Decision Tree Learning
Reinforcement Learning	Policy Learning	Q-Learning, POMDPs, Temporal Difference Learning
Analytical & Inductive Learning	Rule Extraction, Learning in Domain Theories	Inductive Learing, Explanation-Based Learning, KBANNs

Table 2: Some types of learning and some methodological learning approaches

Tensions in Using Logic in AGI Systems

The Variety of Reasoning Types

Logical approaches have been successfully applied to applications like planning, theorem proving, knowledge representation, problem solving, and inductive learning, just to mention some of them. Nevertheless, it is remarkable that for particular domains and applications very special logical theories were proposed. These theories cannot be simply integrated into one uniform framework. Table 1 gives an overview of some important types of reasoning and their corresponding logic formalisms. Although such formalisms can be applied to a variety of different domains, it turns out that the degree of their generalization potential is limited. One will hardly find an approach that integrates inductive, deductive, abductive, fuzzy etc. logic theories in one unified framework.[1]

The Variety of Learning Types

A very similar situation can be found by shifting our attention from reasoning issues to learning aspects. The manifold of learning paradigms used in applications leads to the development of various corresponding formalisms. Major distinctions of learning approaches like supervised, unsupervised, and reinforcement learning structure the field, but are themselves just labels for a whole bunch of learning algorithms. Table 2 summarizes some important learning techniques commonly used in AI.

Although the list of machine learning algorithms guarantees that efficient learning strategies do exist for many applications, several non-trivial problems are connected with these theories. The following list summarizes some of them:

- Learning from noisy data is a classical problem for symbolic learning theories. Neural-inspired machine learning mechanisms have certain advantages in this respect.

- There is no learning paradigm that convincingly can learn from sparse, but highly conceptualized data. Whereas natural agents are able to generate inductive hypotheses based on a few data points due to the availability of background knowledge, classical machine learning mechanisms usually need rather large numbers of (more or less) unconceptualized examples.

- Learning in artificial systems is quite often realized explicitly in a separate module especially designed for a particular task. Contrary to this idea, cognitive learning is usually a continuous adaptation process taking place on many levels: besides explicit learning, rearranging representations in order to align structural properties of two inputs in a complex reasoning process, resolving clashes in a conceptualization by rewriting the representation, implicitly erasing features of an entity to make it compatible with background knowledge etc. may play an important role for adaptation processes (Kühnberger et al., 2008).

- The gap between symbolic and subsymbolic representations is a problem for AI systems because of complementary strengths and weaknesses. A framework that can extract the strengths of both fields would be desirable.

Steps Towards a Logic-Based Theory for Reasoning and Learning

Sketch of HDTP

A framework that proposes to integrate different reasoning types in one framework is heuristic-driven theory projection (HDTP) described in (Gust, Kühnberger, and Schmid, 2006). HDTP has been proposed for analogical reasoning. Due to the fact that establishing an analogical relation between a given source and target domain often requires the combination of different reasoning types, analogical reasoning is a natural starting point for an integration methodology for AGI systems.

We sketch the basic ideas of HDTP in this subsection.[2] HDTP established an analogical relation between a source theory Th_S and target theory Th_T (both described in a first-order language $\mathcal{L}$) by computing a generalization (structural description) Th_G of the given theories. This generalization process is based on the theory of anti-unification (Plotkin,

---

[1]Clearly, there are exceptions: an example may be Wang's NARS architecture (Wang, 2006) where the integration of the whole range of higher cognitive abilities are programmatically modeled in a logic-based framework.

[2]For a comprehensive presentation of the theory, the reader is referred to (Gust, Kühnberger, and Schmid, 2006) and (Schwering et al., 2009).

Solar System		Rutherford Atom	
$\alpha_1:$	$mass(sun) > mass(planet)$	$\beta_1:$	$mass(nucleus) > mass(electron)$
$\alpha_2:$	$\forall t : distance(sun, planet, t) > 0$	$\beta_2:$	$\forall t : distance(nucleus, electron, t) > 0$
$\alpha_3:$	$gravity(sun, planet) > 0$	$\beta_3:$	$coulomb(nucleus, electron) > 0$
$\alpha_4:$	$\forall x \forall y : gravity(x, y) > 0$	$\beta_4:$	$\forall x \forall y : coulomb(x, y) > 0$
	$\rightarrow attracts(x, y)$		$\rightarrow attracts(x, y)$
$\alpha_5:$	$\forall x \forall y \forall t : attracts(x, y) \wedge$		
	$distance(x, y, t) > 0 \wedge$		
	$mass(x) > mass(y)$		
	$\rightarrow revolves\_arround(y, x)$		

