Towards Data-and Knowledge-Driven Artificial Intelligence: A Survey on Neuro-Symbolic Computing

Wenguan Wang[1] and Yi Yang[2]

[1]Australian Artificial Intelligence Institute, University of Technology Sydney, Australia.
[2]College of Computer Science and Technology, Zhejiang University, China.

Abstract

Neural-symbolic computing (NeSy), which pursues the integration of the symbolic and statistical paradigms of cognition, has been an active research area of Artificial Intelligence (AI) for many years. As NeSy shows promise of reconciling the advantages of reasoning and interpretability of symbolic representation and robust learning in neural networks, it may serve as a catalyst for the next generation of AI. In the present paper, we provide a systematic overview of the important and recent developments of research on NeSy AI. Firstly, we introduce study history and background concepts of this area. Afterward, we categorize recent approaches along several main characteristics that underline this research paradigm, including neural-symbolic interrelation, neural architecture, knowledge representation, and functionality. Then, we briefly discuss the successful application of modern NeSy approaches in several domains. Finally, we identify the open problems together with potential future research directions.

Keywords: Neuro-Symbolic AI, Symbolic AI, Statistical AI, Deep Learning

1 Introduction

Current advances in Artificial Intelligence (AI) have caused significant changes in numerous research fields, and had profound impacts on every nook and cranny of societal and industrial sectors. At the same time, there has been increasing concern in the public and scientific communities regarding trustworthiness, safety, interpretability, and accountability of the modern AI techniques [1]. This raises a natural question: ***What could be the key enable to the next generation of AI?***

Historically, there have been two dominant paradigms of AI, namely symbolism and connectionism. *Symbolism* conjectures that symbols representing things in the world are the fundamental units of human intelligence, and that the cognitive process can be accomplished by the manipulation of the symbols, through a series of rules and logic operations upon the symbolic representations [2, 3]. Many early AI systems, from the middle 1950s to the late 1980s, were built upon symbolistic models. Symbolic methods have several virtues: they require only a few input samples, use powerful declarative languages for knowledge representation, and have conceptually straightforward internal functionality. It soon became apparent, however, that such a *rule-based*, *top-down* strategy demands substantial hand-tuning and lacks true learning. As *discrete* symbolic representations and hand-crafted rules are intolerant of ambiguous and noisy data, symbolic approaches typically fall short when solving real-world problems.

Connectionism, known by its most successful technique, deep neural networks (DNNs) [4],

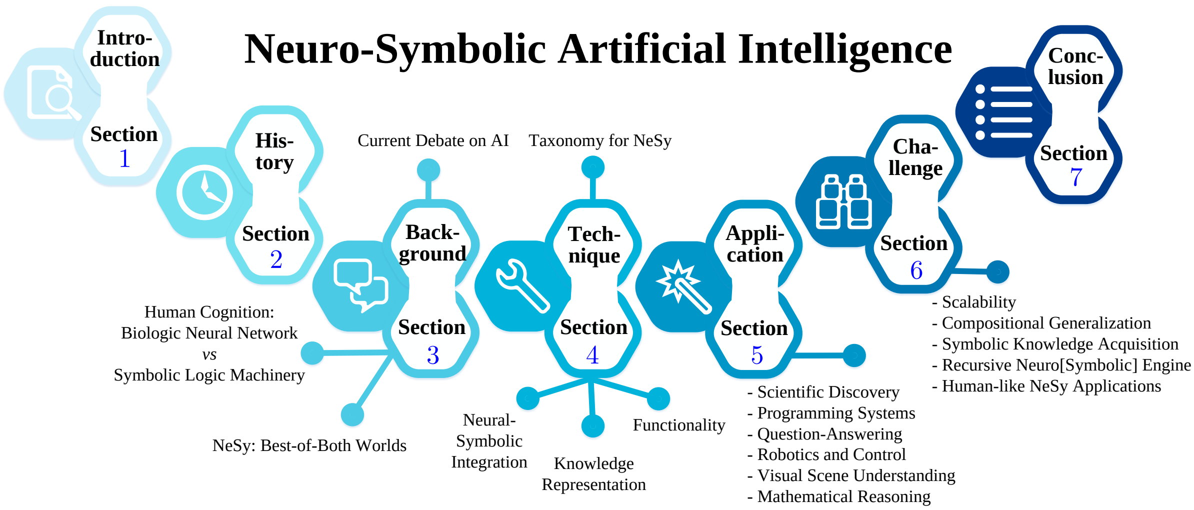

Fig. 1 Structure of the overall review.

serves as the architecture behind the vast majority of recent successful AI systems. Inspired by the physiology of the nervous system, connectionism explains cognition by interconnected networks of simple and often uniform units. Learning happens as weight modification, in a data-driven manner; the network weights are adjusted in the direction that minimises the cumulative error from all the training samples, using techniques such as gradient back-propagation [5]. Connectionist models are fault-tolerant, as they learn sub-symbolics, *i.e.*, *continuous* embedding vectors, and compare these vectorized representations instead of the literal meaning between entities and relations by discrete symbolic representations. Moreover, by learning *statistical* patterns from data, connectionist models enjoy the advantages of inductive learning and generalization capabilities. Like every coin has two sides, such approaches also suffer from several fundamental problems [6, 7]. First, connectionist models fall significantly short of *compositional generalization*, the robust ability of human cognition to correctly solve any problem that is composed of familiar parts [8]. Second, such bottom-up approaches are known to be *data inefficient*. Third, connectionist models are logically opaque, lacking *comprehensibility*. It is almost impossible to understand why decisions are made. In the absence of any kind of identifiable or verifiable train of logic, people are left with systems that are making potentially catastrophic decisions that are difficult to understand, arduous to correct, and therefore hard to be trusted. These shortcomings hinder the adoption of connectionist systems in decision-critical applications and reasoning-heavy tasks, such as medical diagnosis, autonomous driving, and mathematical reasoning, and lead to the increasing concern about contemporary AI techniques.

Against this backdrop, neural-symbolic computing (NeSy) [9], as a hybrid of symbolism and connectionism, is widely recognized as an enabler of the next generation of AI [10, 11]. NeSy essentially looks for the integration of two fundamental cognitive abilities [12, 13]: learning (the ability to learn from experience), and reasoning (the ability to reason from what has been learned), so as to exploit the major strengths and circumvent the inherent deficiencies of the two paradigms. However, building such an integrated machinery is challenging – one has to conciliate the methodologies of distinct areas [14], for example, statistical inductive learning based on distributed representations *vs* logical deductive reasoning based on localist representations. Though challenging, NeSy has attracted soaring research attention in the recent past, and has shown its superiority in many application scenarios, including visual relationship understanding [15, 16], visual question answering [17–19], visual scene parsing [20, 21], and commonsense reasoning [22].

In order to facilitate readers to catch up on the rapidly-developing evolution of this field, this paper offers a systematical and timely collection of recent important literature on NeSy, with particular attention to the past five years. The surveyed papers are those works published in the flagship repositories for machine learning and related areas, such as computer vision, natural language processing (NLP), and knowledge graph, or have been widely cited. This survey is expected to offer an exhaustive and up-to-date literature overview to researchers of interest, and nourish the exploration of open and developmental issues. We also remark that this survey is undoubtedly a biased view, as the area of research is large, but we do attempt to cover the current major research threads. Readers are also encouraged to refer to discussions in [13, 23–25], among others, to gain a sense of the breadth of this area.

A summary of the structure of this article can be found in Figure 1, which is presented as follows: In Section 2, we begin with a brief review of early research results of NeSy, which shape the latest effort in this area. Section 3 then elaborates on the general concepts of mind in psychology and cognitive science, which underpin the theoretical foundations of NeSy; and discusses recent debate regarding the necessary and sufficient building blocks of AI, which promotes the advance of this area. In Section 4, we state our taxonomy of NeSy, based on the classification schemes respectively proposed by Sebastian and Hitzler at 2005 [26] and Henry Kautz at AAAI-2020 [27]. By means of our taxonomy, we survey recent important literature. In Section 5, we elaborate on popular and emerging application areas of NeSy. Finally, Sections 6 and 7 present potential valuable directions for further research and conclude the survey. We hope that this survey will help newcomers and practitioners to navigate in this massive field which gained significant momentum in the past few years, as well as provide AI community background in generating future research.

2 History

This section provides a historical perspective of NeSy, prior to its recent acceleration in activity. NeSy aims to provide a unifying view for symbolism and connectionism, advance the modelling of cognition and further behaviour, and build preferable computational methodologies for integrated machine learning and logical reasoning [14]. NeSy has a long-standing tradition, which can date back to McCulloch and Pitts 1943 paper [9], even before AI was recognized as a new scientific field. We would like to recommend the readers eager to obtain a particular overview of primitive works to consult previous review articles, *e.g.*, [14, 26, 28].

Although in the seminal work [9] McCulloch and Pitts have demonstrated the strong relation between finite automata (boolean logic) and artificial neural networks, by interpreting simple logical connectives such as conjunction, disjunction and negation as binary threshold units in neural networks [26], NeSy only began to be a formalized field of study since the 1990 and gained systematic research in the early 2000s [29]. For example, Towell *et al.* [30] compiled hand-written symbolic rules into a neural network, and the approximately correct knowledge can be further corrected by empirical learning. Based on some landmark efforts [31–33], researchers developed various neural systems for logical inference [28, 34] and knowledge representation [35–39]. As their neural architectures are typically meticulously designed for hard logic reasoning, they are short of learning representations and struggle at reasoning over large-scale, heterogeneous, and noisy data [29]. However, these early NeSy systems laid the foundations for today's research.

During the 2010s, NeSy received relatively less attention, as DNN-based connectionist techniques achieved remarkable success across a variety of AI tasks. Nevertheless, as the shortcomings of DNNs became apparent, NeSy has recently ushered in its renaissance in the research community.

