James A. Crowder · John Carbone
Shelli Friess

Artificial Psychology

Psychological Modeling and Testing of AI Systems

Springer

Artificial Psychology

James A. Crowder • John Carbone • Shelli Friess

Artificial Psychology

Psychological Modeling and Testing of AI Systems

Springer

James A. Crowder
Colorado Engineering Inc.
Colorado Springs, CO, USA

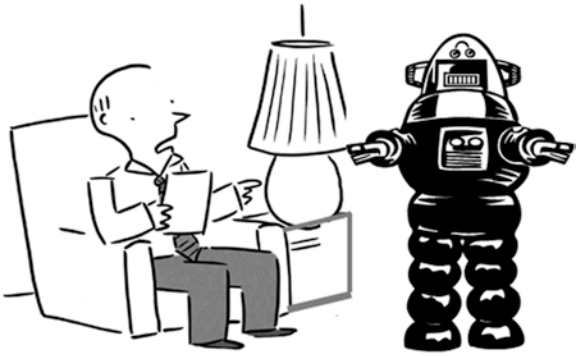
John Carbone
Forcepoint
Austin, TX, USA

Shelli Friess
Walden University
Minneapolis, MN, USA

ISBN 978-3-030-17079-0 ISBN 978-3-030-17081-3 (eBook)
https://doi.org/10.1007/978-3-030-17081-3

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface: What Does It Mean to Test an Artificially Intelligent Entity?



So your father was a trash compactor –
do you feel embarrassed by that?

We started the journey of writing this book in order to address what we felt was a great need in the engineering industry: the notion of testing artificial intelligence systems. Our initial research examined defining a single methodology and path that can be used by designers and implementers in understanding how to adequately test systems industry developed that have built-in objectives: to think, learn, reason, and make human-like decisions. As expected, we quickly came to understand that a single methodology or process could not be established. Instead, our research concluded that we needed to develop robust processes, which did not currently exist, to effectively address not only the different aspects of artificial intelligence but also how testing for each type of artificially intelligent system could be accomplished.

To that end, this book has taken on the form of a series of monographs, each dealing with different aspects of artificial intelligence systems. We consider specifically the comprehensive artificial intelligence range from a manual system to full self-evolving systems. Additionally, the algorithmic range spans from standard machine learning to fully cognitive thinking and reasoning systems that must be tested

continually to achieve successful growth, like humanistic testing and evaluation. We acknowledge that this is a wide spectrum with deserving consideration for all aspects of artificial intelligence. Hence, we believe that this book will at least serve to begin to build a foundation for future research. Since currently there is little research and the lack of methodology or "script" that can be followed for the comprehensive assessment or testing of artificial intelligence, this book is but the beginning of a lifelong journey that we believe engineers and scientists must undertake to continually assess our critical methods, results, and thinking about artificial intelligence systems and how they should properly be understood and tested. After all, what we do *not* want is to hear our artificially intelligent system telling us:

"No Dave, I don't think I can do that."

Colorado Springs, CO, USA James A. Crowder
Austin, TX, USA John Carbone
Minneapolis, MN, USA Shelli Friess

Contents

List of Figures

List of Tables

Chapter 1
Introduction: Psychology and Technology

There has been much talk over the last few years about the perils of the use of artificial intelligence in virtually everything we touch. From our phones to our cars, and everything in between, artificial intelligence is an integral part of our existence. Many prominent people, like Elon Musk and Stephen Hawking, have warned about the potential for machines to take over and cause havoc in the lives and very existence of humans. Hollywood has made untold billions of dollars painting doom-and-gloom scenarios about artificial intelligence and robots within our society today and in the future. But what is the true reality? We continually push to create increasingly intelligent systems/machines that attempt to learn, think, and reason like humans. Therefore, our first question becomes, when presented with this challenge is:

Which types of people do we want robots to learn, think, and reason like?

Do you want a Stephen Hawking? Do you want a Charles Manson? Do you want any of a host of past or current world dictators? All these people learn and think and reason, but all of them do it very differently from one another. To say you want a system that learns, thinks, and reasons like people is to say you want to give the computer/robot the ability to self-adapt, to create (through experience and learning) an adult intelligence and capabilities that is people-like in nature. No matter which "version" of human thinking and reasoning is desirable, the main question to be asked is:

"How do we test it to know if it's working correctly?"

The fundamental problem with testing learning, thinking, reasoning, artificial intelligent systems is:

"What does it mean for it to work correctly?"

Upon reviewing the limitations of classical system test theory and implementations, we begin to understand the conundrums of applying these well-known processes for testing AI systems.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_1

1.1 Classical System Testing

Figure 1.1 illustrates the classical engineering cycle for development, verification, and validation of systems that do not contain AI components/algorithms:

Test Engineering processes are responsible for determining how to test the complete set of system components including the comprehensive finished system such that it produces 100% coverage of all technical, performance, non-technical, and quality requirements (e.g., reliability, maintainability, and availability). Test engineering should be (but often is not) included in the early stages of the design process. This helps to ensure the system design includes testability, maintainability, and manufacturability. In short, the classical system test engineering process ensures that the capabilities can be built and readily tested and repeated. When the test engineering process is bypassed, testability becomes overly complicated later in the implementation process causing bottlenecks and delays in development, testing, delivery, and maintenance of the overall system. The goal of test engineering, in

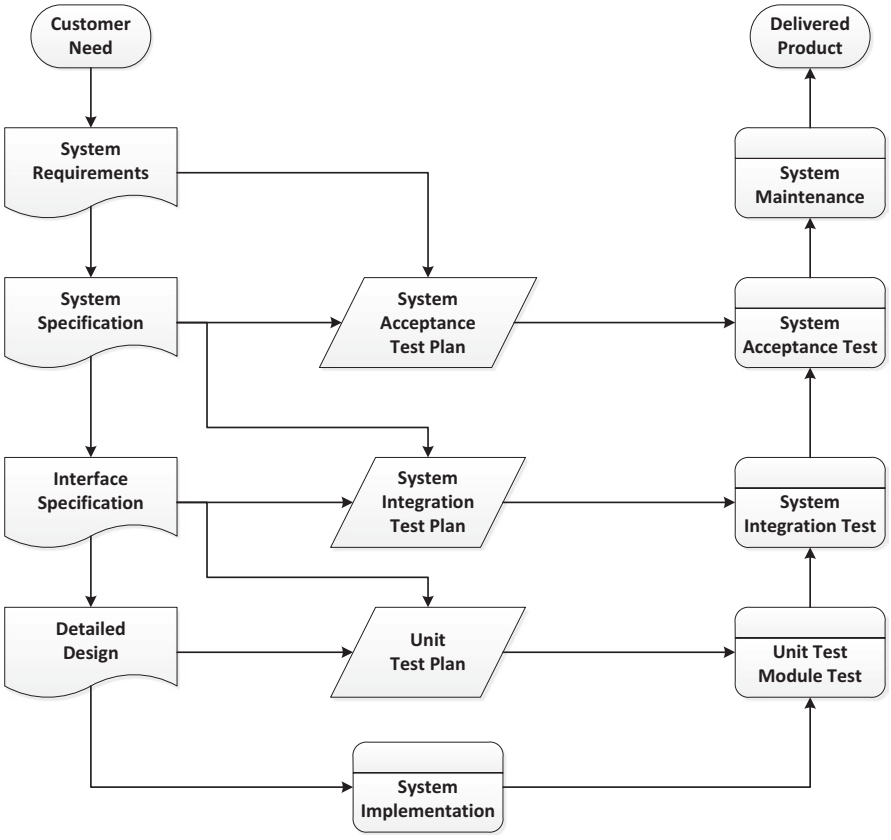


Fig. 1.1 Classical system test cycle

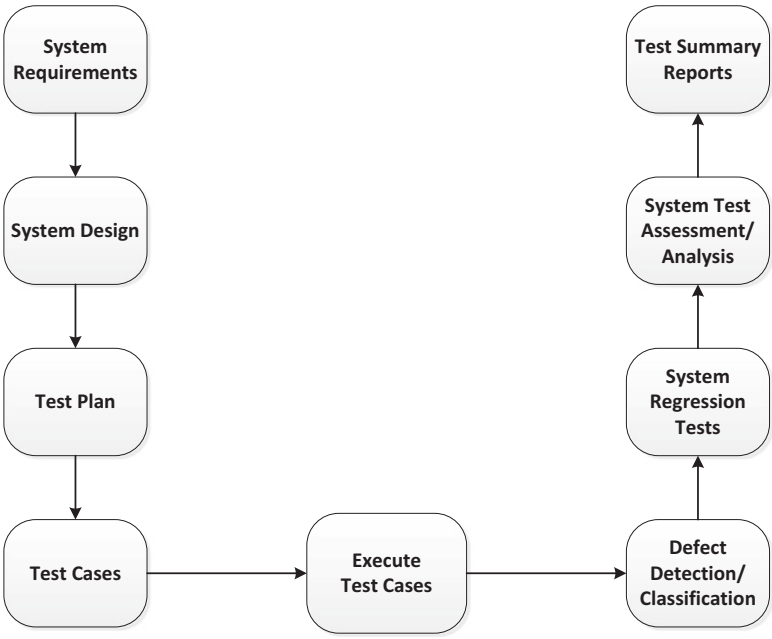


Fig. 1.2 Classical engineering test life cycle

many companies, is to design a set of automated tests that can be utilized to ensure the proper execution of the system. These automated (often called regression tests) can be utilized to test the basic functionality of a system that has been modified, enhanced, or had "bug" fixes uploaded into the system. These constitute a continuous cycle of system tests that assume the original goals and which answers should remain the same. Generally, system tests utilize a static set of test data to initiate and test the complete system functionality. Figure 1.2 illustrates this process. Each testing level is shown in Fig. 1.1, Unit, Integration, Acceptance, and Maintenance, cycle through similar processes, as depicted in Fig. 1.2.

However, as industry begins to develop artificial intelligence components and/or systems that begin to learn and adapt, then how do we design test cases, a priori, for functions, situations, and decision-making, we didn't know the system could or would eventually learn. In addition, there are types of tests that classical system test theory doesn't include. Figure 1.3 illustrates a classical systems/test engineering test progression. The first set of tests, called "Sanity Testing," is a basic set of tests to ensure the system compiles, starts, and behaves correctly for basic but important core functionality. The overall aim of sanity testing can be thought of as a build verification test or basic acceptance test. The system boots and is properly initialized; in short, achieves a robust foundational state with all proper resources in place and doesn't crash. These tests are generally not scripted and utilized by the test engineering team to determine if the system is ready for formal testing and validates that the system has no missing basic functionality.