Generalized Theory	
$\gamma_1:$	$mass(A) > mass(B)$
$\gamma_2:$	$\forall t : distance(A, B, t) > 0$
$\gamma_3:$	$F(A, B) > 0$
$\gamma_4:$	$\forall x \forall y : F(x, y) > 0 \rightarrow attracts(x, y)$

Table 3: A formalization of the Rutherford analogy with obvious analogous structures (Krumnack et al., 2008)

1970). Anti-unification is the dual constructions of unification: if input terms t_1 and t_2 are given, the output is a generalized term t such that for substitutions Θ_1 and Θ_2 it holds: $t_1 = t\Theta_1$ and $t_2 = t\Theta_2$. It is well-known that for first-order anti-unification a generalization always exists, there are at most finitely many generalizations, and there exists a unique least generalization (Plotkin, 1970).

In order to apply the idea of anti-unification to analogies, it is necessary to extend the anti-unification framework in several respects. We will use the Rutherford analogy formalized in Table 3 as a simple example for motivating HDTP:

1. Not only terms but also formulas (theories) need to be anti-unified.

2. Whereas the generalization of α_1 and β_1 to γ_1 uses only first-order variables, the anti-unification of α_3 and β_3 requires the introduction of second-order variables (Table 3).

3. In order to productively generate the conclusion that electrons revolve around the nucleus, it is necessary to project α_5 to the source domain and generate a formula γ_5 in Th_G with

$$\exists A \exists B \forall t : attracts(A, B) \wedge distance(A, B, t) > 0 \wedge$$
$$mass(A) > mass(B) \rightarrow revolves\_arround(B, A)$$

In the following, we mention some properties of the extensions mentioned in the above list. The generalization of term anti-unification to the anti-unification of formulas is rather straightforward and will be omitted here.[3] A crucial point is the fact that not only first-order but also second-order generalizations need to computed, corresponding to the introduction of second-order variables. If two terms $t_1 = f(a, b)$ and $t_2 = g(a, b)$ are given, a natural second-order generalization would be $F(a, b)$ (where F is a function variable) with substitutions $\Theta_1 = \{F \leftarrow f\}$ and $\Theta_2 = \{F \leftarrow g\}$. Then: $t_1 = f(a, b) = F(a, b)\Theta_1$ and $t_2 = g(a, b) = F(a, b)\Theta_2$. It is known that unrestricted second-order anti-unifications can lead to infinitely many anti-instances (Hasker, 1995). In (Krumnack et al.,

2007), it is shown that a restricted form of higher-order anti-unification resolves this computability problem and is appropriate for analogy making.

Definition 1 *Restricted higher-order anti-unification is based on the following set of basic substitutions ($\mathcal{V}_n$ denotes an infinite set of variables of arity $n \in \mathbb{N}$):*

- *A renaming $\rho^{F,F'}$ replaces a variable $F \in \mathcal{V}_n$ by another variable $F' \in \mathcal{V}_n$ of the same argument structure:*

$$F(t_1, \ldots, t_n) \xrightarrow{\rho^{F,F'}} F'(t_1, \ldots, t_n).$$

- *A fixation ϕ_c^V replaces a variable $F \in \mathcal{V}_n$ by a function symbol $f \in \mathcal{C}_n$ of the same argument structure:*

$$F(t_1, \ldots, t_n) \xrightarrow{\phi_f^F} f(t_1, \ldots, t_n).$$

- *An argument insertion $\iota_{V,i}^{F,F'}$ with $0 \leq i \leq n$, $F \in \mathcal{V}_n$, $G \in \mathcal{V}_k$ with $k \leq n - i$, and $F' \in \mathcal{V}_{n-k+1}$ is defined by*

$$F(t_1, \ldots, t_n) \xrightarrow{\iota_{V,i}^{F,F'}}$$
$$F'(t_1, \ldots, t_{i-1}, G(t_i, \ldots, t_{i+k-1}), t_{i+k}, \ldots, t_n).$$