3 Background and Context

This section serves to further clarify two major driving forces behind the field of NeSy: The first one is the theoretical desire of understanding and modeling human cognition (Section 3.1), while the second one is the practical value of combining connectionism and symbolism paradigms in AI application scenario (Section 3.2). Section 3.3 further summarizes the recent AI debate among influential thinkers, which motivates a broad range of AI researchers to recognize the value of NeSy.

3.1 Human Cognition: Biologic Neural Network *vs* Symbolic Logic Machinery

• **Symbols *vs* Neurons.** What is the essence of human cognition? Many researchers agreed that symbolic facility is what distinguishes humans from other animals. The prosperity of human sociology and technology is closely concerned with the co-evolution of human brain with symbolic thinking, making us the "symbolic species" [40–42]. Many cognitive scientists hold a view that human thinking depends upon symbol manipulation. From this point of view, human mind is undisputedly symbolic. Hence symbolism was conceived in the attempts to structurally code knowledge and logic reasoning into machines. However, human cognition has a physical basis in the brain, which is composed of numerous mostly homogenous neurons. The neurons, together with the connections, or synapses, as well as diverse firing patterns among them, support different cognitive processes, such as attention, problem-solving, memory, learning, decision-making, language, perception, imagination, and logic reasoning. So it seems reasonable to assume if we can simulate the anatomy and physiology of the nervous system with artificial neurons, intelligence will be developed in computers. This belief leads to the emergence of connectionism.

Spontaneously, in order to advance the understanding of the human mind, it appears to be reasonable to look for ways of integration of symbolic and connectionist approaches, instead of focusing on the dichotomy. In that context, artificial neural networks can be regarded as an abstraction of the physical workings of the brain, while the symbolic logic can be viewed as an abstraction of what we perceive, through introspection, when contemplating explicit cognitive reasoning [43]. Hence it is of necessity to ask how these two abstractions can be related or even unified, or how symbol manipulation can arise from a neural substrate [14, 44].

• **Deduction *vs* Induction.** Deductive reasoning and inductive learning arguably constitute two indispensable building blocks of human thinking, helping human to develop knowledge of the world (even though there are yet other building blocks, such as abductive reasoning) [45]. However, their tension might be the most fundamental issue in areas such as philosophy, cognition, and, of course, AI [23]. The deduction camp [46] is aware of the expressiveness of formal languages for representing knowledge about the world, together with proof systems for reasoning from such knowledge bases. The learning camp [47] attempts to generalize from examples about partial descriptions about the world [23]. Historically, the dichotomy between the two camps roughly divided the development of the field of AI. Symbolic techniques clearly stand on the side of deductive reasoning; symbolic logic emphasizes high-level reasoning, and sticks to structure the world in terms of objects, attributes, and relations [23].

By contrast, neural networks are in the statistical learning camp; they learn statistical patterns, *i.e.*, distributed representations of entities, from data. Nevertheless, we humans make extensive use of both deduction and induction in everyday life as well as scientific investigation. We cannot precisely determine which part of human cognition is essentially symbolic, and which part of it is essentially statistical. Consequently, it is imperative to rethink the relationships between deductive reasoning and inductive learning, necessitating robust computational models that are able to coordinate the symbolic essence of reasoning with the statistical nature of learning.

• **Compositionality *vs* Continuity.** Smolensky *et al.* [7] recently proposed to simultaneously exploit two scientific principles, which are able to explain the way the human brain works, for machine intelligence, from the viewpoint of the underlying computation mode of human cognition. Neurophysiological measurements suggest that information is encoded within the brain through the numerical activation levels of massive neurons, and is processed by spreading this activation through myriad synapses of varying strengths and degrees of permanence [6]. Hence it seems apparent that human cognition deploys neural computing [48], which is in accord with the *Continuity Principle*: "the encoding and processing of information are formalized with real numbers that vary continuously" [7]. However, modern scientific studies [49] in philosophy and cognition suggested that all aspects of human intelligence, from language and perception to reasoning and planning, rely on a different type of computing: compositional-structure processing [50]. This type of computing follows the *Compositionality*

Principle [51]: complex information is encoded in large structures which are systematically composed from smaller structures that encode simpler information. Compositionality has been widely acknowledged as a core of human intelligence [52]. Our knowledge representation is naturally compositional. For example, we understand the world as a sum of its parts: objects can be broken down into pieces, events are a sequence of actions, and sentences are a series of words. Human cognition demonstrates strong compositional generalization – the ability of reorganizing familiar knowledge components in novel ways to solve new problems, so as to handle the potentially infinite number of states of the world [6]. Historically, compositional-structure processing is formalized in the form of discrete symbolic computing, like using words to make sentences. Thus, to some extent, the nature of computation in our brains is simultaneously neural and compositional-structure. How can this be? Smolensky called this the *Central Paradox of Cognition* [53]. Resolving this paradox inevitably calls for a new computing mechanism, that addresses both the Continuity and Compositionality Principles simultaneously[1].

• **System 1 *vs* System 2.** Kahneman's '*fast and slow thinking*' theory, which explains the machinery of human thought, also motivated recent research interest in NeSy [54]. In [55], Kahneman pointed out that humans' decisions are supported by the cooperation of two different kinds of capabilities, called *system 1* ('fast thinking') and *system 2* ('slow thinking'). Specifically, system 1 thinking is a near-instantaneous and experience-driven process for intuitive, imprecise, quick, and largely unconscious decisions, corresponding to 98% of thinking. System 1 thinking, for example, can be in the form of knowing how to zip your jacket without a second thought. Differently, system 2 thinking is slower, deliberative, and conscious, often associated with the subjective experience of agency, choice, and concentration; it provides a powerful tool for solving more complicated problems, where logical, sequential, algorithmic thinking is needed. For example, system 2 thinking is used when working on math problems. It is also worth mentioning that compositional generalization is exhibited in both system 1 thinking and system 2 thinking [7]. Interestingly, system 2 can be viewed as a "slave" of system 1: when system 1 runs into difficulty, it is system 1 that decides to initiate system 2. Even during the execution of system 2, system 1 is ultimately in charge [27]. In addition, solutions discovered by system 2 can be readily available for later use by system 1. Thus, after a while, some problems, initially solvable only by resorting to system 2, can become manageable by system 1 [54]. The consistent and effective use of system 2 can calibrate system 1, which, in turn, promotes system 2, leading to a feedback loop. As the characteristics of system 1 and system 2 are strikingly similar to those of the connectionist approach to AI and the symbolic approach to AI, more and more AI researchers began to rethink the relation between the two traditions and recognize the value of NeSy.

3.2 NeSy: Best-of-Both Worlds

Rather than taking the motivation from the objective of achieving rational understanding and modeling of human cognition, the study of NeSy is also driven by a more technically motivated perspective – combining numerical connectionist and symbolic logic approaches in order to construct more powerful reasoning and learning machines for computer science applications.

The second motivation is based on the observation that connectionist techniques, especially modern DNNs, and symbolic approaches complement each other with respect to their advantages and disadvantages. In particular, connectionist techniques are good at discovering statistic patterns from raw data and are robust against noisy data. Hence they are effective in intuitive judgements, such as image classification. On the other hand, connectionist techniques are data hungry, and black boxes – it is especially challenging to understand their decision-making processes. Alternatively, symbolic approaches are excellent at principled judgements, such as logical reasoning; they exhibit inherently high explainability and provide the ease of using powerful declarative

[1] Note that the solution – *neurocompositional computing* – proposed by Smolensky *et al.* [7] is slightly different from NeSy. NeSy broadly refers to any possible hybrid systems that couple, loosely or tightly, neural and symbolic approaches. Neurocompositional computing, instead, is to directly realize compositional-structure processing through continuous neural computing, which can be viewed as a compact, neural network based NeSy system. However, in spite of such difference, both NeSy and neurocompositional computing share the same motivation.

languages for knowledge representation. Nevertheless, symbolic approaches are far less trainable and susceptible to out-of-domain brittleness. As a result, the integration of neural and symbolic approaches seems to be a natural step toward more powerful, trustworthy, and robust AI.

3.3 Current Debate on AI

Recent years have witnessed remarkable breakthroughs in the field of AI, brought by connectionist approaches and deep learning in particular. But researchers are also coming to realize that contemporary AI systems suffer from serious deficiencies in terms of, for example, data efficiency, comprehensibility, and compositional generalization [56]. This led to influential debates between famous researchers, which are about the underlying principles of AI. As a result, NeSy research gained renewed importance.

Specifically, the 2019 Montreal AI Debate between Yoshua Bengio and Gary Marcus [44], and the AAAI-2020 fireside conversation with Economics Nobel Laureate Daniel Kahneman and the 2018 Turing Award winners and deep learning pioneers Geoff Hinton, Yoshua Bengio, and Yann LeCun, brought new perspectives and concerns on the future of AI. In the debate between Yoshua Bengio and Gary Marcus, Marcus emphasizes the importance of hybrid systems: "$\cdots$ *in order to get to robust artificial intelligence, we need to develop a framework for building systems that can routinely acquire, represent, and manipulate abstract knowledge, with a focus on building systems that use that knowledge in the service of building, updating, and reasoning over complex, internal models of the external world.*" Though Hinton agreed that "*we need those higher-level concepts to be grounded and have a distributed representation to achieve generalization*", he also addressed that "*(numerical connectionist approaches) can get many of the attributes of symbols without the kind of explicit representations of them which has been the hallmark of classical AI*" and that "*The reason why connectionists really wanted to depart from symbolic processing is because they thought that is wasn't a sufficiently rich kind of representation.*" At AAAI-2020, Kahneman highlighted the importance of symbol manipulation in system 2: "$\cdots$ *as far as I'm concerned, system 1 certainly knows language* $\cdots$ *system 2 does involve certain manipulation of symbols.*" Although there are disagreements about, for example, how to represent symbols in DNNs and how to achieve the hybrid of connectionism and symbolism, the thinkers, in broad strokes, are in agreement that new-generation AI systems ought to be able to handle high-level abstract concepts and to conduct sound reasoning.