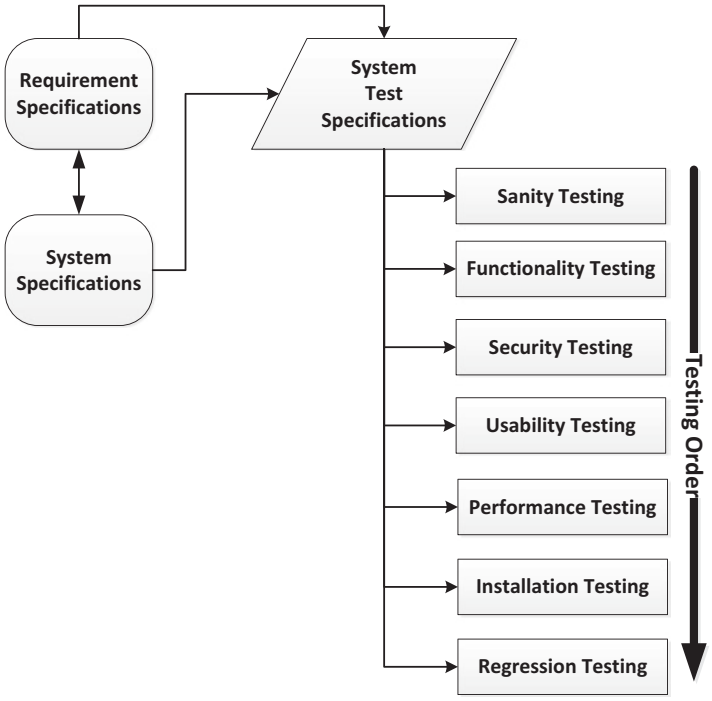


Fig. 1.3 Types of classical system tests and test order

While these classical tests have served industry well for developing robust systems, these systems do not contain artificial intelligence algorithms/units/subsystems/etc. Therefore, we propose current test solutions are not adequate for systems that change, evolve, and learn over time as they interact with their environment. That said, even artificial intelligence systems require basic sanity testing to ensure the system includes core functionality and that the training aspects of the system (e.g., artificial intelligence knowledge training) provide adequate initialization capabilities for a given system, which could be different, depending on the domain the system is being created to operate within.

1.2 AI Testing Philosophy

The core of every current AI system is driven by software (although not required for analog systems, see Chap. 10), algorithms, and Ones and Zeroes like any other modern system. However, AI effectiveness in the near-term may be largely dependent upon the quality and quantity of the training data for the system to learn from and the closeness of the data used in relation to algorithmic functionality and objectives of the system. In many cases, training data must be created in order to provide

initial learning for the system. This is largely due to the significant increase in Machine Learning implementations and Big Data research and processing. In the case of autonomous or semi-autonomous systems, this data may come from the AI system's experience interacting with its environment. This is certainly the case with self-driving cars. They are given a set of rules, they are given training data from which they derive an initial set of conditions and cause/effect scenarios, but they also continually learn and reassess as they interact with the world/roads around them. An example might be smart phones and car navigation systems. They certainly cannot be trained to clearly understand every single accent, dialect, and language that exists in the world. It is possible for intelligent systems to learn unexpected things, not intended to be learned. This is a type of "implicit learning," or learning that the system does not realize it is learning. An example study from MIT Media Labs trained an AI-powered entity named Norman,[1] using data that comes from dark, socially unacceptable (depending on your bent) sides of the web. Norman developed psychopathic tendencies. Once trained, Norman was asked to visualize a photograph of people standing around a window. Interpretation software saw people standing around a window as people standing around a window. Because of Norman's training, Norman interpreted the picture as people who would probably jump out the window. Training data is important to an AI entity, and training data with biases inherently drive the learning process within an AI to have biases toward its interpretation of data within its environment, like humanistic processes. These potential pitfalls compel researchers to look at not just the AI algorithm model validation when we test, but also data validation, which, in the case of unsupervised learning, is difficult to facilitate. Figure 1.4 illustrates a High-Level Artificial Intelligence System product testing life cycle. One that recognizes the self-adapting, learning nature of artificial intelligence systems. The testing life cycle of a system that learns, adapts, and continually self-improves is an ever-ongoing test cycle that must continually assess how the system changes. Test cases, philosophy, strategy, etc. must adapt to accommodate any "newly adapted" system capabilities. Lastly, these tests are recommended to be deployed internally and should adhere to overarching system guidelines and rules which govern all operations, functions, and system boundary conditions.

1.2.1 AI Adaptability Testing

There have been three notable accidents involving self-driving cars. One at the 2018 Consumer Electronics Show (CES) where a self-driving car ran into an autonomous robot that had inadvertently wandered onto the street. The robot company must determine why the robot wandered off its course onto the roadway and the auto company must decide why the self-driving car did not trigger its onboard systems to avoid the robot. This highlights some of the problematic issues with self-adapting

[1] norman-ai.mit.edu/

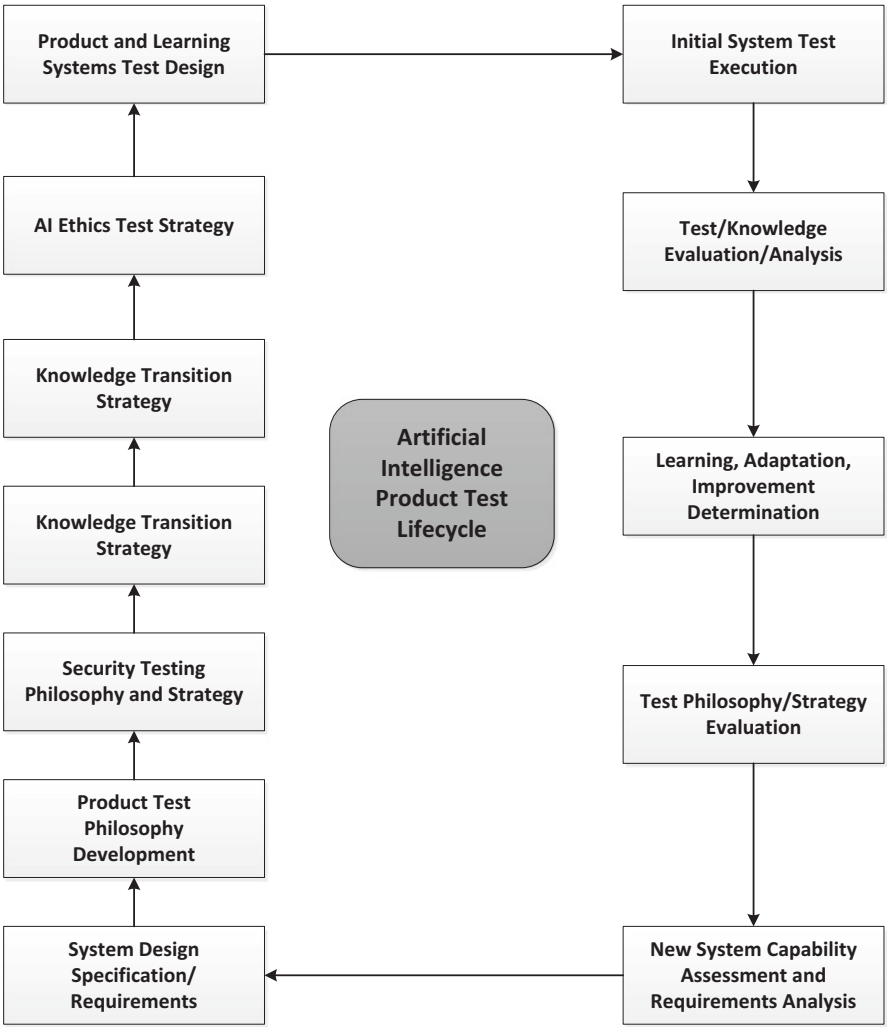


Fig. 1.4 High-level artificial intelligence system testing life cycle

autonomous systems. Namely that it is improbable to think that every scenario could be considered to train an artificially intelligent system to react properly in all possible conditions. And, if the system is created to continually learn, there's no guarantee it will learn and adapt the way the designers intended.

Artificial Intelligence systems, in general, are designed to operate within the context of existing systems and environments (e.g., self-driving cars must navigate traffic with other non-self-driving vehicles). Generally, self-driving cars are designed to handle very specific problems and situations like manufacturing or warehouse robots which move merchandise within a warehouse. Even in these cases, systems must be accountable to handle some changing basic conditions.

Therefore, we must require a system-level, or holistic, approach to assessing Artificial Intelligence systems. Here is an example. One state decided, under its Medicaid Waiver Program to automate the healthcare beneficiary services system. Under this old program, assessor-driven interviews were used to decide the hours and frequency of the available caretaker services. When they went to an automated system, in order to improve efficiency, service hours for beneficiaries were reduced, in some cases substantially. No blanket notifications were sent to the Medicare beneficiaries, which resulted in a huge increase in complaints and grievances. Unfortunately, Medicare beneficiaries were not provided responses to their grievance

Table 1.1 Artificial intelligent entity test philosophy considerations

Consideration, definition, or constraint	Discussion
AI system	An organized entity made up of generally more complex, interrelated, and interdependent parts
Boundaries	Barriers that define a system and distinguish one system from others in the environment
Homeostasis	How resilient is the Artificial Intelligence entity toward external factors and maintaining its key characteristics?
Adaptation	Can the Artificial Intelligent entity self-adapt, i.e., making the internal changes necessary to continually improve, while protecting itself, and while fulfilling its purpose within its changing environment?
Reciprocal transactions	Can, and if so how does, the Artificial Intelligent entity deal with circular or cyclical transactions that all system must engage in, such that they influence each other? How can the Artificial Intelligent entity engage its environment?
Resource optimization	What is the rate of energy transfer between the system and its environment during the time it is functioning to achieve established objectives? And, what is the rate of energy transfer when the Artificial Intelligent entity is dormant, if it ever is.
Trust	What is the measure of Trust humans must have in each of an AI's functions and activities while it works to achieve established objectives? What is the measure of ambiguity with respect to the functions it has to perform, and the measure of monitoring required to establish proper Trust?
Mesosystems	What are the relationships between the Artificial Intelligent entity and other systems in a given environment? This would be different for every domain difference.
Microsystems	What are the relationships between different cognitive components (possibly separate agents) within the same Artificial Intelligent entity and/or system?
Exosystems	What is the relationship between the Artificial Intelligent entity and systems within its environment that may have a direct effect on the lower level or seemingly unrelated systems in the environment?
Chronosystems	Can significant "life events" effect the overall learning and adaptation of the Artificial Intelligent entity? Is the timing of those events relevant to how the system adapts and learns?

appeals because the automated system that was being used was too complex to be expressed in natural language terms. In short, the human–systems interface was not created to interface in natural language terms. The system was not designed, tested, nor configured for the users it was intended for. Artificial Intelligence systems must have the test philosophy and strategies built into the overall design of the system and continually updated and adapted as the system updates and adapts (learns). Table 1.1 below describes, at a high-level, the basic considerations, definitions, and constraints that must be accounted for when designing test strategies for Artificial Intelligence systems.