- *A permutation $\pi_\alpha^{F,F'}$ with $F, F' \in \mathcal{V}_n$ and bijective $\alpha : \{1, \ldots, n\} \rightarrow \{1, \ldots, n\}$ rearranges the arguments of a term:*

$$F(t_1, \ldots, t_n) \xrightarrow{\pi_\alpha^{F,F'}} F'(t_{\alpha(1)}, \ldots, t_{\alpha(n)}).$$

As basic substitutions one can rename a second-variable, instantiate a variable, insert an argument, or permute arguments. Finite compositions of these basic substitutions are allowed. It can be shown that every first-order substitution can be coded with restricted higher-order substitution and that there are at most finitely many anti-instances in the higher-order case (Krumnack et al., 2007).

A last point concerns the transfer of knowledge from the source to the target domain. In the Rutherford example, this is the projection of α_5 to the target domain governed by the analogical relation computed so far. Such projections allow to creatively introduce new concepts on the target side. The importance of these transfers for modeling creativity and productivity cannot be overestimated.

---

[3]Cf. (Krumnack et al., 2008) for more information.

Integrating Different Types of Reasoning and Learning with HDTP

As should be already clear from the rough introduction to HDTP, analogy making involves several types of reasoning. Not only that the result of the process is the establishment of an analogical relation between Th_S and Th_T, but as a side-effect a generalization is computed that can be interpreted as a learning result: the generalized theory Th_G together with appropriate substitutions cover not only the input theories, but potentially also some further theories. With respect to the Rutherford analogy the result is a central body system. Therefore, already at the level of the core theory, a form of inductive inference is computed.[4]

Another type of inference is involved in cases where the domain theories Th_S and Th_T are not in a form such that an association of corresponding terms and formulas is possible, although the theories can be analogically associated with each other. Assume that β_2 in Table 3 is replaced by

$$\beta_2' : \forall t : distance(electron, nucleus, t) > 0.$$

A simple association of α_2 and β_2' is no longer possible, because the argument positions do not coincide. In order to resolve this problem, HDTP uses a theorem prover to rewrite the axioms and deduce a representation that is more appropriate for analogy making. In our example, such a piece of background knowledge may be the formula

$$\forall x \forall y \forall t : distance(x, y, t) = distance(y, x, t)$$

Because the system is based on a first-order logical input, and for every first-order theory there are infinitely many possible axiomatizations, such cases may occur often. In (Krumnack et al., 2008), an algorithm is presented that shows how such a re-representation of a theory can be implemented into the HDTP core framework.

The remarks so far make clear that HDTP already covers different types of reasoning:

- Analogical inferences are used to establish the analogical relation between source and target.

- Anti-unification computes generalizations that can be interpreted as general hypotheses about the structure of the instance theories.

- Deductive inferences are used to rewrite the input for applicability of anti-unification.

- Vague reasoning is implicitly covered by the approach, because analogies are intrinsically fuzzy. There are no right or wrong analogies, rather there are psychologically preferred and psychologically less preferred analogies. HDTP models this aspect by using heuristics.

In alternative analogy models other types of reasoning were proposed for the integration into the analogy making process. As an example abductive reasoning is mentioned that can be modeled in the structure mapping engine (Falkenhainer, Forbus, and Gentner, 1989).

---

[4]The generalization has similarities to an induction step. Nevertheless, it is important to notice that anti-unification is governed by two structured theories and not by many examples like in the case of classical inductive reasoning.
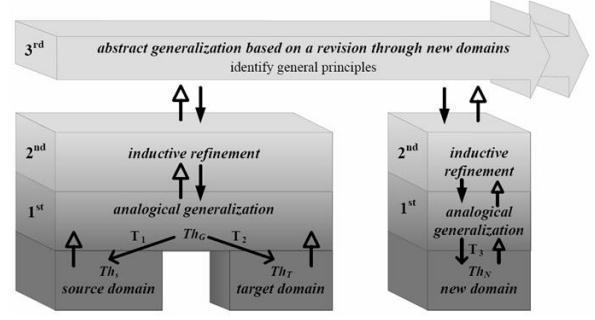

Figure 1: A graphical representation of the analogy making process involving different forms of reasoning and learning