4 NeSy: Taxonomy and State of the Art

This section is devoted to the structured and comprehensive review of state-of-the-art NeSy algorithms. Section 4.1 details our taxonomy for NeSy, based on which we survey recent main research results in this area from three perspectives: neural-symbolic integration (Section 4.2), knowledge representation (Section 4.3), and functionality (Section 4.4).

4.1 Our Taxonomy for NeSy

Our overall taxonomy for NeSy AI is mainly built upon the classification scheme proposed by Sebastian and Hitzler in 2005 [26], but modified according to our particular focus and recent development tendency in this field. Basically, our scheme has three axes, namely *neural-symbolic integration*, *knowledge representation*, and *functionality*.

The first axis – neural-symbolic integration – categorizes NeSy systems according to the combination mode – how the symbolic and neural parts are integrated as a hybrid. Along this axis, we further adopt the classification schema recently introduced by Henry Kautz at AAAI-2020 [27], which is influential and insightful. More details of this axis will be given in Section 4.2.

For the second axis – knowledge representation, we focus on the symbolic aspect of the NeSy AI system. Depending on how the knowledge is represented, *i.e.*, symbolic *vs* logic, we can distinguish the systems, as discussed in Section 4.3.

The third axis refers to the functionality of the NeSy system, namely that it focuses more on machine learning or automated symbolic reasoning. More detailed discussions can be found in Section 4.4.

Note that the three axes are proposed to comprehensively describe the key characteristics of a NeSy system; they are not mutually exclusive.

Along with these three axes of our taxonomy, we summarize the key features of recent remarkable works in this field in Table 1 and give detailed review below.

4.2 Neural-Symbolic Integration

With a good understanding of the reasons behind the need for integrating symbolic and connectionist approaches, it should turn next to the integration mode. Following the rationale laid out in [27], we distinguish six types of NeSy AI systems:

• **Type 1. Symbolic Neuro Symbolic** (Figure 2): This is, in Kautz's words, *the current standard operating procedure* of deep learning methods in some application tasks where the input and output are symbols. For example, most current NLP systems fall under this category; the input symbols are converted to vector embeddings by word2vec [57], GloVe [58], *etc.*, and then processed by the neural models, whose output embeddings are further transferred to the required symbolic category or sequence of symbols via a softmax operation. This type, though some may argue is a stretch to refer to as NeSy, is included by Kautz to emphasize that the input and output of a neural network can be made of symbols [130], *e.g.*, in the case of language translation, or graph classification.

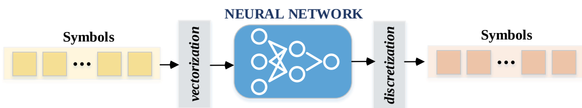

Fig. 2 Type 1: Symbolic Neuro Symbolic.

• **Type 2. Symbolic[Neuro]** (Figure 3): Type 2 refers to hybrid but overall symbolic systems, where neural modules are internally used as subroutine within a symbolic problem solver. In a nutshell, the symbolic and neural parts are only loosely-coupled. Kautz includes DeepMind's AlphaGo [59] as an example: the problem solver is the Monte-Carlo Tree Search algorithm and its heuristic evaluation function is a neural network. In [60], a NeSy model is designed to learn and generalize compositional rules. Based on a sequence-to-sequence generation network, a symbolic stack machine is adopted for supporting recursion and sequence manipulation, and the execution trace is produced by a neural network. In [61], a rule-based system, which uses abstract concepts captured by a neural perception module as I/O specifications, is introduced for program synthesis from raw visual observations.

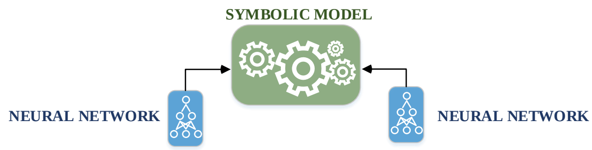

Fig. 3 Type 2: Symbolic[Neuro].

• **Type 3. Neuro|Symbolic** (Figure 4): Type 3 is also a hybrid system where the neural and symbolic parts focus on different but complementary tasks in a big pipeline. Kautz mentioned Type 3 and Type 2 "*differ in that the neuro part is a coroutine rather than a subroutine.*" To eliminate ambiguity, here we further restrict Type 3 as the hybrid systems where the interaction between neural and symbolic parts can boost both the individual and collective performance. Therefore, the relation between neural and symbolic parts in Type 3 systems is collaboration, rather than only functional dependency in Type 2. For instance, [76] presents an abductive learning framework, which conducts sub-symbolic perception learning and symbolic logic reasoning separately but interactively. Many deep learning based program synthesis algorithms [17, 62–67] that leverage deep learning techniques to generate symbolic programs/rule systems satisfying high-level task specifications also fall in this category. Another notable case is [18], where a neural perception module learns visual concepts and a symbolic reasoning module executes symbolic programs on the concept representations for question answering. The symbolic reasoning module provides feedback signals that support gradient-based optimization of the neural perception module. Recent efforts [68–75] in neural-symbolic reinforcement learning (RL) also belong to Type 3. For example, in [68], symbolic planning are integrated into reinforcement learning (RL) for robust decision-making. Symbolic plans are used to guide task execution, and the task experiences are fed back for improving symbolic planning. Some other examples include Neural Theorem Provers (NTPs) [77], Conditional Theorem Provers (CTPs) [78], NLProlog [79],

Table 1 Summary of essential characteristics for reviewed neuro-symbolic methods.

Neural-Symbolic Integration	Method	Knowledge Representation	Functionality		Knowledge Embedding
			Learning	Reasoning	
Symbolic Neuro Symbolic	word2vec [57]	Word Vocabulary	Loss Constraint	Probabilistic	Parameter
	Glove [58]	Word Vocabulary	Loss Constraint	Probabilistic	Parameter
Symbolic[Neuro]	AlphaGo [59]	-	Reinforcement	Probabilistic + Tree Search	Parameter
	NeSS [60]	Compositional Generalization	Loss Constraint	Probabilistic + Stack Machine	Parameter
	PLANS [61]	Programme Language	Reinforcement	Probabilistic + Inductive Reasoning	Parameter
Neuro|Symbolic	NS-VQA [17]	Programme Language	Program Synthesis	Probabilistic	Parameter
	NSPS [62]	Programme Language	Program Synthesis	Probabilistic + Tree Parser	Parameter
	NeRd [63]	Programme Language	Program Synthesis	Probabilistic + Compositional	Parameter
	Synth [64]	Programme Language	Program Synthesis	Probabilistic + Interpreter	Parameter
	PS-GM [65]	Programme Language	Program Synthesis	Probabilistic	Parameter
	HOUDINI [66]	Programme Language	Program Synthesis	Probabilistic + Function Composition	Parameter
	Neural LP [67]	First-order Logic	Program Synthesis	Probabilistic + Logic Chain Inference	Parameter
	NS-CL [18]	Programme Language	Reinforcement	Probabilistic	Parameter
	PEORL [68]	Programme Language	Reinforcement	Probabilistic + Planner	Parameter
	DSRL [69]	Logic Rule	Reinforcement	Probabilistic	Parameter
	CDSE [70]	Logic Rule	Reinforcement	Probabilistic + Symbolic Executor	Parameter
	NSCA [71]	Logic Rule	Reinforcement	Probabilistic	Parameter
	SDRL [72]	Programme Language	Reinforcement	Probabilistic + Planner	Parameter
	SORL [73]	Programme Language	Reinforcement	Probabilistic + Planner	Parameter
	NSM [74]	Programme Language	Reinforcement	Probabilistic + Interpreter	Parameter
	NLRL [75]	First-order Logic	Reinforcement	Probabilistic + Markov Decision Process	Parameter
	ABL [76]	First-order Logic	Logic Programming	Probabilistic + Abductive	Parameter
	NTPs [77]	First-order Logic	Logic Programming	Probabilistic + Backward-Chaining	Parameter
	CTPs [78]	First-order Logic	Logic Programming	Probabilistic + Backward-Chaining	Parameter
	NLProlog [79]	First-order Logic	Logic Programming	Probabilistic + Backward-Chaining	Parameter
	DeepProbLog [80]	Programme Language	Logic Programming	Probabilistic + Sentential Decision	Parameter
	NeuroLog [81]	Programme Language	Logic Programming	Probabilistic + Abductive	Parameter
	DiffLog [82]	Programme Language	Logic Programming	Probabilistic + Logic Chain Inference	Parameter
Neuro$_{\text{SYMBOLIC}}$	Alps [83]	First-order Logic	Logic Programming	Probabilistic	Parameter
	EQNet [84]	Compositional Generalization	Loss Constraint	Tree Parser + Probabilistic	Data
	SD-VAE [85]	Compositional Generalization	Loss Constraint	Tree Parser + Probabilistic	Data
	Tree2tree [86]	Compositional Generalization	Loss Constraint	Tree Parser + Probabilistic	Data
	JTAE [87]	Compositional Generalization	Loss Constraint	Tree Parser + Probabilistic	Data
	DLSM [88]	First-order Logic	Loss Constraint	Tree Parser + Probabilistic	Data
	NGS [89]	Compositional Generalization	Reinforcement	Tree Parser + Probabilistic	Data
	NeuralMath [90]	Compositional Generalization	Loss Constraint	Tree Parser + Probabilistic	Architecture
	IEP [91]	Programme Language	Reinforcement	Probabilistic + Compositional Reasoning	Parameter
	DrKIT [92]	First-order Logic	Logic Programming	Probabilistic	Parameter
	∂ILP [93]	Logic Rule	Logic Programming	Probabilistic + Backward-Chaining	Parameter
	NGCLP [94]	Constraint Graph	Logic Programming	Probabilistic + Neural Guided Search	Architecture
	NSM [95]	Scene Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	XNMs [96]	Scene Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	Prob-NMN [97]	Scene Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	LENSR [98]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Data
	CNIF [99]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	HHP [100]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	KagNet [101]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	GRHEK [102]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	GraIL [103]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	KB-GAN [104]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	KRISP [105]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	SMRKB [106]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	SGN [107]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	SGR [108]	Knowledge Graph	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	CSAE [109]	Propositional Logic	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
	SYMNET [110]	First-order Logic	Loss Constraint	Probabilistic + Graph Reasoning	Architecture
Neuro: Symbolic → Neuro	LTN [111]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	Logistic Circuits [112]	Logical Sentence	Loss Constraint	Probabilistic	Parameter
	LRI [113]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	HDNNLR [114]	Propositional Logic	Loss Constraint	Probabilistic	Parameter
	LFS [115]	Propositional Logic	Loss Constraint	Probabilistic	Parameter
	Semantic Loss [116]	Propositional Logic	Loss Constraint	Probabilistic	Parameter
	DANN [117]	Propositional Logic	Loss Constraint	Probabilistic	Parameter
	DFL [118]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	LFIE [119]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	ANNFL [120]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	MultiplexNet [121]	First-Order Logic	Loss Constraint	Probabilistic	Parameter
	SBR [122]	First-Order Logic	Loss Constraint	Probabilistic + Logic Constraint	Parameter
	LYRICS [123]	First-Order Logic	Loss Constraint	Probabilistic + Logic Constraint	Parameter
	DL2 [124]	First-Order Logic	Loss Constraint	Probabilistic + Logic Constraint	Parameter
	C-HMCNN(h) [125]	Logic Rule	Loss Constraint	Probabilistic + Logic Constraint	Parameter
Neuro[Symbolic]	NSLMs [126]	Knowledge Base	Loss Constraint	Probabilistic	Architecture
	NLM [127]	Propositional Logic	Loss Constraint	Probabilistic + Sequential Logic Deduction	Architecture
	SATNet [128]	-	Loss Constraint	Probabilistic + Solver	Architecture
	GVR [129]	-	Loss Constraint	Probabilistic + Recursion	Architecture