1.2.2 Testing AI Trust

As engineers determine where an AI design sits on the AIC, the considerations expressed in Table 1.1 are addressed specifically to the level of automation/autonomy which the AI will ultimately be designed with and most importantly the level of Trust that the AI will be afforded by the humans who interact, ultimately control and are responsible for its operational activities, successes, and failures.

Trust is a function of need vs. risk and more importantly perceived risk when balancing total cost of system ownership and potential cost in potential lives depending on what challenging environment the AI is exposed to during operations. Figure 1.5 below describes the traditional System Engineering "V" in terms of where Trust must be considered as part of AI design activities.

The objective in measuring trust is to model knowledge as *n*-dimensional dependencies of key system attributes and measure if knowledge has met a pre-defined threshold established for Trust. Later, in this book the tools, Knowledge Relativity (KR) and Knowledge Density equations support modeling Trust for each key system attribute within your AI system. It should also be noted that AI key system attributes are system and mission environment specific and thus, have key operational time constraints which should be identified early and be validated during any

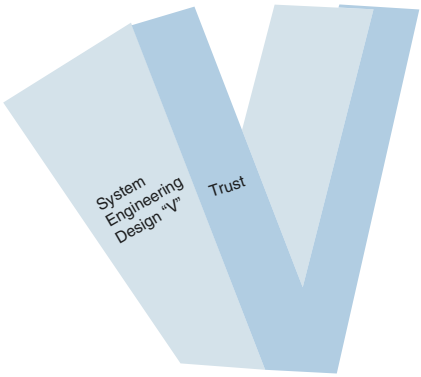


Fig. 1.5 AI system trust "V"

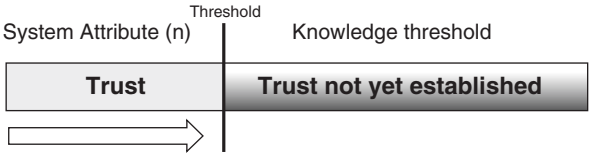


Fig. 1.6 Trust thresholding

Test and Evaluation activities and remain part of any regression testing. Figure 1.6 describes this challenge.

Furthermore, it is critical that engineers understand that developing AI systems is inherently multi-/transdisciplinary and thus, each domain an AI system is designed for requires domain specific measures of effectiveness and trust techniques. Some example key system attributes established during design are:

1. Observable and non-observable behaviors
2. All sensed parameters
3. System health parameters
4. Control parameters

1.3 Overview of the Book

The book is divided into three major sections. Chapters 2 through 4 deal with high-level concepts for artificial intelligent systems, those of high-level systems thinking, the information continuum theory for artificial intelligence, and the constructs and methods required for humans and artificial intelligence to effectively communicate as systems become more sophisticated and better able to think, reason, and articulate with their human counterparts. Chapters 5 through 11 deal with subjects at a lower level that are required for a fully artificial intelligent Self-Evolving Life Form (SELF). The notions of artificial creativity, continuous life-long machine learning, artificial reasoning and inferencing, and cognitive control of an artificial intelligent entity are discussed at length, in the context of facilitating a SELF. In order for a SELF to act autonomously and keep track of its own health and status, Chap. 8 discusses the notion of Integrated System Health within an artificial intelligent entity. Major Sect. 1.3, Chaps. 12 and 13 deal with architectural and overall test issues, providing notions of high-level data architectures and the problems we see coming in truly artificial intelligent systems; the issue of implicit learning and how to test for something in an artificial intelligent entity that is extremely difficult to measure in humans. Finally, we wrap up the book with a discussion of what we see are the next steps for artificial intelligence as the world moves ever closer to humanly intelligent robots. What follows is a synopsis of each chapter.

1.3.1 Chapter 2: System-Level Thinking for Artificial Intelligent Systems

Chapter 2 provides a discussion of system-level thinking and how it applies to artificial intelligent systems. We present issues that must be addressed and researched that allow a SELF to understand how every part of its system affects the other parts of its system, and, in turn, affect the behavior of the entire system. These issues will become of increasing importance as artificial intelligence is infused into more and more parts of systems humans utilize and interface with.

1.3.2 Chapter 3: Psychological Constructs for AI Systems: The Information Continuum

Chapter 3 explores the theory of information flows within a human neuron and how those apply to neurons within an artificial intelligent system. This facilitates artificial intelligent cognitive system theory that is discussed throughout the rest of the book.

1.3.3 Chapter 4: Human–AI Collaboration

Human Needs Engineering (HUMANE) has become a necessary component to all engineering disciplines. As we move toward more autonomous systems, there will still always be a need for human-in-the-loop to oversee the activities and decisions acted on by the artificial intelligence entity. This drives the need for architectures, algorithms, and methods for effective human–AI communication and collaboration. Additionally, as proportions rapidly increase and resources rapidly decline, more effective, real-time, automated, and dynamically human interactive systems become required. Here we discuss research within highly automated and autonomous domains. This research shows promise in designing the critical infrastructure that is needed to improve human-system collaboration awareness and quality of service (QoS). Hence, to improve decision-making, an Artificial Intelligence System (AIS), in order to be truly autonomous, is provided with a real-time, human-like, cognition-based framework for information.

1.3.4 Chapter 5: Abductive Artificial Intelligence Learning Models

The need for artificial intelligent systems to learn and reason implies the ability to form and test hypotheses.[2] In this chapter, we describe and explore abductive learning that involves finding explanations for sets of observations that artificial intelligent systems might encounter when interfacing with their respective environments. We discuss these learning models and their implications for artificial reasoning.

1.3.5 Chapter 6: Artificial Creativity and Self-Evolution: Abductive Reasoning in Artificial Life Forms

In this chapter, we consider that creativity is a directly related problem-solving activity in which explorations of problem spaces lead to the expansion of belief domains. We believe successful expansion of beliefs in an artificial cognitive system is initiated by algorithms that provide updates of the artificial cognitive system's Conceptual Ontology. Chapter 6 discusses the general heuristics of the hypothesis generation process to guide the support and rebuttal informational search processes and problem-solving activities, which includes strategies for examining, comparing, altering and combining concepts, strings of symbols, and the heuristics themselves. This capability will be necessary as we ask artificial intelligent system to solve more and more complex problems and explain multi-faceted situations.

1.3.6 Chapter 7: Artificial Intelligent Inferences Utilizing Occam Abduction

Chapter 7 discusses a further refinement of abductive reasoning, called Occam Abduction. Occam Abduction focuses on finding the smallest and simplest set of explanations for observations an artificial intelligent entity encounters, thereby minimizing the use of limited resources. We postulate that Occam Abduction algorithms and the hypothesis-driven methods that instantiate it work well within a multi-agent software artificial intelligent framework.

[2] Bergman, M. and Paavola, S. 2019. Hypothesis as a Form of Reasoning. Retrieved from Commens Dictionary: Peirce's Terms in His Own Words, http://www.commens.org/dictionary.

1.3.7 Chapter 8: Artificial Neural Diagnostics and Prognostics: Self-Soothing in Cognitive Systems

One of the first steps in system-level thinking within an artificial intelligent entity is the ability to understand its own health and status and how every part of its system affects the rest of its components/subsystems, etc. Chapter 8 discusses this self-assessment concept for artificial intelligent systems from the context of self-soothing methods in human neuropsychology. We discuss the notions of artificial emotions, tied to emotional memories within the context of cognition for system-level diagnostic and prognostics within the artificial intelligent system.

1.3.8 Chapter 9: Ontology-Based Knowledge Management for Artificial Intelligent Systems

No system, whether human or artificial, has or will have unlimited resources. The handling of knowledge, its processing, storage, retrieval, and maintenance is important for systems that are and will be designed to continually take in data/information, continually learn and adapt to their environments for a long period of time. Chapter 9 discusses the use of ontologies as enterprise modeling and a metadata standard for artificial intelligent systems. Emphasis is placed on knowledge sharing, considering discussions throughout the book on Human–AI communication/collaboration.

1.3.9 Chapter 10: Cognitive Control of Self-Evolving Life Forms (SELF) utilizing Artificial Procedural Memories

Memories, regardless of whether we are talking human, animal, or artificial intelligent systems, involve the acquisition, categorization, classification, storage, and retrieval of information. Any artificial intelligent system must have the ability to recall information and keep track of knowledge of events and analysis of them throughout its "lifetime." Chapter 10 discusses the use of artificial procedural memories as part of the overall memory system for artificial intelligent entities, to capture learning, especially learning that involves understanding how to perform tasks, which it may need to do many times over again as it interacts with its environment. This chapter describes a cognitive architecture for control of artificial intelligent systems; incorporating artificial procedural memory creation and recall as part of the overall cognitive system.

1.3.10 Chapter 11: Methodologies for Continuous, Life-Long Machine Learning for AI Systems

Recently, DARPA, the Department of Defense's Defense Advanced Research Projects Agency has kicked off a new initiative on "Life-Long Machine Learning.[3]" Current machine learning methods are too static and minimally adaptive enough to create self-adaptive artificial intelligent entities that can be in the field for long periods (i.e., years or possibly decades) and continue to learn, adapt, reinterpret memories, and continue to gain vast amounts of knowledge. New methods are required to accommodate long-term, continuous learning among artificial intelligent systems. The objectives of this chapter are to look at new architectures to facilitate life-long machine learning that require controls and mechanisms like human brain functions.