Remarks on Learning with HDTP

As mentioned above not only the variety of different types of reasoning, but also the variety of different types of learning jeopardize the integration of learning into an AGI methodology, as well as learning from sparse and noisy data. The analogy making process presented here gives us some hints for possible solutions of some of these problems. First, notice that analogies do not require many data points. In fact, the input is given by just two theories (in the regular case). This suffices to learn a new conceptualization of the target and to find a generalization of both, source and target. Second, in (Kühnberger et al., 2008) the authors argue that learning from inconsistencies can be seen as a basic principle for adaptation processes in the cognitive architecture I-Cog: clashes occurring in reasoning, learning, or representation processes can trigger adaptation procedures that resolve such clashes. HDTP can be interpreted as one example where such adaptation processes can be specified on the algorithmic level of establishing analogies.

We explain some further issues in the analogy making process with respect to learning using Figure 1.

1. Level: The source and the target domain are compared via analogy to identify common structures between the domains. The commonalities are generalized to a theory Th_G. The analogy making process can transfer knowledge from the source to the target yielding a new conceptualization of the target. This results in a learning effect on the target side.

2. Level: The formulas projected from the source to the target need to be tested, because the analogical knowledge transfer might be true only for prototypical situations. The process runs through a number of different refinement steps and yields a parameter setting specifying in which range an analogy holds.

3. Level: The aim of this level is the identification of general principles. This type of learning requires the comparison (typically analogical comparison) of many different domain theories. At this level, the learning process starts with an intuitive hypothesis about a general principle and compares this iteratively with other domains to gain more confidence.

	Symbolic Approaches	Subsymbolic Approaches
Methods	(Mostly) logical and/or algebraic	(Mostly) analytic
Strengths	Productivity, Recursion Principle, Compositionality	Robustness, Learning, Parsimony, Adaptivity
Weaknesses	Consistency Constraints, Lower Cognitive Abilities	Opaqueness Higher Cognitive Abilities
Applications	Reasoning, Problem Solving, Planning etc.	Learning, Motor Control, Vision etc.
CogSci Relation	Not Biologically Inspired	Biologically Inspired
Other Features	Crisp	Fuzzy

Table 4: Differences between symbolic and subsymbolic theories

The three levels proposed above outline the different mechanisms that occur. Analogical learning in this sense is therefore a process in which different learning mechanisms interact and are iteratively repeated to refine and correct knowledge on the various abstraction levels.[5]

Logic as Lingua Franca for AGI Systems

In this section, we sketch ideas of how logic as the underlying representation formalism can be used for integrating subsymbolic devices to a logical reasoning system.

Building Neural-Symbolic Learning Systems

There is an obvious gap between symbolic and subsymbolic theories. As a matter of fact there is not only a methodological difference between these approaches, but furthermore strengths and weaknesses of the two paradigms are complementary distributed. Table 4 mentions some important distinctions between these two types of modeling options. In particular, for an AGI system intended to cover a large part of the breadth of cognitive abilities, a natural idea would be to bring these two research traditions together.

There has been the research endeavor "neural-symbolic integration" attempting to resolve this tension between symbolic and subsymbolic approaches.[6] The idea is to transform a highly structured input (e.g. a logical theory) into a flat input appropriate for connectionist learning. It has been shown that learning theories of propositional logic with neural networks can be achieved by using different frameworks like the "core method" or KBANNs (Hölldobler and Kalinke, 1994; Towell and Shavlik, 1994). In recent years, extensions to predicate logic were proposed (Bader, Hitzler, and Hölldobler, 2008).

We sketch some ideas of learning predicate logic with neural networks based on topos theory (Gust, Kühnberger, and Geibel, 2007). First, we introduce some concepts. A topos $\mathcal{T}$ is a category where all finite diagrams have limits (and colimits), any two objects have an exponential object,

and there exists a subobject classifier (Goldblatt, 1984). As a consequence, $\mathcal{T}$ has an initial and terminal object, as well as finite products and coproducts. The prototypical example for a topos is the category $\mathcal{SET}$. The objects of $\mathcal{SET}$ are sets, connected by set theoretic functions (called arrows). A product $a \times b$ can be identified with the well-known Cartesian product of two sets a and b, and an exponent a^b with the set of functions $f : b \to a$. The terminal object ! is the one-element set $\{0\}$ with the property that for all sets a there is exactly one arrow from a into $\{0\}$. The truth value object $\Omega = \{0, 1\}$ and the subobject classifier *true*: ! $\to \Omega$ mapping 0 to 1 generalizes characteristic functions and therefore interpretations of predicates.