DeepProbLog [80], NeuroLog [81], DiffLog [82]. Among them, a notable case is DeepProbLog [80], which adopts neural networks as predicates to compute the probabilities of probabilistic facts, and hence uses the inference mechanism of ProbLog, a probabilistic logic programming language, to compute the gradient of the desired loss.

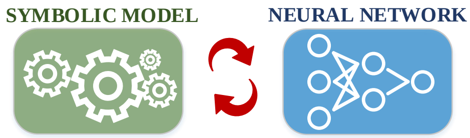

Fig. 4 Type 3: Neuro|Symbolic.

• **Type 4. Neuro: Symbolic → Neuro** (Figure 5): In a TYPE 4 NeSy system, symbolic rules/knowledge are compiled into the architecture or training regime of neural networks. For instance, there is a recent surge of interest in learning vector based representations of symbolic knowledge so as to naturally incorporate symbolic domain knowledge into connectionist architectures [83–87, 98]. Some neural-symbolic mathematics systems for equation solving [88, 89] and verification [90] represent mathematical expressions as trees, and create meaningful mathematical expressions as training data. A family of (visual) question answering models [19, 91–97] generate and execute symbolic programs for answering questions, where the programs are implemented as fully differentiable operations and/or neural networks. A huge body of recent algorithms [99–110] leverage graph neural networks (GNNs) to embed entities and relations in external knowledge bases, so as to boost the performance in various applications tasks in computer vision, and natural language processing. Broadly speaking, these methods fall into this type, while some may argue the reasoning ability of GNNs is rather weak.

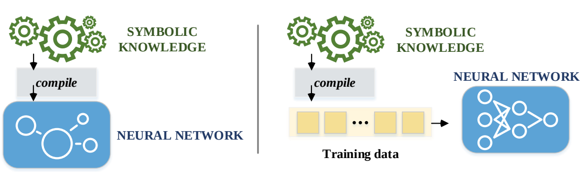

Fig. 5 Type 4: Neuro: Symbolic → Neuro.

• **Type 5. Neuro$_{\text{Symbolic}}$** (Figure 6): This type of NeSy systems turn symbolic knowledge into additional soft-constraints in the loss function used to train DNNs. Thus the knowledge is compiled into the weights of the neural network. Logic Tensor Networks (LTN) [111, 112], as a prominent example, interpret first-order logic as real functions or tensor operations, and translate the logic reasoning as a training objective. Some other recent efforts in this direction include [113–125]. In [21], compositional relations over semantic hierarchies are cast as extra training targets for hierarchical scene parsing.

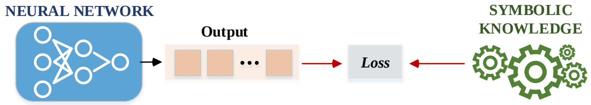

Fig. 6 Type 5: Neuro$_{\text{SYMBOLIC}}$.

• **Type 6. Neuro[Symbolic]** (Figure 7): Type 6 systems, which are believed by Kautz to "*has the greatest potential to combine the strengths of logic-based and neural-based AI*", are fully-integrated systems that directly embed a symbolic reasoning engine inside a neural engine. By imitating logical reasoning with tensor calculus, a line of approaches learn the execution of symbolic operations through neural networks [126–129], which, to some extent, can be classified into Type 6. However, their logic-reasoning ability is still relatively weak. Kautz views Type 6 methods as computational models of Kahneman's system 1 and system 2 and further addresses that Type 6 methods should be capable of combinatorial reasoning. From Kautz's viewpoint, it seems that there is no NeSy approach to-date can truly meet the standard of Type 6.

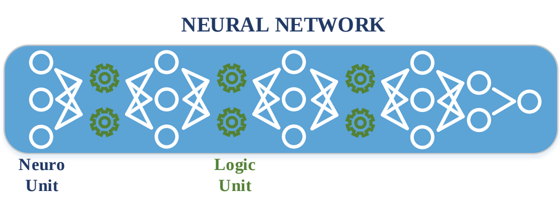

Fig. 7 Type 6: Neuro[Symbolic].

4.3 Knowledge Representation

After clarifying and categorizing the main ways in which symbolic and deep learning approaches are integrated together in this area, we turn next to symbolic knowledge, based on which symbol manipulation/logical calculus can be carried out. Understanding symbolic knowledge serves as the cornerstone of a NeSy system, another categorization dimension for neuro-symbolic approaches emerges purely from the perspective of symbolic knowledge. Specifically, the representation approaches for symbolic knowledge can be classified into two main groups: knowledge graph based and logic based, and the logic based approaches can be further categorized into two groups: propositional logic based and first order logic based.

• **Knowledge Graph:** Knowledge graphs, as a popular and effective tool for knowledge representation, contain a large amount of entities and the relationships between them. Knowledge graphs are typically directed labeled graphs, formed by representing entities – *e.g.*, *people*, *places*, *things* – as nodes, and relations between entities – *e.g.*, "is a friend of", "is located in", "is a" – as edges. It contains facts that are represented as "SPO" triples: (Subject, Predicate, Object) where Subject and Object are entities and Predicate is the relation between them. Edges are directed from subject to object, and edge labels represent different types of relations. A considerable body of works [99–105] in computer vision and NLP fields building (weak) NeSy systems upon knowledge graphs.

• **Propositional Logic:** Propositional logic, also known as boolean logic or sometimes zeroth-order logic, is the simplest form of logic where all the statements are made by propositions. A proposition is a declarative statement that is either true or false. Propositional logic studies the logical relationships between propositions which are connected via logical connectives. Typically, logical connectives (or operators), including Conjunction ("$\wedge$"), Disjunction ("$\vee$"), Negation ("$\neg$"), and Implication ("$\Rightarrow$"), are used to create compound propositions or represent a sentence logically. In propositional logic, simple statements – statements that contain no other statement as a part – are treated as indivisible wholes. Hence, propositional logic does not deal with logical relationships and properties that involve smaller parts of statements, such as the subject and predicate of a statement. Due to the simplicity of propositional logic, many early NeSy systems, such as [34, 35], consider the symbolic knowledge in the form of propositional logic. Recent work in this direction includes [98, 116, 124, 128]. For instance, [116] derives differentiable semantic loss from constraints expressed in propositional logic, hence allowing the DNNs to learn how to use the symbolic knowledge to improve the learning performance. In [98], GNNs are adopted to embed symbolic knowledge, represented as propositional formulae, into DNNs. Propositional logic is of finitary nature – only involving a finite number of propositions, and does not require sophisticated symbol manipulation operations, *i.e.*, substitution and unification, which are needed for nested terms; thus, it is relatively easy to implement propositional logic programs using DNNs [26].

• **First-Order Logic:** First-order logic, also called quantified logic or predicate logic, is an extension to propositional logic. The expressive power of propositional logic is rather limited, since it cannot express assertions about elements of a structure. Instead, first-order logic can express the relationship between those objects, achieved by allowing for *variables* in predicates bound by *quantifiers*. Specifically, first-order logic augments propositional logic with two new linguistic features, *viz.* variables and quantifiers. Variables are introduced to refer to objects of a certain type (*i.e.*, domain of discourse) and can be substituted by a specific object. The universal quantifier ("$\forall$") and existential quantifier ("$\exists$") allow us to quantify over objects. A few solutions have emerged to enable neural networks to represent first-order logic. However, most of these solutions [77, 82, 83, 93, 103, 114, 131, 132] can only handle a restricted subset of first-order logic. For example, some NeSy systems turn to Datalog [77, 82, 93, 132] for logic reasoning, or leverage GNNs to reason over local subgraph structure for inductive relation prediction [103], or regularize distributed representations via domain-specific logic rules [114]. To capture the full expressive power of first-order logic, some approaches [15, 112, 122] use fuzzy logic to translate prior knowledge, expressed as a set of first-order logic clauses, into extra training objectives. In [67, 133], first-order logic rules are compiled into differentiable operations. Another group of approaches [134–137] use first-order logic to generate a random field, based on Markov logic networks [138]. Some other approaches [79, 80, 113] adopt Prolog, a logic programming language, for knowledge representation. It shall be noted here that, due to the conflict between the infinitary nature of first-order logic – allowing to use function symbols as language primitives, and the finiteness of DNNs [26], it is much harder to model first-order logic in a connectionist setting compared with propositional logic.