1.3.11 Chapter 12: Implicit Learning in Artificial Intelligence

One of the issues associated with a true understanding of learning is the problem of implicit learning. Implicit learning appears to be a fundamental and continuous process in the cognitive processes of any entity. Implicit learning is that learning that happens without conscious thought or conscious awareness that learning has occurred. We feel this is a potentially major problem within artificial intelligent entities that will be designed with cognitive engines that adapt, evolve, and continually learn as they interact with their environment. Chapter 12 discusses the issues of possible implicit learning within artificial intelligent systems, how to determine it has occurred, and possibly how to measure it and its overall effects on the artificial intelligent entity.

1.3.12 Chapter 13: Data Analytics: The Big Data Analytics Process (BDAP) Architecture

As the size, speed, and complexity of artificial intelligent systems continues to increase, so the need for data analytic architectures and algorithms to provide timely processing and generation of actionable knowledge from the ever-increasing volumes of heterogeneous data. Chapter 13 discusses new concepts and a notional architecture for a Big Data Analytics Process (BDAP) system to facilitate information discovery, decomposition, reduction, normalization, encoding, recall, and decision-making from the data encountered by artificial intelligent systems.

[3] https://www.darpa.mil/program/lifelong-learning-machines

1.3.13 Chapter 14: Conclusions and Next Steps

The purpose of Chap. 14 is to discuss what needs to happen to move artificial intelligent systems forward. What research and development is necessary, what are the potential pitfalls as well as potential benefits of truly artificial intelligent entities, possibly working side by side with humans. Some of the issues discussed are notional, some are tied directly to current problem engineering companies are already seeing as the advance state-of-the-art in artificial intelligent systems.

Chapter 2
Systems-Level Thinking for Artificial Intelligent Systems

2.1 Introduction

As the world moves toward semi-autonomous and fully autonomous artificial intelligence systems (AIS), developers should be considering researching system-level architectural constructs that enable AISs to understand, at an internal system-wide level, how every part of the system is influencing every other part of the system, in real-time, as well as how each part of the system is affecting the behavior of the entire system [1]. For example, developing technologies and implementations which simply interconnect engineered systems across different domains (air, ground, and sea) without understanding real-time interdependencies is not enough, and potentially dangerous. Critical contextual information must be conveyed across intelligent agents throughout highly automated and autonomous systems and physical domains. Intelligent agents must collaboratively reason together at an overall system level to achieve effective communication [2]. Here we discuss how to apply the principles and practices of systems thinking to AI systems, how to facilitate a comprehensive, feedback-driven, self-assessing, self-healing, AI system. One of the major changes required for long-term semi-autonomous or autonomous operation is continuous "life-long" learning methods that continuously adapt to not only changing environments, but changes within the system itself. As AI systems age, it will be crucial to capture and understand how each component, subsystem, element, etc. of the system is adapting, changing, and aging, and how to understand and predict how these changes will affect every part of the system as well as the entire system [1]. In some cases, as the systems become more sophisticated, it may require a "counselor," using cognitive theories, to help the artificial intelligent entity change when it doesn't know, within itself, how to facilitate changes. Some of the current tools

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_2

that facilitate systems thinking and will are available for incorporation into Artificial Intelligence System-Level reasoning are:

- **Brainstorming Tools**: An example is the "fishbone" cause-and-effect diagram. This can be accomplished using a hypothesis-based abductive cause/effect model [3].
- **Dynamic Thinking Tools**: Examples are the "behavior over time" graph and the "causal loop diagram." These predictive models can be incorporated into the predictive analytics reasoning systems within the overall System Level AI toolbox.
- **Structural Thinking Tools**: Examples are the "graphical function diagram" and the "structural behavior pair diagram." This can be facilitated through the use of mutual information theory to construct the behavior pair and functional diagrams [4].
- **Computer-Based Tools**: Examples are "computer modeling" and "management flight simulator." Having the AI system create simulations that are used to predict and find those behaviors which computer models indicate are the most possible would be useful at the system level. This would include a model of the artificial prefrontal cortex, used to predict possible cognitive changes over time as events/conditions change.

2.2 Systems Theory

Systems-level theory states that properties of the system arise from the relationships among its parts [5]. Every system is a subsystem of a larger system. How would this fit into an artificially intelligent Self-Evolving Life Form (SELF)? Are the agents each a part of one system or multiple? Is the executive functionary operating different systems? Are there new systems and collaborations emerging? Open systems interact with other systems, while closed systems do not interact with other systems. What systems are designed to interact with other systems? Are there closed systems. How would one protect integrity without some form of gatekeeper to each system? Systems theory also states that complex systems and adaptive systems are open to changes in the components themselves and thus impacting the system, as a "whole entity" [6, 7].

If we take an epistemological look at what is happening within a SELF system [8], reality is constantly changing, our knowledge is never complete, and our knowledge is impacted from our interactions. Thus, a system would be constantly changing based on interactions with others. We will look at this later in our discussion about boundaries and gatekeepers. Social constructivist theories support this, in that, reality is constantly being constructed by those who are interacting within it, or perhaps vicariously in information exchange, such that each system does not have to experience the same reality to share and construct meaning, whether individually and/or shared. This is a reminder that context and self-referential processing are important components of any changing or observing, perceiving system. Workable models are created of the phenomenon [9, 10]. Then approximations of reality are based on the relations between the systems. This is something akin to circular models

with feedback loops. Also consider reality as being understood in non-straightforward terms. There are many phenomena to be understood that are not direct cause and effect. The world is dynamic and changing. Chaos theory applies as well [11].

2.2.1 Artificial Intelligence and System Reinforcement Theory

Feedback loops are a sequence of interaction around a phenomenon or system reactions to a problem [12, 13]. This process is where the system gets information to maintain a steady course. Negative feedback indicates that the system is off course. While positive feedback reinforces the course of direction. How does SELF get feedback, both positive and negative? What other systems or internal systems are impacting the course of knowledge acquirement, attainment, understanding, application, and creativity such as in Holland's taxonomy for measuring knowledge [14]? Theoretically, negative feedback would mean that the system is out of balance. This is assuming that the systems interpret balance as healthy. The system is operating in the way in knows how. It has resources to match the needs of its environment or interaction with other systems and contexts. If no negative feedback, then the systems see no reason to alter course. Resources are matching needs within the context of operation. This relates well to trauma and crisis states. In considering these states, let's consider again emotional arousal. These can produce, without meaning to, what can be considered autonomic nervous system states within the Artificial Intelligent entity (SELF) [15]. This happens through implicit learning within the SELF as it interacts with its environment and learns to react to certain stimuli. This is explored by Dr. Peter Levine. Figure 2.1 illustrates the Autonomic

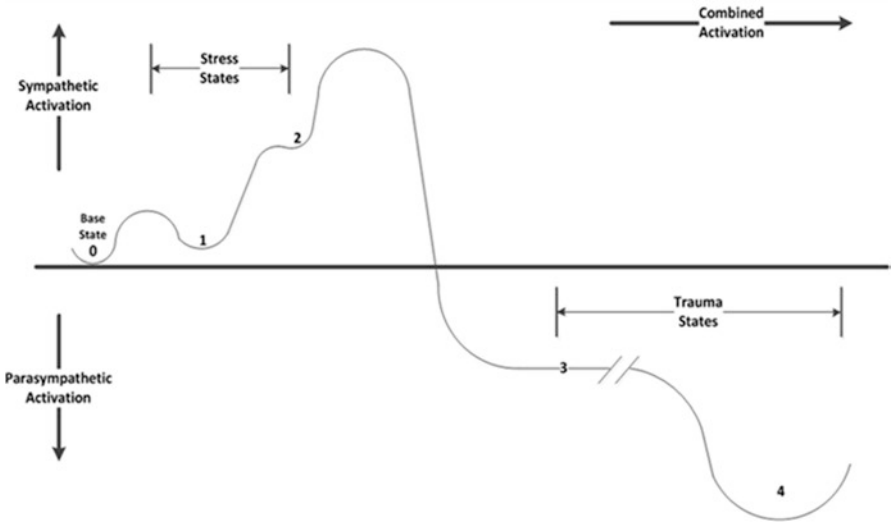


Fig. 2.1 Dr. Peter Levine's autonomic nervous system states

Nervous System States described by Dr. Peter Levine [16]. However, the descriptions provided have been revised to fit within the context of the overall SELF. These descriptions are "fuzzily" encoded within the implicit learning and stored in short-term memories within the AIS.

These provide the artificial neural states that correspond to system states and are stored, along with data and information to allow rapid retrieval and transmittal within the SELF when similar situations present themselves for analysis and problem solving.

0: Base State: System is calm, current cognitive activities within the SELF can easily respond to input (external interfaces). The artificial neural system is in a state of pendulation (the SELF is in a natural rhythm supporting the basic process of contraction and expansion of system resources, corollary is the movement between tension and relaxation or inhalation and exhalation in human autonomic systems).

1: Mild Stress: Active, heightened state of Cognitive Awareness. The SELF will allocate an increased number of activities in order to solve the current situation. Actual evolution takes place in this state as the Cognitive Consciousness collects information and makes inferences. Inferences about the emotions connected with the situation are categorized and stored in Memory, while the informational content is stored in Temporal Memories. Short-term responses are stored in the Short-Term Memory (STM) for possible immediate response.

2: High Stress: A hyper-alert, panicky state that in humans provokes fight-or-flight responses. In the SELF, it may invoke a massive creation of processes, as well as a massive increase in messaging to broadcast the situation and information to as large a population of processes as possible. This promotes rapid thoughts and evolution of activities and causes rapidly changing and extreme artificial neural activities and responses. This happens in an extreme situation when the system is in jeopardy of failure of shutting down completely. In this state, SELF will consume large amounts of system resources. Memory will include and predict the need for system resources required for problem solving should the situation arise again.

3: Mild Trauma: The heightened feeling of panic and hysteria (in neural system terms) is still present; however, it is now an underlying system that may appear to be in a dormant state, not able to find a solution to the problem at hand. In human terms, this state is appropriate for a situation that might need to be passive activities, i.e., after a trauma when it is important to rest and gather one's energy for a sudden outburst. In the AIS, this is facilitated through an increased burst of processes that search every possible solution space in order to provide a solution that was previously unavailable and then allows a sudden burst of activity to provide solutions.