We summarize how a topos can be used for neural learning of a logical theory T given in a first–order language $\mathcal{L}$.

- T is translated into a variable-free representation in a topos $\mathcal{T}$. The result is a representation of logic expressions by commuting diagrams, where objects are sets and arrows are set theoretic functions (in case we work in $\mathcal{SET}$). Logical terms can be interpreted as mappings from the terminal object into the universe U and logical 2-ary connectives as mappings $\Omega \times \Omega \to \Omega$. Quantified formulas correspond to an operation mapping (complex) predicates to (complex) predicates.

- An algorithm is generating equations of the form $f \circ g = h$ and inequations of the form $f \neq g$ corresponding to equations and inequations of arrows in the $\mathcal{T}$, i.e. set theoretic functions in $\mathcal{SET}$.

- As representations for objects and arrows it is possible to choose vectors of the vector space $\mathbb{R}^n$. The resulting equations and inequations can be used in order to train a feedforward neural network by backpropagation.

- The network learns a representation of objects and arrows of the underlying topos (which themselves represent symbols and expressions of $\mathcal{L}$), such that the truth conditions of the underlying axioms of T are satisfied. In other words, the network learns a model of T.

Although the approach still has many problems to solve, the very possibility to code axioms of a logical theory T, such that a neural network can learn a model of T, can be considered as the missing link between symbolic and subsymbolic representations. Applied to AGI systems, this means that logic can play the role of a lingua franca for general intelligence, without neglecting one of the two separated worlds of symbolic and subsymbolic computations. Rather it is possible to integrate both worlds into one architecture. A proposal for such an architecture (I-Cog) integrating both devices – the analogy engine and the neural-symbolic integration device – can be found in (Kühnberger et al., 2008).

Related Work

Analogies have been playing an important role in cognitive science and cognitively inspired AI for a long time. Classical frameworks are, for example, the symbolic approach SME (Falkenhainer, Forbus, and Gentner, 1989), the connectionist system LISA (Hummel and Holyoak, 1997), or the hybrid approach AMBR (Kokinov and Petrov, 2001). A

---

[5]This distinguishes HDTP from other learning approaches based on analogy. For example, the SME tradition (Falkenhainer, Forbus, and Gentner, 1989) focuses on alignment and transfer, whereas abstraction and interactions of different mechanisms do not play a crucial role.

[6]A good overview of neural-symbolic integration can be found in (Hammer and Hitzler, 2007).

good overview of analogy models can be found in (Gentner, Holyoak, and Kokinov, 2001). There are numerous classical papers on neural-symbolic integration, e.g. (Barnden, 1989) as one of the early ones. More recent work is concerned with extensions of propositional logic (D'Avila Garcez, Broda, and Gabbay, 2002), and the modeling of predicate logic using the "core method" (Bader, Hitzler, and Hölldobler, 2008).

Conclusions

This paper argues for the usage of logic as the basis for an AGI system without neglecting other nature-inspired approaches for modeling intelligent behavior. Although there is a zoo of logical formalisms that are sometimes hard to integrate with each other, we claim that directions towards such an integration of a variety of different reasoning types already exist. As an example we proposed the analogy engine HDTP in order to integrate such diverse types of reasoning like analogical, deductive, inductive, and vague reasoning. Furthermore, non-trivial learning mechanisms can be detected by adapting and fine-tuning the analogical relation. Finally, this paper claims that even the interaction between symbolic theories and subsymbolic theories can be achieved by the usage of techniques developed in neural-symbolic integration.

Obviously, many issues remain open. Besides the challenge of an empirical justification of the presented framework, other issues for future work need to be addressed. Currently it is impossible to solve challenging problems (e.g. benchmark problems) for theorem provers with neural-symbolic integration. Similarly, many questions of analogy making remain open and are not solved yet. One example is the integration of large knowledge bases, in particular, the retrieval of relevant knowledge, another one is the scalability of analogy making to applications in the large. Nevertheless, taking these ideas together it turns out that a variety of different cognitive abilities can be addressed in a uniform framework using logic as a mediating tool.

References

Bader, S., Hitzler, P., and Hölldobler, S. 2008. Connectionist model generation: A first-order approach. *Neurocomputing*, 71:2420–2432.