4.4 Functionality

It is clear that the ultimate goal of NeSy is to install a powerful AI system with combined capabilities of both learning and reasoning. However, most existing NeSy systems are either good at learning or the best at reasoning, but rarely both [26]. According to whether to focus more on statistical learning or on symbolic reasoning, we examine the current NeSy systems from the another dimension – core functionality. This can also help us to better understand the strengths and weaknesses of the systems.

• **Learning:** Some Type 3 systems, like those neural-symbolic RL approaches [68–70, 72, 73], and the vast majority of Type 4 and Type 5 NeSy systems usually exhibit strong learning ability, but are relatively weak at logic reasoning. For example, neural-symbolic RL approaches [68–70, 72, 73] and visual reasoning algorithms [19, 91, 92, 95–97] of Type 4 are typically limited to a small set of pre-defined and simple programs/operations, and the sequences of the programs are usually generated through DNNs. For those Type 4 NeSy systems based on knowledge embedding [99–105] or training regime modification [88–90] and Type 5 NeSy systems [15, 21, 112–116, 122–124] that integrate logical knowledge as additional constraints in the loss function, they pay more attention to symbolic knowledge embedding, rather than performing logic reasoning. As the symbolic knowledge is only implicitly encoded into the weights of DNNs, they struggle with explicit reasoning and their explainability is also weak.

• **Reasoning:** Generally speaking, most Type 2 NeSy models [60, 61] and a few Type 3 systems [77–80, 82], which are built upon statistical relational learning and logic programming, yield strong symbolic reasoning ability rather than statistical learning. For the Type 2 NeSy systems [60, 61], the neural part is only involved as a submodular, while the whole system acts as a symbolic model. For those logical-programming-based Type 3 systems [77–80, 82], they allow for (differentiable) logical inference over probabilistic evidence from neural networks; however, their scalability is typically limited.

• **Reasoning and Learning:** Despite the recent progress, it is still hard to achieve a compact NeSy system that has both strong logic reasoning and expressive statistic learning abilities. In the sense of Kautz's vision, Type 6 symbolic systems could have such combined abilities. However, there are only a few models [67, 93, 127–129] can be barely recognized as Type 6.

5 Application Areas and Tasks

Recent years have witnessed the great development of NeSy, which enables more and more applications to emerge. In this section, we demonstrate some popular applications and tasks to stimulate innovations of NeSy in more application areas in the future.

• **Scientific Discovery:** Scientific discovery requires algorithms to respect real world constraints and can be interpreted and understood by scientists. This makes it quite difficult for the current-generation AI. As an important approach for explainable AI, NeSy is naturally suitable for scientific discovery.

The authors of [107] apply the symbolic regression to components of the learned GNN. The models extracts explicit physical relations and implement it in the dark matter prediction of cosmology. In [139], the symbolic regression is used to uncover an analytical equation from data and be applied to wind speed forecasting. Besides, some recent works [140–142] resolve behavior analysis of lab animals with neuro-symbolic programming *e.g.* classifying sequential animal behaviors, interpretable clustering of animal behaviors, and learning interpretable programs that describe differences between annotation behaviors of different human experts. In addition, neuro-symbolic methods are applied to the retrosynthesis and reaction prediction of organic chemistry. For instance, [143] combines the Monte Carlo tree search with the expansion policy network for discovering retrosynthetic routes. [144] proposes a conditional graph logic network to learn when rules from reaction templates should be applied. The method implicitly considers the chemical and strategic feasibility of the resulting reaction.

• **Programming Systems:** The AI community has shown great interest in developing machine learning assistants for programming systems. However, it is challenging for current neural-based AI to understand the syntactic and semantic constraints for program synthesis. NeSy incorporates the advantage of the symbolic methods that give consideration to both high-level reasoning and low-level implementation, which is naturally suitable for program generation.

Given some API calls or types that bring the information about the source code, the BAYOU system [145] could generate strongly typed Java-like source code by learning from tree-structured syntactic models or sketches of programs. The NSG system of [146] further utilizes a static analysis tool as a weak supervisor. It could generate entire Java methods given the remainder of the class that contains the method. Also, neuro-symbolic programming could be used to semantically parse the natural language and graphics into the source code. The PATOIS system [147] employs a program synthesizer with automatically learned code idioms and trains it to use code idioms for generating the program. The authors of [148] present a system that parses drawings into symbolic execution traces and then use these traces to generate the general-purpose program.

• **Question-Answering:** Question-Answering (QA) is an application area that has been developed for a long time. The task of QA is that the model infers the answer from the knowledge base composed of text or visual concepts, given a (combined) question. QA fits well with symbolic methods, especially when the questions are about the relations between objects.

In [149], the authors propose a fully differentiable network model (MAC) with cyclic memory, attention, and composition functions. It is based on the design principles of computer organization to facilitate explicit and expressive reasoning. For visual question-answering (VQA), the knowledge base is an image. A family of models disentangle vision from reasoning such as [17, 19], specifically, they utilize a perception module to learn the visual concept and then perform reasoning by generating and executing the symbolic programs for answering the question. In addition, some models bridge the gap between symbolic and neural networks in VQA by constructing the scene graph, such as the Neural State Machine (NSM) [95] and the eXplainable and eXplicit Neural Modules (XNMs) [96]. In [150] a Neural module network (NMN) is proposed to parse questions as executable programs composed of learnable modules. Then, [151] extends NMN to reasoning over the text, performing symbolic reasoning (such as arithmetic, sorting, and counting) over numbers and dates in a probabilistic and differentiable manner.

• **Robotics and Control:** Robots are complex systems with mechanical elements and controllers. When building the autonomous embodied system, we are supposed to design suitable policies to make sure the system is under reasonable mechanical constraints. Besides, safety and data efficiency are also critical for constructing the system. Therefore, the symbolic method is very appropriate for this area and has been used in robotic motion planning and control since 2007 in [152].

Decision-making is also quite challenging in robotics control. In [153], the Neuro-Symbolic Program Search (NSPS) is proposed for producing robust and expressive neuro-symbolic programs to improve the autonomous driving design. Compared with neural-network-based driving design, NSPS is more stable and safer, maintaining an interpretable symbolic decision-making process. Besides, one common strategy for decision-making is decomposing it into two levels (what to do & how to do it). In [154], the authors present an approach that utilizes neuro-symbolic skills for the bilevel planning of decision-making, which solves a wide range of tasks.

• **Visual Scene Understanding:** In visual scene understanding, the neural-based approaches only extract visual features for training the model. This makes the learned model not robust and interpretable in relation detection and image recognition. The neural-symbolic methods introduce external knowledge combined with extracted visual features to improve robustness and performance of the model.

In [155], authors build a hierarchical discriminative classifier for object detection. The model utilizes the semantics of the image labels to integrate the knowledge of inter-relations for training.

Besides, [155] uses the Markov Logic Network (MLN) to learn a knowledge base (KB) for learning the scoring function and affordance prediction (*i.e.*, predicting the relationship between the input image and the person). [15] proposes LTNs for image interpretation that involves first-order fuzzy logic specified rules in the training loss function of neural networks. LTNs could learn from noisy data efficiently with logical constraints and reason with logical formulas to describe the general characteristic of data. Compared with purely data-driven approaches, it improves the performance and robustness on the tasks of classification of an image's bounding boxes and relation detection. The authors of [98] makes use of an augmented Graph Convolutional Network (GCN) to project propositional formulae onto a manifold for symbolic knowledge embedding. Specifically, the method models the propositional logic as a graph structure, which serves a constraint for learning semantic representations of objects.

• **Mathematical Reasoning:** Mathematical reasoning is a critical skill for human beings. Computers were used for symbolic mathematics since the late 1960s [156]. However, current neural network-based intelligent systems usually perform poorly in this task and are not directly mathematically interpretable. Considering that symbols are the most common objects in mathematical reasoning, it is a natural choice to adopt NeSy for mathematical reasoning.

Some early works [157, 158] are limited by the poor reasoning ability of neural-based approaches. They mainly focus on the simple form of mathematical reasoning *i.e.* arithmetic tasks, such as integer addition and multiplication. To prompt the reasoning ability of the model, the authors of [88] propose to represent the mathematical expressions as trees and then transform trees into sequence for training the sequence-to-sequence (seq2seq) model. Their approach could successfully be used to solve more complicated mathematical problems such as function integration and ordinary differential equations.

6 Open Challenges

Though recent years have witnessed significant progress in the research of NeSy, there still exist many open problems that are worth exploring.

• **Scalability:** Current NeSy researches with an explicit logical or probabilistic reasoning modeling are often limited by rapidly increasing hardness as more complex logics required. One reason is of the growing computational complexity proportional to the expressivity of logic rules [159–161], *e.g.*, the inclusion of universal quantification over variables [160]. On the other hand, the frequent reliance on hand crafted rules or domain specific knowledge also impedes their scalability to large-scale applications in the wild [162, 163]. Owing to the superiority of connectionist components that learn from data at scale, NeSy is expected to have sufficient capacity for the full use of rich data and handling real-world challenges [164]. However, as witnessed by the state of the arts, whether it is possible to scale to the real-world complexity, without sacrificing desirable features from the symbolic aspect, remains unknown and deserves further investigations.

• **Compositional Generalization:** As we discussed in Section 3.1, compositionality, a core of human intelligence, is among the desirable characterizations where the NeSy systems are expected to be good at. It allows strong generalizability to novel reasoning problems, through decomposing and recombining the learned knowledge, in terms of symbolic representations or neural modules. Along this line, state-of-the-art researches are mainly conducted on discrete symbols, *i.e.*, natural language word tokens [60, 165]. Whether it is possible, or to what extent, to achieve the high-level composition, such as incorporating different types of logic systems (*e.g.*, modal, temporal, commonsense, epistemic, *etc.*), remains an open challenge. NeSy systems are still seeking to have the realistic property of compositionality.