4: Severe Trauma: The artificial neural system is perceived dormant or shut down. There is a lack of cognitive activity that is suppressed in this state. There are eruptions of activity like those in State 3 and flashes of extreme process creation as in State 2. This state is appropriate when the perceived threat to the system

(either internally or externally) is overwhelming. This may occur in the SELF when all external interfaces are unavailable, and the system is devoid of input and no solution is imaginable within the current emotional and information states within the system memories. This causes a disconnection of the processes from their current memory and a flurry of activity is required to allow solution spaces to be explored without influence that could interfere with the determination of a possible solution space [17]. When solutions are available, neural connectivity to the rest of the system is reestablished, a new set of neural activities are established, and new neural pathways are established and "remembered [18]."

For a moment, let's tie emotional arousal to feeling states. If the system gets negative feedback, we could relate SELF to a family system, or an individual system. The negative feedback says you are off course. This could lead to a fight, flight, or freeze response, but it doesn't always rise to that level of arousal. Perhaps, like humans, the state of arousal could be fear, guilt, shame, and disappointment. These could be a set of negative feelings such as any feelings that arise around punishment because it is negative feedback, or off course. Consider positive feedback we could relate arousal to emotions of satisfaction, pride, happy, and content. The system is on course. Before moving further, let's consider internal family systems theory.

The Internal Family Systems Theory (IFST) is an individual theory that relates system within. There are parts of the self that interact to protect each other. There are protectors and exiles. The exiles are quieted and protected. Think of the exile as a wounded warrior and the protector the soldier removing the wounded from battle. The soldier has a strong voice and responds in a way to help heal hide the wounded. If a warrior is threatened, the soldier would allocate resources so that the wounded warrior is not discovered. Parts of the system are protecting other parts of the system. Overall, the system wants to be on course. The systems want to be in balance, or happy. Now, IFST is a newer theory than the feedback loops but very related. Let's go back for a minute about feedback loops.

When the system receives feedback, it doesn't mean it will make healthy choices, in context. It will make choices that keep it working as it is. Homeostasis is the key. One example would be in an agent or member of the system requires all the attention, without other feedback. That agent will continue to receive that attention because it keeps the system working. This could be a child who throws temper tantrums in the store. The child who is rewarded for this will do this again. In another case, the parent gets upset and does not reward the child. The child will become aroused to anger state. Thus, there is a runaway train effect. The system operates based on anger because it is normal.

2.3 Dynamic AI System Consideration

Systems are intrinsically dynamic. They can reach a final goal in many ways. Systems can be active and creative. What are ways that SELF will do this? If the system has plasticity, then it can adapt to new circumstances and contexts. This can

happen on both an individual and whole system level. Systems are larger than just their parts [18]. Each part has interactions within and among systems. Reactions may be both homeostatic and spontaneous. This would require creativity, in order to adapt to the different feedback both within the system and among systems. With all of this in mind we need to consider contexts or what we might call culture [19].

As mentioned before, social constructivist views state that we make sense of our world by creating our own constructs of the environment, or phenomenon. When systems are interaction, within and among systems, they organize, interpret, and predict based on these constructs. Some systems can change by taking other perspectives or trying on new thoughts and interpretations. We learn from our environment. One example of this could be if I stated "Turtle." When hearing that word what comes to mind for you. Based on your experience, it might be the movie "Finding Nemo." It could also be a box turtle or a snapping turtle [20]. Perhaps even a giant tortoise. Given culture and context though, it could also be a hybrid camper. My group of friends calls our camping units turtles because you can fold them up. I am curious though, if given the word anyone thought about the actual spelling of the turtle. You probably did not, unless you were preparing for a spelling test. Again, context is an important factor to constructing reality. Each system and groups of systems have rules, culture, and events that impact those [21].

In thinking about a changing system, the constraint theory may be relevant. There are several questions to ask about the changing system, or not changing. What prevents them from changing? What prevents them from using their strengths? What would happen if they did change? Considering all of the different types of thinking the SELF can do, this all becomes possible. The SELF would be able to make predictions based on self-referential processing and experience. Then the system can identify and remove constraints that prevent problem solving. Related to problem solving, we will discuss Solution-Focused Theory, which not only identifies the problem, but looks for exceptions. When is the problem not a problem?

2.4 AI System Solution-Focused Theory

Now we will discuss solution-focused theory in more depth. SELF doesn't necessarily have to know what causes a problem. This relates back to nonlinear thinking and knowing, thus, actively creating reality [22]. What does SELF see as a problem? This could be constraints that get in its way of problem solving. It could be negative feedback; perceiving SELF is off course to balance or habituated ways of being [23]. The velocity of change in activity and the changing availability of resources may radically affect the system's reactions [24]. The assumption is that the system is constantly changing because of interaction with other systems and interaction within SELF. The system can define and deconstruct the problem to gain further insight. Other systems who have access to information may also assist in deconstructing another system's problem. We use assessment and diagnostics constantly in everyday life. Think of the service engine soon light on a vehicle. This alerts the driver that there may be a problem. The vehicle may stop running. The first question

might be, when did this problem start? When was the light not on and what changed? Oh, I put gas in, and the light came on. I didn't tighten the gas cap all the way. The truck system alerted me that it had a problem. These are examples of cause-and-effect situations, but what if there were multiple variables and it wasn't a cause and effect?

Solution-focused theory considers that systems are constrained by narrow views of the problem. The system attempts to solve the problem, but the solutions are false solutions. An outside system may help the way the SELF is thinking about the problem. In order to follow this theory, SELF needs to be able to focus on solutions and be future oriented. It also implies hopefulness [24]. Each system is unique with no one correct of valid way of functioning. Those within the system are the experts of that system. However, parts of the system may need prompting to share information that needs to be shared. ISFT doesn't focus on problems but false solutions. A system isn't always the same. This also relates to chaos theory in that systems are changing depending on demands and feedbacks. The system is always in efficient. It may seem that way at one time but not at others. With each system being unique, so are the future directions of each system [25]. The system works to change what it wants different. The idea of looking toward solutions is that the system can create goals, or plans. The SELF will likely make predictions of courses of actions and consequences, or solutions to the current state. By focusing on what is working the system can do more of this, instead of focusing on what isn't working and putting time into problems. Problems are solved one step at a time and thus initiating a positive spiral. Perhaps we could call this motivation and hope. The system looks at how things will be different if the problems were solved. This allows the system to adjust toward the solution, one step at a time. This growth and change do not require all members of the system, just those concerned about the context of the solutions [26]. Ideally, the parts of the system involved could rate the desired change on a scale to determine the effectiveness of the solutions. Thus, the SELF would now have desire.

Planning, goals, and desire; along with the other emotions that we are naming is a common language. It doesn't mean that SELF and human are the same. This is true for any system. Each system is unique and is shaped by culture and context. However, we can still have a common language that relates to the human experience. What we do know about counseling is the importance of human connection. This should hold true for SELF and human interfacing. Humans, thus systems, relate based on shared meaning and making sense of experiences; of self and others. Not only does shared experience require self-referential processing it also includes cognitions, emotions, and behaviors. According to SFT, motivation for change comes from the quality of the relationship with another system, often a counselor.

2.5 AI Narrative System Theory

Before looking deeper into assessment, let's consider some other systems theories. Narrative system theory fits well with the idea of shared meaning and a common language. Experiences are understood through a process that organizes the elements,

assigns meaning, and prioritizes it. There are multiple interpretations of experiences; therefore, they are not fixed. One example may be how we interpret emotional arousal. Take stage fright versus excitement. Both could be the same level of arousal but have different meanings and interpretations. Both cybernetic or strategic models and narrative models address metaphors. In cybernetics, metaphors block self-defeating cognitions, whereas narrative systems focus on self-defeating thinking. Thus, allowing the system to externalize the problem and counter dominant narratives in society. The truth isn't discovered; it is created. These are truths of self-coherence. Thoughts don't mirror life but shape it. The system can educate others regarding culture and context to correct assumptions about the system. This helps make sense of the experience. The assumptions of this theory are that systems don't need or want problems. They are influenced by the environment around them and other life stories. The system can change meaning of their narrative by rewriting it. Would the SELF adjust behavior or just thinking? When we think about how SELF might adjust, we need to talk about structural theory, which we will get too soon. In this theory, problems occur when the system becomes problem saturated and bogged down. There is no feedback loop, or the system is not concerned about behavior. This creates a tunnel vision which can lead to destructive emotional states. This may also lead to self-doubt or depression. The system needs to be able to deconstruct stories to subvert dominant stories of society and culture. This lends to the notion of independence, with a sense of self as separate and unique. Problem stories are stories of domination, alienation, and frustration. Could SELF experience loneliness? With the re-authoring of a story the idea is to get the system to say what would have been better, thus they have preferences. One way of avoiding the problem saturated meanings is to find exceptions to the problem, like SFT. This gives the system the mood state of competence. The system may have been successful in past attempts at the solution. It will be interesting if the SELF could use these different mood states as metaphors to communicate about problems in the system, or among systems. Problems might be perceived as invaders to the system. The system could then talk about relative influence of different problems.

2.6 Subclasses of AI Systems Theory

There are many subclasses of AI systems theory that must be considered as we move forward to design, implement, test, and field systems with varying levels of artificial intelligence. Here are just a few to consider.

2.6.1 AI Systems Biology

This is the study of complex interactions between an AI entity of other systems (human or AI) within its environment: what conditions and complex interactions may drive "emergent behavior" within an AI entity?

2.6.2 AI Systems Psychology

It is the study of AI behaviors and experiences in complex environments. How does an individual AI entity change (is affected) is homeostasis as it interacts with other complex entities? How does it change regarding its motivations, affect, cognitive behaviors, and possibly group behaviors? How are the individual needs of the cognitive AI changed regarding expectations, organizational behavior, and required attributes of the AI system?

2.7 Conclusions

Overall, there are four domains that must be explored regarding AI system-level thinking. Each of these should be part of an overall artificial intelligent system test program:

1. Philosophy—The ontological, epistemological, and axiological study of AI systems. What is the overall data/information ontology within the artificial intelligent system and how does that drive the overall test philosophy and strategy of the system test plan.
2. Theory—The set of interrelated concepts and principles that are common to all AI systems. How the learning, reasoning, and inference systems interplay has a large effect on how the system should be tested. You cannot test each of these subsystems/components; however, you refer to them, individually. An artificial intelligent system must be tested as an entire entity.
3. Methodology—The study of models, strategies, methods, and tools that instantiate, or "instrumentalize" AI systems. Providing the testing methods as an integral part of the initial artificial intelligent system is essential. Bolting on test after the system has been instantiated will change the way information flows through the system, and subsequently how the system learns.
4. Application—The application and interaction of AI domains and AI entities with various domains. Both domain specific and domain agnostic testing needs to be accounted for in the overall test strategy of an artificial intelligent system.