Barnden, J.A. 1989. Neural net implementation of complex symbol processing in a mental model approach to syllogistic reasoning. In Proceedings of the International Joint Conference on Artificial Intelligence, 568-573.

Brachman, R. and Schmolze, J. 1985. An Overview of KL-ONE Knowledge Representation System, *Cognitive Science* 9(2):171–216.

D'Avila Garcez, A., Broda, K., and Gabbay, D. 2002. *Neural-Symbolic Learning Systems: Foundations and Applications*. Berlin Heidelberg, Springer.

Falkenhainer, B., Forbus, K., and Gentner, D. 1989. The structure-mapping engine: Algorithm and example, *Artificial Intelligence*, 41:1-63.

Gentner, D., Holyoak, K., and Kokinov, B. 2001. *The Analogical Mind. Perspectives from Cognitive Science*, Cambridge, MA: MIT Press.

Goldblatt, R. 1984. *Topoi, the categorial analysis of logic*. North-Holland, Amsterdam.

Gust, H., Kühnberger, K.-U., and Geibel, P. 2007. Learning Models of Predicate Logical Theories with Neural Networks Based on Topos Theory. In: P. Hitzler and B. Hammer (eds.): Perspectives of Neural-Symbolic Integration, SCI 77, Springer, pp. 233-264.

Gust, H., Kühnberger, K.-U., and Schmid, U. 2006. Metaphors and Heuristic-Driven Theory Projection. *Theoretical Computer Science*, 354:98-117.

Hammer, B. and Hitzler, P. (eds.) 2007. *Perspectives of Neural-Symbolic Integration*. Springer, Berlin.

Hasker, R. 1995. The replay of program derivations. PhD thesis, Champaign, IL, USA.

Hölldobler, S. and Kalinke, Y. 1994. Ein massiv paralleles Modell für die Logikprogrammierung. Proceedings of the Tenth Logic Programming Workshop, WLP 94:89-92.

Hummel, J. and Holyoak, K. 1997. Distributed representations of structure: A theory of analogical access and mapping, *Psychological Review* 104(3):427–466.

Kokinov, B. and Petrov, A. 2001. Integrating Memory and Reasoning in Analogy–Making: The AMBR Model, in D. Gentner, K. Holyoak, B. Kokinov (eds.): *The Analogical Mind*. Perspectives from Cognitive Science, Cambridge Mass. (2001).

Krumnack, U., Schwering, A., Kühnberger, K.-U., and Gust, H. 2007. Restricted Higher-Order Anti-Unification for Analogy Making. *20th Australian Joint Conference on Artificial Intelligence*, Springer, pp. 273-282.

Krumnack, U., Gust, H., Kühnberger, K.-U., and Schwering, A. 2008. Re-representation in a Logic-Based Model for Analogy Making. *21st Australian Joint Conference on Artificial Intelligence*.

Kühnberger, K.-U., Geibel, P., Gust, H., Krumnack, U., Ovchinnikova, E., Schwering, A., and Wandmacher, T. 2008. Inconsistencies in an Integrated Cognitive Architecture. In: P. Wang, B. Goertzel, and S. Franklin (eds.): *Artificial General Intelligence 2008. Proceedings of the First AGI Conference*, IOS Press, pp. 212-223.

McCarthy, J. 1963 Situations, actions, and causal laws. *Stanford Artificial Intelligence Project Memo 2*, Stanford University, 1963.

Minsky, M. 1975. A Framework for Representing Knowledge. In P. Winston (ed.): *The Psychology of Computer Vision*, New York, pp. 211–277.

Plotkin, G. 1970. A note on inductive generalization. *Machine Intelligence* 5:153-163.

Schwering, A., Krumnack, U., Kühnberger, K.-U. and Gust, H. 2009. Syntactic Principles of HDTP. To appear in *Cognitive Systems Research*.

Sowa, J. 1987. Semantic Networks. In S. Shapiro (ed.): *Encyclopedia of Artifical Intelligence*, John Wiley & Sons.

Towell, G. and Shavlik, J. 1994. Knowledge-Based Artificial Neural Networks. *Artificial Intelligence* 70(1-2):119-165.

Wang, P. 2006. *Rigid Flexibility: The Logic of Intelligence*, Springer, 2006.