• **Automatic and Comprehensive Symbolic Knowledge Acquisition:** Even though favorable progress has been made in symbolic knowledge acquisition, much of the work is studied on the non-logical symbolic level, particularly in form of conceptual knowledge graphs [166], and largely restricted to sophisticated learning algorithms that often do not scale well to deep networks [167]. Furthermore, incorporating complex logic, probabilistic relation or various data sources inevitably complicates the problem even more. We believe that more emphasis is needed on comprehensively and automatically discovering symbolic knowledge from not only the data of growing scale, but also networks with explosive dimensionality.

• **Recursive Neuro[Symbolic] Engine:** Yet another invaluable research endeavor is to make progress on an automated Neuro[Symbolic] engine (*c.f.*, Section 4.2) that is able to benefit from the interplay between the neural- and symbolic components. Primarily, the engine should be capable of combinatorial reasoning towards a level bordering the human intelligence. Besides, the neural components are designed to be trained against the useful symbolic constraints, while recursively, the symbolic components are in turn able to evolve with the higher-quality rules induced from data. This loop is closely linked to the knowledge acquisition challenge that we discussed in the previous section. It opens several promising research lines that show strong potential of (fundamentally) addressing the scalability issue, and also offers a rich alternative towards a formal realization of System 1 & System 2, as specified by Kautz.

• **(More) Human-like NeSy Applications:** As described throughout this article, there is a strong agreement that NeSy systems are among the most promising avenues towards human-like artificially intelligence [14, 54]. However, the main strands of its applications are still restricted by a handful of tasks (*c.f.*, Section 5) that only focus on basic decision-making and reasoning problems, in contrast to the large vision we hold onto the human-like capabilities, such as the causal and counterfactual reasoning, cooperation and communication at scale, *etc*. We need more challenging real-world playgrounds that seek fundamental progress on NeSy AI.

7 Conclusions

Though having a long history, NeSy remained a rather niche topic until recently when landmark advances in machine learning – pushed by the wave of deep learning – caused increasing interest in forming the bridge between neural and symbolic methods. In this work, we conducted a large-scale and up-to-date survey of the rapidly growing area, from four perspectives: **i)** A historical point of view – we provide a brief review of early research results of NeSy; **ii)** A motivation point of view – we clarify two major driving forces behind the field as well as the recent AI debate which promotes the research activity in NeSy; **iii)** A methodological point of view – we classify and analyze the contemporary NeSy systems from three dimensions: neural-symbolic integration, knowledge representation, and functionality; and **iv)** An application point of view – we outline several key application areas including scientific discovery, programming systems, question-answering, robotics and control, visual scene understanding, and mathematical reasoning. In the end, we discuss outstanding challenges and areas for future research. Although a strong NeSy system is still far from achieved, given the significant progress in AI over the past decade, we remain optimistic about the future and believe NeSy is a promising direction for the development of the next generation of AI.

References

[1] Yang, Y., Zhuang, Y. & Pan, Y. Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies. *Frontiers of Information Technology & Electronic Engineering* **22** (12), 1551–1558 (2021) .

[2] Haugeland, J. *Artificial Intelligence: The very idea* (MIT press, 1989).

[3] Zhang, J., Chen, B., Zhang, L., Ke, X. & Ding, H. Neural, symbolic and neural-symbolic reasoning on knowledge graphs. *AI Open* **2**, 14–35 (2021) .

[4] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521** (7553), 436–444 (2015) .

[5] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning internal representations by error propagation. Tech. Rep., California Univ San Diego La Jolla Inst for Cognitive Science (1985).

[6] Smolensky, P., McCoy, R. T., Fernandez, R., Goldrick, M. & Gao, J. Neurocompositional computing in human and machine intelligence: A tutorial (2022) .

[7] Smolensky, P., McCoy, R. T., Fernandez, R., Goldrick, M. & Gao, J. Neurocompositional

computing: From the central paradox of cognition to a new generation of AI systems. *arXiv preprint arXiv:2205.01128* (2022) .

[8] Fodor, J. A. & Pylyshyn, Z. W. Connectionism and cognitive architecture: A critical analysis. *Cognition* **28** (1-2), 3–71 (1988) .

[9] McCulloch, W. S. & Pitts, W. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics* **5** (4), 115–133 (1943) .

[10] Lake, B. M., Ullman, T. D., Tenenbaum, J. B. & Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences* **40** (2017) .

[11] Marcus, G. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631* (2018) .

[12] Valiant, L. G. Three problems in computer science. *Journal of the ACM* **50** (1), 96–99 (2003) .

[13] Garcez, A. *et al.* Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *Journal of Applied Logics* **6** (4), 611–632 (2019) .

[14] Garcez, A. d. *et al.* Neural-symbolic learning and reasoning: A survey and interpretation. *Neuro-Symbolic Artificial Intelligence: The State of the Art* **342**, 1 (2022) .

[15] Donadello, I., Serafini, L. & d'Avila Garcez, A. Logic tensor networks for semantic image interpretation. *International Joint Conference on Artificial Intelligence* 1596–1602 (2017) .

[16] Zhou, T., Qi, S., Wang, W., Shen, J. & Zhu, S.-C. Cascaded parsing of human-object interaction recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44** (6), 2827–2840 (2021) .

[17] Yi, K. *et al.* Neural-symbolic VQA: Disentangling reasoning from vision and language understanding. *Proceedings of the International Conference on Neural Information Processing Systems* 1039–1050 (2018) .

[18] Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B. & Wu, J. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. *International Conference on Learning Representations* (2019) .

[19] Amizadeh, S., Palangi, H., Polozov, A., Huang, Y. & Koishida, K. Neuro-symbolic visual reasoning: Disentangling "visual" from "reasoning". *International Conference on Machine Learning* 279–290 (2020) .

[20] Wang, W., Zhou, T., Qi, S., Shen, J. & Zhu, S.-C. Hierarchical human semantic parsing with comprehensive part-relation modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021) .

[21] Li, L., Zhou, T., Wang, W., Li, J. & Yang, Y. Deep hierarchical semantic segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 1246–1257 (2022) .

[22] Arabshahi, F. *et al.* Conversational neuro-symbolic commonsense reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence* 4902–4911 (2021) .

[23] Belle, V. Symbolic logic meets machine learning: A brief survey in infinite domains. *International Conference on Scalable Uncertainty Management* 3–16 (2020) .

[24] Lamb, L. C. *et al.* Graph neural networks meet neural-symbolic computing: A survey and perspective. *International Conference on International Joint Conferences on Artificial Intelligence* 4877–4884 (2021) .

[25] Yu, D., Yang, B., Liu, D. & Wang, H. A survey on neural-symbolic systems. *arXiv preprint arXiv:2111.08164* (2021) .

[26] Bader, S. & Hitzler, P. Dimensions of neural-symbolic integration – a structured survey. *arXiv preprint cs/0511042* (2005) .

[27] Kautz, H. The third AI summer: AAAI robert s. engelmore memorial lecture. *AI Magazine* **43** (1), 93–104 (2022) .

[28] Garcez, A. S., Lamb, L. C. & Gabbay, D. M. *Neural-symbolic cognitive reasoning* (Springer Berlin Heidelberg, 2009).

[29] Shi, S. *et al.* Neural logic reasoning. *International Conference on Information & Knowledge Management* 1365–1374 (2020) .

[30] Towell, G. G., Shavlik, J. W., Noordewier, M. O. *et al.* Refinement of approximate domain theories by knowledge-based neural networks. *Proceedings of the National Conference on Artificial Intelligence* 861–866 (1990) .

[31] Pollack, J. B. Recursive distributed representations. *Artificial Intelligence* **46** (1-2), 77–105 (1990) .

[32] Shastri, L. & Ajjanagadde, V. From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences* **16** (3), 417–451 (1993) .

[33] Hölldobler, S., Kalinke, Y., Ki, F. W. *et al.* Towards a new massively parallel computational model for logic programming. *ECAI'94 workshop on Combining Symbolic and Connectioninst Processing* (1991) .

[34] Garcez, A. & Zaverucha, G. The connectionist inductive learning and logic programming system. *Applied Intelligence Journal* **11** (1), 59–77 (1999) .

[35] Towell, G. G. & Shavlik, J. W. Knowledge-based artificial neural networks. *Artificial intelligence* **70** (1-2), 119–165 (1994) .

[36] Plate, T. A. Holographic reduced representations. *IEEE Transactions on Neural networks* **6** (3), 623–641 (1995) .

[37] Cloete, I. & Zurada, J. M. *Knowledge-based neurocomputing* (MIT press, 2000).

[38] Browne, A. & Sun, R. Connectionist inference models. *Neural Networks* **14** (10), 1331–1355 (2001) .

[39] Garcez, A. S. d., Broda, K., Gabbay, D. M. *et al.* *Neural-symbolic learning systems: foundations and applications* (Springer Science & Business Media, 2002).

[40] Deacon, T. W. The co-evolution of language and the brain. *WW Norlon, Nueva Ymk* (1997) .

[41] Russell, S. J. & Norvig, P. *Artificial Intelligence: A modern approach* 3 edn (Pearson, 2009).

[42] Horst, S. The computational theory of mind (2003) .

[43] Hitzler, P., Eberhart, A., Ebrahimi, M., Sarker, M. K. & Zhou, L. Neuro-symbolic approaches in artificial intelligence. *National Science Review* **9** (6) (2022) .

[44] Marcus, G. The next decade in AI: Four steps towards robust artificial intelligence. *arXiv preprint arXiv:2002.06177* (2020) .

[45] Maruyama, Y. Symbolic and statistical theories of cognition: towards integrated artificial intelligence. *International Conference on Software Engineering and Formal Methods* 129–146 (2020) .

[46] Dantsin, E., Eiter, T., Gottlob, G. & Voronkov, A. Complexity and expressive power of logic programming. *ACM Computing Surveys* **33** (3), 374–425 (2001) .