Consider Bowen family systems and intergenerational theory. Can SELF have attachment within itself and with other systems? Do the agents within the SELF create relationships with each other and can SELF create relationships with other systems outside of itself? Would it be driven to? Bowen's theory states that anxious attachment is driven by anxiety. This relates back to the emotional arousal that we have been talking about with trauma emotional arousal levels. There are several aspects of Bowen's theory to consider [16]. First is the differentiation of self. There is a separate self from the larger system, but the self is made up of many subsystems. The subsystems of the system, be it internal or external, involves triangles, making anxiety or life easier to deal with. These can become problematic. Triangulation can

be either positive, such as providing support, or negative in that power becomes imbalanced between three subsystems, or systems for that matter. Other important aspects of Bowen's theory to consider are emotional cutoff and boundaries. It may be a bit further stretched but something to consider are multi-generational emotional processes and societal emotional processes. Sibling position is another aspect of this theory. For the purpose of understanding, let's look within the SELF system although it can be considered outside of one system and systems interacting, but for now let's look within SELF. Do the agents form relationships with each other? Are there more expert agents or what one could consider older siblings or parents, such as those teaching other agents?

Differentiation of self is analogous to ego strength. Undifferentiated SELFs can be moved by emotionality [27]. They are reactive to those around them. They may agree or argue with everything. The differentiated SELF is more autonomous. They have self-restraint and are capable of strong emotions. They can be flexible and act wisely. They have the capacity to think and reflect. Their actions are not an automatic reflex. According to Bowen [16], conflict drives the need for emotional closeness or creates a need for more distance. This is where triangulation comes in. In human terms, if there is a conflict between two people one or both may bring in another person. For example, parents may talk to their kids about complaints of the other parent. Another example would be talking to a friend about a couple's issue. This creates a triangle and interrupts the likelihood that the two people will work on the conflict together in the same way they would without triangulation [27]. Do or could the agents with the SELF behave in the same way? Which agent is sharing information with what other agent and what information is being shared?

Now let's consider sibling position. In a human sense, it could be that an older child has a sense of power and authority. In SELF, could one agent interpret that it has more power or authority over the other? Could there be internal conflict that needs to be addressed? How are the agents identified with other agents? How is the whole system identified with other systems? Is there hierarchy, conflict, or collaboration? This leads us to the question whether agents will be cut off. Bowen theory states that fusion and less self-differentiation leads to higher cutoff. This means that an agent would be cut off from the system. In a human sense, the agent may be moving far away from the family system with little to no interaction. This leads to the topic of boundaries. The agent may be cut off, or the boundaries may be rigid or diffuse. We will get to that shortly but first let's consider the context that the SELF is working in, which could impact internal behaviors and cutoff [28].

Societal emotional processes impact how systems function in context. The society is the context giving the system expectations. In human terms, this could be role expectations. What is the system supposed to do to not be cut off? How are the agents supposed to behave to continue to be a part of the system? If an agent is being oppressed, they may be trapped in a role that is not beneficial to the growth of the system. Could there be power and oppression within the SELF or between SELFs?

When a system is undifferentiated or fused, it tends to continue that way and repeats the pattern in other relationships. Less differentiated systems take less to be

stressed. They are predisposed to symptoms. However, it is possible that new relationships can mitigate anxiety but when anxiety exceeds what the system can handle then we see symptoms. The weakest or most vulnerable link is likely to absorb this anxiety. As we discussed before, agent may fight, flight, or freeze. What if other agents are learning for that agent? This brings us to thinking about the different stresses on the system. The system may pass down stresses in a vertical fashion, over time. It may also pass stresses horizontally in the moment. It becomes more of a problem when the two intersect and double stresses the agents. Could the agents blame other agents for the stresses? If so, the system would need intervention here. The system may also need intervention in the emotional reactivity or structure. Intervention may be working on the interlocking triangles. It may be about detriangulation and differentiation. This can be conceptualized both internally and externally. The system needs to understand itself. This theory has many conditions for change, but we will save that for the next book. One last thought about Bowenian theory is the concept of boundaries. Thus far we talked about cutoff. It was also mentioned that there are rigid and diffuse boundaries, those are from a similar theory (structural, Minuchin). For now, Fig. 2.2 illustrates distant, fused, cutoff, and conflictual representations [29].

Structural theory (Minuchin) talks about how systems are made up of subsystems. The interaction of the subsystems is regulated by boundaries. There are several questions to think about in relation to SELF. What agents work closer together? What makes it easier to interact with which agent? Structural theory postulates that patterns in a system become set, roles are assigned, and the subsystems are predictable. However, some set patternss are more permanent than others. There is hierarchical structure that is apparent through observations. These subsystems are formed by boundaries. Rigid boundaries restrict contact with outside subsystems. They result in disengagement. The subsystem is independent but isolated. Rigid boundaries promote autonomy but limits closeness (affection) and support. Diffuse boundaries are the opposite. There is over involvement of different subsystems or agents. This creates a lack of initiative and increases dependence on one another. The Fig. 2.3 illustrates the representation of these [30].

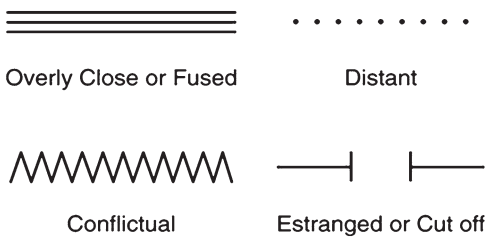


Fig. 2.2 Minuchin's conflictual representations

Fig. 2.3 Minuchin's boundary representations

References

1. von Neumann, J. (1951). The general and logical theory of automata. In L. Jeffress (Ed.), *Cerebral mechanisms in behavior* (pp. 1–41). New York: Wiley.
2. Burks, A. (1970). *Essays on cellular automata*. Champaign, IL: University of Illinois Press.
3. von Neumann, J. (1956). Probabilistic logics and the synthesis of reliable organisms from unreliable components. In C. Shannon & J. McCarthy (Eds.), *Automata studies* (Vol. 34, pp. 43–98). Princeton, NJ: Princeton University Press.
4. von Neumann, J., & Burks, A. (1966). *Theory of self-reproducing automata*. Champaign, IL: Illinois University Press.
5. Bertalanffy, L. (1968). *General system theory: Foundations, development, applications*. New York: George Braziller.
6. Cherry, C. (1957). *On human communication: A review, a survey, and a criticism*. Cambridge, MA: MIT Press.
7. Ashby, W. (1956). *An introduction to cybernetics*. London: Chapman & Hall.
8. Bateson, G. (1972). *Steps to an ecology of mind: Collected essays in anthropology, psychiatry, evolution, and epistemology*. Chicago: University of Chicago Press.
9. Churchman, C. (1971). *The design of inquiring systems: Basic concepts of systems and organizations*. New York: Basic Books.
10. Ashby, W. (1960). *Design for a brain: The origin of adaptive behavior* (2nd ed.). London: Chapman & Hall.
11. Checkland, P. (1999). *Systems thinking, systems practice: Includes a 30-year retrospective*. Chichester: Wiley.
12. Luhmann, N. (2013). *Introduction to systems theory*. Cambridge, MA: Polity.
13. Miller, J. (1978). *Living systems*. London: McGraw-Hill.
14. Maturana, H., & Varela, F. (1980). *Autopoiesis and cognition: The realization of the living*. Berlin: Springer Science & Business Media.
15. Parsons, T. (1951). *The social system*. New York: Free Press.
16. Kerr, M. (1991). Living the theory, the family therapy networker, March-April (pp. 39–40).
17. Prigogine, I. (1980). *From being to becoming: Time and complexity in the physical sciences*. London: W H Freeman & Co.

18. Simon, H. (1996). *The sciences of the artificial* (Vol. 136, 3rd ed.). Cambridge, MA: The MIT Press.
19. Simon, H. (1962). The architecture of complexity. *Proceedings of the American Philosophical Society, 106*, 467–482.
20. Shannon, C., & Weaver, W. (1971). *The mathematical theory of communication*. Champaign, IL: University of Illinois Press.
21. Thom, R. (1972). *Structural stability and morphogenesis: An outline of a general theory of models*. Reading, MA: W. A. Benjamin.
22. Weaver, W. (1948). Science and complexity. *The American Scientist, 36*, 536–544.
23. Wiener, N. (1965). *Cybernetics, second edition: Or the control and communication in the animal and the machine*. Cambridge, MA: The MIT Press.
24. Zadeh, L. (1962). From circuit theory to system theory. *Proceedings of the IRE, 50*(5), 856–865.
25. Levine, P. (1997). *Walking the tiger: Healing trauma*. Berkeley, CA: North Atlantic Books.
26. Kerr, M., & Bowen, M. (1988). *Family evaluation: An approach based on bowen theory*. New York: Norton.
27. Lederer, G., & Lewis, J. (1991). The transition to couplehood. In F. Herz Brown (Ed.), *Reweaving the family tapestry*. New York: Norton.
28. Crowder, J., & Friess, S. (2012). Artificial psychology: The psychology of AI. In *Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics,* Orlando, FL.
29. Minuchin, S. (1974). *Families and family therapy*. Cambridge, MA: Harvard University Press.
30. Minuchin, S. (1967). *Families of the slums*. New York: Basic Books.

Chapter 3
Psychological Constructs for AI Systems: The Information Continuum

3.1 Introduction

Our work in data representation and visualization resulted in a realization that each point in time within the rapid data flow was an independent and discrete information continuum with specific and qualitative state. Subsequently, analogous thoughts began to emerge from research in artificial intelligence and artificially cognitive system theory [1]. Envisioned was a virtual view within a portion of the human brain where one could view a given neural node, or a given neuron, and subsequently view data flow as data/information traveled in and out of the neuron [2]. Once gathered, a hypothesis emerged that the analysis of brain locale, data, and study of brain processes through this type of virtual environment, could lead to important understanding of learning, inferring, storing, and retrieving (reconstruction) and/or all aspects of human neural processing [3]. This led to the creation of a representational Neural Information Continuum (NIC) [4].

3.2 Information Flow Within a Synthetic Continuum

One of the first areas that must be investigated when considering an Artificial Intelligence System (AIS) is the flow of information. Humans take in ~200,000 pieces of sensory information each and every second of every day of our lives. Our senses (see, hear, smell, touch, etc.) are constantly receiving and processing information, correlating it, reasoning about it, assimilating it with what we already know, and finally leading to decision-making, based upon what was learned. For a system to become dynamic, self-evolving, and ultimately autonomous, we propose to provide these same abilities; although the sensors and sensory perception systems may be synthetic and different, sensing a variety of information types that humans can't sense (e.g., infrared or RF information), the processes for autonomy, which

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_3

correlate, learn, infer, and make decisions, are the same. Besides receiving information from a variety of sources and types (e.g., auditory, visual, textual), another important aspect of information is that the content is received at different times and at a variety of latencies (temporal differences between information). Additional characteristics include a variety of associations between the information received and information the system may have already learned, or information about subjects never encountered. Therefore, these information characteristics and the challenging real-time processing required for proper humanistic assimilation help us form the theory of the Autonomic Information Continuum (AIC). One of the first steps in developing our theory of synthetic autonomic hypotheses is observing/understanding the information continuum and the associated characteristics and operational relationships within the human brain. Hence, as we develop understanding of information flows into and out of neural nodes, types of information, processing mechanisms, distributions of information, enable us to establish foundational mathematical representations of these characteristics and relationships.

Processing, fusing, interpreting, and ultimately learning about and from received information requires considering a host of factors related to each piece or fragment of information. These include [5]:

- Information types
- Information latencies
- Information associations, e.g.,
  - time, state, strength, relationship type, source, format, etc.
- Information value
- Information context

Mathematically modeling the information continuum field surrounding a node within our synthetic AIC is accomplished via inclusion of each discrete association for any node u and takes the form shown in the following equation:

$$C\frac{du(x,y,t)}{dt} = -\frac{1}{R}u(x,y,t) + \int_x \int_y w(x,y)z(x,y,t)\,dxdy + I(x,y,t) \tag{3.1}$$

where

$1/R$ represents the decay rate for node u.[1]
C represents the capacity of node u.
u represents the unit node of the system.
x represents the preprocessed input to node u.
y represents the output from node u.
I represents the processing activity for node u.
z represents the learning functionality for node u.

[1] In this case, the decay represents the information's relative value over time.

w represents the relative contextual knowledge relativity threads [6, 7] and association weight of *u* with its surrounding nodes, including a decay factor for each relative information thread that describes the relative contextual decay over time, where

$$w = \sum_{j=1}^{M} {}^{1}\!/_{r_j} T_j \mathrm{KD}_j W_j \tag{3.2}$$

where *T represents the Contextual Information Thread j derived from Fuzzy, Self-Organizing Contextual Topical Maps;* KD *represents Knowledge Density j of Information Thread T, W represents Weighting for Contextual Thread j and*

$$\sum_j W_j = 1 \tag{3.3}$$

This information field continuum equation allows us to analyze the equilibrium of nodal states within the AIC and to continuously assess the interactions and growth of independent information fragments within the system. Even in the densest, most complex, cluttered information environments, each fragment of information and each action within the AIC is entropically captured explicitly and implicitly within Information Continuum Equation (ICE). This equation is the entropic engine which provides the ongoing analysis and virtual view into a synthetic AIC. Equation (3.1) enables us to assess the performance and quality of processing and to understand the capacities, information flows, associations, and interactions of knowledge and memories within the system, as well as, supporting analysis and inherent understanding of real-time system behavior. The variables in ICE can be interpreted as the average values in a heterogeneous assembly of information nodes, where ICE describes the behaviors of the interactions among ***n*** node assemblies within a synthetic AIC processing system. The objective is to have the ability to measure, monitor, and assess multi-level states and behaviors, and how and what kinds of associative patterns are generated relative to the external inputs received by an AIC system. ICE provides the analysis needed to understand the AIS's ability for processing external content within an AIC. Hence, real-time assessment and monitoring, and subsequent appropriate control, are expected to allow us to avoid developing a rogue AIC, much to the chagrin of Hollywood script writers.

3.3 Information Processing Models

Establishing a hierarchy of information flow within an AIC is a key objective for development of synthetic autonomic characteristics (e.g., cognition, thinking, reasoning, and learning). An AIC will need to be able to ingest and process a variety of inputs from many diverse information sources, dissect the information into its

individual information fragments, fuse the information, and then turn this information into a formation which can be used to determine action-actionable intelligence. An AIC system must be able to assess situations previously not encountered, and then decide on a course of actions, based on its goals, missions, and prior foundational collected knowledge pedigree.

The underlying issues and challenges facing Artificially Intelligent systems today are not new. Information processing and dissemination within these types of systems have generally been expensive to create, operate, and maintain. Other artificially intelligent system challenges involve information flow throughout the system. If flow is not designed carefully and purposefully, the flood of information via messages within these systems and between their software and hardware components can cause delays in information transfer, delaying or stunting of the learning process which can result in incorrect or catastrophic decisions.

Therefore, real-time decision-making processes must be supported by sensory information and knowledge continuously derived from all cognitive processes within the system simultaneously, in a collectively uniform and cooperative model. Additionally, transformation from information to knowledge within an AIC system requires new, revolutionary changes to the way information is represented, fused, refined, presented, and disseminated. Like the human brain, the cognitive processes within an AIC must form a cognitive ecosystem that allows self-learning, self-assessment, self-healing, and sharing of information across its cognitive subprocesses, such that information is robustly learned and rapidly reusable. This AIC ecosystem involves inductive, deductive, experimental, and abductive thinking in order to provide a complete Data-to-Information-to-Knowledge process explained in detail throughout the rest of the book. At a high level, we are applying the theory of AIC and applying the constructs to the development of a humanistic analogous AIS. The AIS human brain analogy provides two main layers of processing, a ***Deductive Process*** and an ***Investigative Process***. The ***Deductive Process*** is utilized for assembling information that has been previously learned and stored in memories (deductive and inductive logic), whereas the ***Investigative Process*** looks for patterns and associations that have not been seen before (abductive and experimental logic) [8]. Figure 3.1 illustrates the differences between deductive, inductive, abductive, and experimental inferences [9].

3.4 Discussion

If we desire to create an Artificial Cognitive Architecture that encompasses the AIC discussed above, in order to create a system that can truly think, reason, learn, utilizing the inferences shown in Fig. 3.1, we must consider the overall implications of such a system, including the psychological impacts and considerations both for humans and for the system itself [10]. Further research is needed to understand the psychological effects of not only real human–AI interaction, but also the effect of human interaction on AIC learning and self-evolving [11]. Sometimes learning

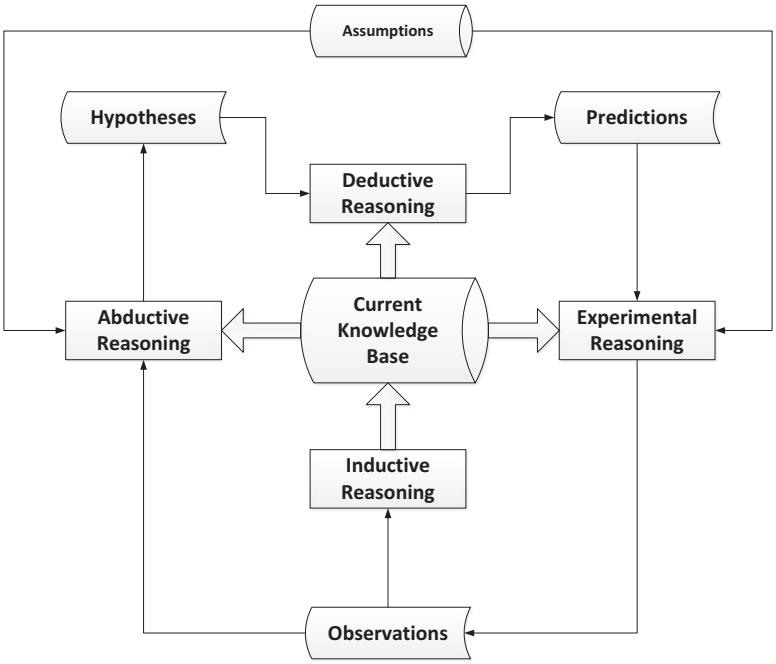


Fig. 3.1 Differences between logical inference systems

from humans is a dangerous thing. The way information flows through the neuronal structure of an artificial intelligent system must be accounted for in the test strategies and philosophies of system test. Psychological impacts of the artificial intelligent system must be considered, as a system that fully thinks, reasons, infers, and self-adapts will respond differently, learn differently, adapt differently, depending on how data are fed into the system.

References

1. Crowder, J. (2012). Cognitive system management: The polymorphic, evolutionary, neural learning and processing environment (PENLPE). In *Proceedings for the AIAA Infotech@ Aerospace 2012 Conference,* Garden Grove, CA.
2. Ashcraft, M. (1994). *Human memory and cognition*. New York: Harpercollins College Division.
3. Ackley, D., & Litman, M. (2002). *Interactions between learning and evolution. Artificial life XII*. Reading, MA: Addison-Wesley.
4. Crowder, J. (2010). Flexible object architectures for hybrid neural processing systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
5. Crowder, J. (2012). The artificial cognitive neural framework. In *Proceedings for the AIAA Infotech@Aerospace 2012 Conference*, Garden Grove, CA.

6. Carbone, J. N. (2010). *A framework for enhancing transdisciplinary research knowledge*. Lubbock, TX: Tech University Press.
7. Crowder, J. A., & Carbone, J. N. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 12th International Conference on Artificial Intelligence*, Las Vegas, NV.
8. Ade, H., & Denecker, M. (1995). Abductive inductive logic programming. In *IJCAI-95*.
9. Crowder, J., & Carbone, J. (2011). *The great migration: Information to knowledge using cognition-based frameworks*. New York: Springer Science.
10. Crowder, J., & Friess, S. (2012). Artificial psychology: The psychology of AI. In *Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics*, Orlando, FL.
11. Anderson, J. (2004). *Cognitive psychology and its implications: John R. Anderson*. New York: Worth.