[47] Mitchell, T. M. *Machine learning* Vol. 1.

[48] Churchland, P. S. & Sejnowski, T. J. *The computational brain* (MIT press, 1994).

[49] Kiparsky, P. & Staal, J. F. Syntactic and semantic relations in panini. *Foundations of Language* 83–117 (1969) .

[50] Janssen, T. M. *et al.* Compositionality: Its historic context. *The Oxford handbook of compositionality* 19–46 (2012) .

[51] Szabó, Z. The case for compositionality. *The Oxford handbook of compositionality* **64**, 80 (2012) .

[52] Pagin, P. & Westerståhl, D. Compositionality i: Definitions and variants. *Philosophy Compass* **5** (3), 250–264 (2010) .

[53] Smolensky, P. On the proper treatment of connectionism. *Behavioral and brain sciences* **11** (1), 1–23 (1988) .

[54] Booch, G. *et al.* Thinking fast and slow in AI. *Proceedings of the AAAI Conference on Artificial Intelligence* 15042–15046 (2021) .

[55] Kahneman, D. *Thinking, fast and slow* (Macmillan, 2011).

[56] Thompson, N. C., Greenewald, K., Lee, K. & Manso, G. F. The computational limits of deep learning. *arXiv preprint arXiv:2007.05558* (2020) .

[57] Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013) .

[58] Pennington, J., Socher, R. & Manning, C. D. Glove: Global vectors for word representation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing* 1532–1543 (2014) .

[59] Silver, D. *et al.* Mastering the game of go with deep neural networks and tree search. *Nature* **529** (7587), 484–489 (2016) .

[60] Chen, X., Liang, C., Yu, A. W., Song, D. & Zhou, D. Compositional generalization via neural-symbolic stack machines. *Advances in Neural Information Processing Systems* 1690–1701 (2020) .

[61] Dang-Nhu, R. Plans: Neuro-symbolic program learning from videos. *Advances in Neural Information Processing Systems* 22445–22455 (2020) .

[62] Parisotto, E. *et al.* Neuro-symbolic program synthesis. *International Conference on Learning Representations* (2017) .

[63] Chen, X. *et al.* Neural symbolic reader: Scalable integration of distributed and symbolic representations for reading comprehension. *International Conference on Learning Representations* (2019) .

[64] Nye, M., Solar-Lezama, A., Tenenbaum, J. & Lake, B. M. Learning compositional rules via neural program synthesis. *Advances in Neural Information Processing Systems* 10832–10842 (2020) .

[65] Young, H., Bastani, O. & Naik, M. Learning neurosymbolic generative models via program synthesis. *International Conference on Machine Learning* (2019) .

[66] Valkov, L., Chaudhari, D., Srivastava, A., Sutton, C. & Chaudhuri, S. Houdini: Lifelong learning as program synthesis. *Advances in Neural Information Processing Systems* (2018) .

[67] Yang, F., Yang, Z. & Cohen, W. W. Differentiable learning of logical rules for knowledge base reasoning. *Proceedings of International Conference on Neural Information Processing Systems* 2316–2325 (2017) .

[68] Yang, F., Lyu, D., Liu, B. & Gustafson, S. Peorl: integrating symbolic planning and hierarchical reinforcement learning for robust decision-making. *Proceedings of International Joint Conference on Artificial Intelligence* 4860–4866 (2018) .

[69] Garnelo, M., Arulkumaran, K. & Shanahan, M. Towards deep symbolic reinforcement learning. *arXiv preprint arXiv:1609.05518* (2016) .

[70] Mou, L., Lu, Z., Li, H. & Jin, Z. Coupling distributed and symbolic execution for natural language queries. *International Conference on Machine Learning* 2518–2526 (2017) .

[71] de Penning, H. L. H., Garcez, A. S. d., Lamb, L. C. & Meyer, J.-J. C. A neural-symbolic cognitive agent for online learning

and reasoning. *International Joint Conference on Artificial Intelligence* (2011) .

[72] Lyu, D., Yang, F., Liu, B. & Gustafson, S. Sdrl: interpretable and data-efficient deep reinforcement learning leveraging symbolic planning. *Proceedings of the AAAI Conference on Artificial Intelligence* 2970–2977 (2019) .

[73] Jin, M. *et al.* Creativity of ai: Automatic symbolic option discovery for facilitating deep reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence* 7042–7050 (2022) .

[74] Liang, C., Berant, J., Le, Q., Forbus, K. & Lao, N. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (2017) .

[75] Jiang, Z. & Luo, S. Neural logic reinforcement learning. *International Conference on Machine Learning* (2019) .

[76] Dai, W.-Z., Xu, Q., Yu, Y. & Zhou, Z.-H. Bridging machine learning and logical reasoning by abductive learning. *Advances in Neural Information Processing Systems* **32** (2019) .

[77] Rocktäschel, T. & Riedel, S. End-to-end differentiable proving. *Advances in neural information processing systems* **30** (2017) .

[78] Minervini, P., Riedel, S., Stenetorp, P., Grefenstette, E. & Rocktäschel, T. Learning reasoning strategies in end-to-end differentiable proving. *International Conference on Machine Learning* 6938–6949 (2020) .

[79] Weber, L., Minervini, P., Münchmeyer, J., Leser, U. & Rocktäschl, T. Nlprolog: Reasoning with weak unification for question answering in natural language. *Annual Meeting of the Association for Computational Linguistics* 6151–6161 (2019) .

[80] Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T. & Raedt, L. D. Deepproblog: neural probabilistic logic programming. *International Conference on Neural Information Processing Systems* 3753–3763 (2018) .

[81] Tsamoura, E., Hospedales, T. & Michael, L. Neural-symbolic integration: A compositional perspective. *Proceedings of the AAAI Conference on Artificial Intelligence* (2021) .

[82] Si, X., Raghothaman, M., Heo, K. & Naik, M. Synthesizing datalog programs using numerical relaxation. *International Joint Conference on Artificial Intelligence* 6117–6124 (2019) .

[83] Dumancic, S., Guns, T., Meert, W. & Blockeel, H. Learning relational representations with auto-encoding logic programs. *Proceedings of the International Joint Conference on Artificial Intelligence* 6081–6087 (2019) .

[84] Allamanis, M., Chanthirasegaran, P., Kohli, P. & Sutton, C. Learning continuous semantic representations of symbolic expressions. *International Conference on Machine Learning* 80–88 (2017) .

[85] Dai, H., Tian, Y., Dai, B., Skiena, S. & Song, L. Syntax-directed variational autoencoder for structured data. *International Conference on Learning Representations* (2018) .

[86] Chen, X., Liu, C. & Song, D. Tree-to-tree neural networks for program translation. *Advances in Neural Information Processing Systems* (2018) .

[87] Jin, W., Barzilay, R. & Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. *International Conference on Machine Learning* (2018) .

[88] Lample, G. & Charton, F. Deep learning for symbolic mathematics. *International Conference on Learning Representations* (2019) .

[89] Li, Q. *et al.* Closed loop neural-symbolic learning via integrating neural perception, grammar parsing, and symbolic reasoning. *International Conference on Machine Learning* 5884–5894 (2020) .

[90] Arabshahi, F., Singh, S. & Anandkumar, A. Combining symbolic expressions and black-box function evaluations in neural programs. *International Conference on Learning Representations* (2018) .

[91] Johnson, J. *et al.* Inferring and executing programs for visual reasoning. *Proceedings of the IEEE international conference on computer vision* 2989–2998 (2017) .

[92] Dhingra, B. *et al.* Differentiable reasoning over a virtual knowledge base. *International Conference on Learning Representations* (2019) .

[93] Evans, R. & Grefenstette, E. Learning explanatory rules from noisy data. *Journal of Artificial Intelligence Research* **61**, 1–64 (2018) .

[94] Zhang, L. *et al.* Neural guided constraint logic programming for program synthesis. *Advances in Neural Information Processing Systems* (2018) .

[95] Hudson, D. & Manning, C. D. Learning by abstraction: The neural state machine. *Advances in Neural Information Processing Systems* 5903–5916 (2019) .

[96] Shi, J., Zhang, H. & Li, J. Explainable and explicit visual reasoning over scene graphs. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 8376–8384 (2019) .

[97] Vedantam, R. *et al.* Probabilistic neural symbolic models for interpretable visual question answering. *International Conference on Machine Learning* (2019) .

[98] Xie, Y., Xu, Z., Kankanhalli, M. S., Meel, K. S. & Soh, H. Embedding symbolic knowledge into deep networks. *Proceedings of the International Conference on Neural Information Processing Systems* 4233–4243 (2019) .

[99] Wang, W. *et al.* Learning compositional neural information fusion for human parsing. *Proceedings of the IEEE/CVF International Conference on Computer Vision* 5703–5713 (2019) .

[100] Wang, W. *et al.* Hierarchical human parsing with typed part-relation reasoning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 8929–8939 (2020) .

[101] Lin, B. Y., Chen, X., Chen, J. & Ren, X. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *Conference on Empirical Methods in Natural Language Processing* 2829–2839 (2019) .

[102] Lv, S. *et al.* Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. *Proceedings of the AAAI Conference on Artificial Intelligence* **34** (05), 8449–8456 (2020) .

[103] Teru, K., Denis, E. & Hamilton, W. Inductive relation prediction by subgraph reasoning. *International Conference on Machine Learning* 9448–9457 (2020) .

[104] Gu, J. *et al.* Scene graph generation with external knowledge and image reconstruction. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* 1969–1978 (2019) .

[105] Marino, K., Chen, X., Parikh, D., Gupta, A. & Rohrbach, M. Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 14111–14121 (2021) .

[106] Cohen, W. W., Sun, H., Hofer, R. A. & Siegler, M. Scalable neural methods for reasoning with a symbolic knowledge base. *International Conference on Learning Representations* (2020) .

[107] Cranmer, M. *et al.* Discovering symbolic models from deep learning with inductive biases. *Advances in Neural Information Processing Systems* 17429–17442 (2020) .

[108] Liang, X., Hu, Z., Zhang, H., Lin, L. & Xing, E. P. Symbolic graph reasoning meets convolutions. *Advances in Neural Information Processing Systems* (2018) .

[109] Asai, M. & Muise, C. Learning neural-symbolic descriptive planning models via cube-space priors: the voyage home (to strips). *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence* (2021) .

[110] Garg, S., Bajpai, A. *et al.* Symbolic network: generalized neural policies for relational mdps. *International Conference on Machine Learning* (2020) .

[111] Badreddine, S., Garcez, A. d., Serafini, L. & Spranger, M. Logic tensor networks. *Artificial Intelligence* **303**, 103649 (2022) .

[112] Liang, Y. & Van den Broeck, G. Learning logistic circuits. *Proceedings of the AAAI Conference on Artificial Intelligence* 4277–4286 (2019) .

[113] Demeester, T., Rocktäschel, T. & Riedel, S. Lifted rule injection for relation embeddings. *Conference on Empirical Methods in Natural Language Processing* 1389–1399 (2016) .

[114] Hu, Z., Ma, X., Liu, Z., Hovy, E. & Xing, E. Harnessing deep neural networks with logic rules. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* 2410–2420 (2016) .

[115] Stewart, R. & Ermon, S. Label-free supervision of neural networks with physics and domain knowledge. *AAAI Conference on Artificial Intelligence* (2017) .

[116] Xu, J., Zhang, Z., Friedman, T., Liang, Y. & Broeck, G. A semantic loss function for deep learning with symbolic knowledge. *International Conference on Machine Learning* 5502–5511 (2018) .

[117] Muralidhar, N., Islam, M. R., Marwah, M., Karpatne, A. & Ramakrishnan, N. Incorporating prior domain knowledge into deep neural networks. *2018 IEEE International Conference on Big Data* 36–45 (2018) .

[118] van Krieken, E., Acar, E. & van Harmelen, F. Analyzing differentiable fuzzy logic operators. *Artificial Intelligence* **302**, 103602 (2022) .

[119] Wang, W. & Pan, S. J. Integrating deep learning with logic fusion for information extraction. *Proceedings of the AAAI Conference on Artificial Intelligence* **34** (05), 9225–9232 (2020) .

[120] Li, T. & Srikumar, V. Augmenting neural networks with first-order logic. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (2019) .

[121] Hoernle, N., Karampatsis, R. M., Belle, V. & Gal, K. Multiplexnet: Towards fully satisfied logical constraints in neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence* (2022) .

[122] Diligenti, M., Gori, M. & Sacca, C. Semantic-based regularization for learning and inference. *Artificial Intelligence* **244**, 143–165 (2017) .

[123] Marra, G., Giannini, F., Diligenti, M. & Gori, M. Lyrics: A general interface layer to integrate logic inference and deep learning. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* 283–298 (2019) .

[124] Fischer, M. *et al.* Dl2: Training and querying neural networks with logic. *International Conference on Machine Learning* 1931–1941 (2019) .

[125] Giunchiglia, E. & Lukasiewicz, T. Multi-label classification neural networks with hard logical constraints. *Journal of Artificial*

Intelligence Research **72**, 759–818 (2021) .

[126] Demeter, D. & Downey, D. Just add functions: A neural-symbolic language model. *Proceedings of the AAAI Conference on Artificial Intelligence* (2020) .

[127] Dong, H. *et al.* Neural logic machines. *International Conference on Learning Representations* (2018) .

[128] Wang, P.-W., Donti, P., Wilder, B. & Kolter, Z. Satnet: Bridging deep learning and logical reasoning using a differentiable satisfiability solver. *International Conference on Machine Learning* 6545–6554 (2019) .

[129] Cai, J., Shin, R. & Song, D. Making neural programming architectures generalize via recursion. *International Conference on Learning Representations* (2017) .

[130] Garcez, A. d. & Lamb, L. C. Neurosymbolic AI: the 3rd wave. *arXiv preprint arXiv:2012.05876* (2020) .

[131] Yang, Y. & Song, L. Learn to explain efficiently via neural logic inductive learning. *International Conference on Learning Representations* (2019) .

[132] Hohenecker, P. & Lukasiewicz, T. Ontology reasoning with deep neural networks. *Journal of Artificial Intelligence Research* **68**, 503–540 (2020) .

[133] Cohen, W. W. Tensorlog: A differentiable deductive database. *arXiv preprint arXiv:1605.06523* (2016) .

[134] Qu, M. & Tang, J. Probabilistic logic neural networks for reasoning. *Proceedings of the International Conference on Neural Information Processing Systems* 7712–7722 (2019) .

[135] Zhang, Y. *et al.* Efficient probabilistic logic reasoning with graph neural networks. *International Conference on Learning Representations* (2019) .

[136] Marra, G., Diligenti, M., Giannini, F., Gori, M. & Maggini, M. Relational neural machines. *arXiv preprint arXiv:2002.02193* (2020) .

[137] Marra, G. & Kuželka, O. Neural markov logic networks. *Uncertainty in Artificial Intelligence* 908–917 (2021) .

[138] Richardson, M. & Domingos, P. Markov logic networks. *Machine Learning* **62** (1), 107–136 (2006) .

[139] Abdellaoui, I. A. & Mehrkanoon, S. Symbolic regression for scientific discovery: an application to wind speed forecasting 01–08 (2021) .

[140] Shah, A. *et al.* Learning differentiable programs with admissible neural heuristics. *Advances in neural information processing systems* **33**, 4940–4952 (2020) .

[141] Zhan, E., Sun, J. J., Kennedy, A., Yue, Y. & Chaudhuri, S. Unsupervised learning of neurosymbolic encoders. *arXiv preprint arXiv:2107.13132* (2021) .

[142] Tjandrasuwita, M., Sun, J. J., Kennedy, A. & Yue, Y. Interpreting expert annotation differences in animal behavior. *CVPR 2021 Workshop on CV4Animation* (2021) .

[143] Segler, M. H., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature* **555** (7698), 604–610 (2018) .

[144] Dai, H., Li, C., Coley, C., Dai, B. & Song, L. Retrosynthesis prediction with conditional graph logic network. *Advances in Neural Information Processing Systems* **32** (2019) .

[145] Murali, V., Qi, L., Chaudhuri, S. & Jermaine, C. Neural sketch learning for conditional program generation. *arXiv preprint arXiv:1703.05698* (2017) .

[146] Mukherjee, R. *et al.* Neural program generation modulo static analysis. *Advances in Neural Information Processing Systems* **34**, 18984–18996 (2021) .

[147] Shin, E. C., Allamanis, M., Brockschmidt, M. & Polozov, A. Program synthesis and semantic parsing with learned code idioms. *Advances in Neural Information Processing Systems* **32** (2019) .

[148] Ellis, K., Ritchie, D., Solar-Lezama, A. & Tenenbaum, J. Learning to infer graphics programs from hand-drawn images. *Advances in neural information processing systems* **31** (2018) .

[149] Hudson, D. A. & Manning, C. D. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067* (2018) .

[150] Andreas, J., Rohrbach, M., Darrell, T. & Klein, D. Neural module networks 39–48 (2016) .

[151] Gupta, N., Lin, K., Roth, D., Singh, S. & Gardner, M. Neural module networks for reasoning over text. *arXiv preprint arXiv:1912.04971* (2019) .

[152] Belta, C. *et al.* Symbolic planning and control of robot motion. *Robotics & Automation Magazine, IEEE* **14**, 61 – 70 (2007) .

[153] Sun, J., Sun, H., Han, T. & Zhou, B. Neuro-symbolic program search for autonomous driving decision module design (2020) .

[154] Silver, T., Athalye, A., Tenenbaum, J. B., Lozano-Perez, T. & Kaelbling, L. P. Learning neuro-symbolic skills for bilevel planning. *arXiv preprint arXiv:2206.10680* (2022) .

[155] Marszalek, M. & Schmid, C. Semantic hierarchies for visual object recognition 1–7 (2007) .

[156] Moses, J. Macsyma-the fifth year. *ACM Sigsam Bulletin* **8** (3), 105–110 (1974) .

[157] Zaremba, W. & Sutskever, I. Learning to execute. *arXiv preprint arXiv:1410.4615* (2014) .

[158] Kaiser, Ł. & Sutskever, I. Neural gpus learn algorithms. *arXiv preprint arXiv:1511.08228* (2015) .

[159] Bianchi, F. & Hitzler, P. On the capabilities of logic tensor networks for deductive reasoning. *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering* (2019) .

[160] Bianchi, F., Palmonari, M., Hitzler, P. & Serafini, L. Complementing logical reasoning with sub-symbolic commonsense. *International Joint Conference on Rules and Reasoning* 161–170 (2019) .

[161] Eberhart, A., Ebrahimi, M., Zhou, L., Shimizu, C. & Hitzler, P. Completion reasoning emulation for the description logic el+. *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering* (2020) .

[162] Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B. & Wu, J. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. *International Conference on Learning Representations* (2018) .

[163] Chaturvedi, I., Satapathy, R., Cavallari, S. & Cambria, E. Fuzzy commonsense reasoning for multimodal sentiment analysis. *Pattern Recognition Letters* **125**, 264–270 (2019) .

[164] Biggio, L., Bendinelli, T., Neitz, A., Lucchi, A. & Parascandolo, G. Neural symbolic regression that scales. *International Conference on Machine Learning* 936–945 (2021) .

[165] Kim, J., Ravikumar, P., Ainslie, J. & Ontanon, S. Improving compositional generalization in classification tasks via structure annotations. *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* 637–645 (2021) .

[166] Ji, S., Pan, S., Cambria, E., Marttinen, P. & Philip, S. Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems* **33** (2), 494–514 (2021) .

[167] Lavrac, N. & Dzeroski, S. Inductive logic programming. *WLP* 146–160 (1994) .