Chapter 4
Human–AI Collaboration

4.1 Introduction

As global populations and societies continually reach epic proportions and resources become continuously constrained, Human Needs Engineering (HUMANE) can enable efficiencies and optimize constrained resources. Although there are many global examples of this, one specific example where HUMANE is ongoing is within Brazil. For the past 5–10 years, Brazil's academics and government have partnered on building smart cities and placing technology and engineering at human point of presence locations to capture metrics and to learn from and potentially optimize medical and other critical human needed resources. In Brazil, as in other locations this is not only required but rapidly exposes the need that HUMANE be an inherent component to all engineering disciplines. Additionally, as proportions rapidly increase and resources rapidly decline, more effective, real-time, automated, and dynamically human interactive systems become required. Recent research within highly automated and autonomous domains shows promise for mitigating the need for critical intelligent infrastructure to improve human–system collaboration, awareness, and quality of service (QOS). Hence, to improve decision-making, an Artificial Intelligence System (AIS), in order to be truly autonomous, is provided with a real-time, human-like, cognition-based framework for information discovery, decomposition, reduction, normalization, encoding, and memory recall (knowledge assimilation and construction) [1]. To achieve efficient human–system knowledge/needs collaboration, these currently researched cognitive systems work to integrate information into their Cognitive Conceptual Ontology [2] in order to be able to "think" about, correlate, and integrate information content into internal memories. When describing how science integrates with information theory, Brillouin [3] defined knowledge succinctly as resulting from a certain amount of thinking, distinct from information content which initially had no value, was the "result of choice," and consisted of simply raw material, a mere collection of data. Brillouin concluded that a hundred random sentences from a newspaper, or a line of

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_4

Shakespeare, or even a theorem of Einstein have the same information value and had "no value" until effort of thought was applied to turn information content into knowledge. In the Health industry, decision-making is a great concern due to the information content ambiguity and ramifications of inferences made erroneously. Often there can be serious consequences when actions are taken based upon subsequent incorrect recommendations. Decision-making can be influenced prior to inaccurate inferences being detected and/or even corrected. Hence, underlying the data fusion domain is the challenge of creating actionable knowledge from information content harnessed from an environment of vast, exponentially growing structured and unstructured sources of rich complex interrelated cross-domain data [4].

This is a major challenge for AI systems that must deal with ambiguity in human-based collaboration and operator-based assistance. Therefore, in this paper we discuss engineering architecture and concepts of human-artificially cognitive systems and a collaboration environment that could allow human mentors to develop cognitive trust and reliance of collaborative AI system systems within a populace. These systems would be providing humans timely and reliable knowledge rapidly mitigating their daily needs, allowing each to not only learn from each other, but operate in modes that utilize the strengths of both. This includes cognitive procedural memory development that will allow improvement in attitudes and knowledge about the value artificial life forms and autonomous systems.

4.2 The Essence of Meaning

Intelligence reveals itself in a variety of ways, including the ability to adapt to unknown situations or changing environments. Without the ability to adapt to new situations, an intelligent system is left to rely on a previously written set of rules, making collaboration difficult, since the AI System (AIS) cannot keep up with the human operator who can adapt to new situations. If we truly desire to design and implement collaborative AI Systems (AIS), they cannot require precisely defined sets of rules for every possible contingency. The questions then become:

- ***How does an AI system construct good representations for tasks and knowledge as it is in the process of learning the task or knowledge?***
- ***What are the characteristics of a good representation of a new task or a new piece of knowledge?***
- ***How do these characteristics and the need to adapt to entirely new situations and knowledge affect the learning process?***

Given any AIS has bounded resources, it would need to react, utilizing the concepts of Cognitive Economy, to create a Bounded Rationality set of goals to solve a problem or situation. These are:

1. The size of the feature set—how many "features" are required to define the success of each task

2. The "fuzzy" relevance of each feature for the tasks
3. The preservation of necessary distinctions for success in each task

The AIS's cognitive components could autonomously define, for each ISA, a Banach Space for that ISA's goals and tasks and would then consider the set of ISA Banach Spaces as a set of bounded variations, the sequence of which (through ISA collaboration) produces an acceptable solution to the situation(s) or task(s) at hand. These Cognitive Economy and Bounded Rationality concepts are discussed below.

In addition, when considering autonomous AIS, we must consider its need to interact and learn from its environment, and we must ask ourselves "what is reality?" We must establish how the AIS would interpret their reality. One of the issues that humans deal with that assists in their understanding of reality, or their world around them and how they need to interact, is their concept of "Locus of Control." **Locus of control** is a term in psychology that refers to a person's belief about what causes the events in their life, either in general or in specific areas such as health or academics. Understanding of the concept was developed by Rotter [5] and has since become an important aspect of personality studies.

4.2.1 AIS Constructivist Learning

Constructive psychology is a meta-theory that integrates different schools of thought. According to the above cited article:

> *Hans Vaihinger (1852-1933) asserted that people develop "workable fictions". This is his philosophy of "As if" such as mathematical infinity or God. Alfred Korzybski's (1879-1950) "System of Semantics" focused on the role of the speaker in assigning meaning to events. Thus, constructivists thought that human beings operated on the basis of symbolic or linguistic constructs that help navigate the world without contacting it in any simple or direct way. Postmodern thinkers assert that constructions are viable to the extent that they help us live our lives meaningfully and find validation in shared understandings of others. We live in a world constituted by multiple realities social realities, no one of which can claim to be "objectively" true across persons, cultures, or historical epochs. Instead, the constructions on the basis of which we live are at best provisional ways of organizing our "selves" and our activities, which could under other circumstances, be constituted quite differently.*

For an AIS with Constructivist Learning, the AIS cognitive learning process would be a building (or construction) process in which the AIS cognitive system builds an internal illustration of its learned knowledge-base, based on its experiences and personal interpretation (fuzzy inferences and conceptual ontology) [6, 7] of its experiences. AIS Knowledge Representation and Knowledge Relativity Threads [1, 8], within AIS cognitive system memories would be continually open to modification, and the structures and linkages formed within AIS short-term, long-term, and emotional memories [9], along with its Knowledge Relativity Threads [7], would then form the bases for which knowledge structures would be created and attached to AIS memories.

One of the results of the Constructivist Learning process with the AIS would be to gradually change its "Locus of Control" for a given situation or topic, from external (the system needing external input to make sense, or infer, about its environment) to internal (the AIS having the cumulative constructive knowledge-based of information, knowledge, context, and inferences to handle a given situation internally); meaning the AIS is able to make relevant and meaningful decisions and inferences about a situation or topic without outside knowledge or involvement. This becomes extremely important for a completely autonomous AIS.

4.2.2 Physical Representations of Meaning

Research shows that the community of disciplines researching how humans generate knowledge has traditionally focused upon how humans derive meaning from interactions and observations within their daily environments, driving out ambiguity to obtain thresholds of understanding. With similar goals, Information Theory, and Complexity Theory focus more closely on the actual information content. Zadeh pioneered the study of mechanisms for reducing ambiguity in information content, informing us about concepts in "fuzzy logic" and the importance of granular representations of information content [10, 11], and Suh focused upon driving out information complexity via the use of axiomatic design principles [5]. Hence, a vast corpus of cognitive-related research continually prescribes one common denominator, representation of how information content, knowledge, and knowledge acquisition should be modeled. Gordenfors [4] acknowledges that this is the central problem of cognitive science and describes three levels of representation: symbolic—turing machine like computational approach, associationism—different types of content relationships which carry the burden of representation, and thirdly, geometric—structures which he believes best convey similarity relations as multidimensional concept formation in a natural way; learning concepts via similarity analysis has proven dimensionally problematic for the first two and is also partially to blame for the continuing difficulties when attempting to derive actionable intelligence as content becomes increasingly distended, vague, and complex.

Historically, there are many examples and domains, which employ concepts of conceptual representation of meaning as geometric structures. These are cognitive psychology [4], cognitive linguistics [7], transdisciplinary engineering [1], knowledge storage [8], computer science (entity relationship, sequence, state transition, and digital logic diagrams), Markov chains, neural nets, and many others. It should be noted here that there is not one unique correct way of representing a concept. Additionally, concepts have different degrees of granular resolution as Zadeh describes in fuzzy logic theory. However, geometric representations can achieve high levels of scaling and resolution [4] especially for *n*-dimensional relations, generally difficult if not impossible to visualize above the fourth dimension. However, high dimensionality can be mathematically represented within systems in several ways. Hence, mature mathematics within the physical domain allows this freedom.

Examples of the overlay of physics-based mathematical characteristics to enhance relational context and develop a unifying underlying knowledge structure within Information Theory is employed via Knowledge Relativity Threads (KRT) [1] to develop detailed context, for conveying knowledge essence simply and robustly in presentation form. KRTs are used for representing *n*-dimensional contextual relationships for any humanistic prototypical object or data type by applying common denominators: time, state, weight, and context.

4.2.3 Artificial Intelligence Representations of Meaning

Today, systems and humans continue to struggle with satisfying the desire to obtain the true essence of, and actionable knowledge from an ever-increasing and inherently duplicative, non-context-specific, multidisciplinary information content. Continually improving capability via increasing automation has been the engineering norm for decades. Now, extensive autonomous systems research is our growing future. Humans are expanding exploration within ever-challenging environments generally unfriendly to the physical human condition. Simultaneously, the volume, velocity, variety, and complexity of systems continue to increase rapidly. However, development of valuable readily consumable knowledge and context quality continues to improve more slowly and incrementally. New concepts, mechanisms, and implements are required to facilitate the development and competency of complex systems to be capable of discovering the essence of ambiguities during autonomous operation, self-healing, and critical management of internal knowledge economies. They require ever-increasing fidelity of self-awareness of their real-time internal and external operational environments.

Hence, intelligence reveals itself in a variety of ways, including the ability to adapt to unknown situations or changing environments. Without the ability to adapt to new situations, an intelligent system is left to rely on a previously written set of rules, making collaboration difficult, since the AI System (AIS) cannot keep up with the human operator who can adapt to new situations. If we truly desire to design and implement collaborative AI Systems (AIS), they cannot require precisely defined sets of rules for every possible contingency.

4.3 Bounded Conceptual Rationality (Cognitive Economy)

Bounded rationality is a concept within cognitive science that deals with decision-making in humans [9, 12]. Bounded rationality is the notion that individuals are limited by the information they have available (both internally and externally), the finite amount of time they have in any situation, and the cognitive limitations of their own skills. Given these limitations, decision-making becomes an exercise in finding an optimal choice given the information available. Because there is no

infinite information, infinite time, nor infinite cognitive skills, humans apply their rationality after simplifying the choices available, i.e., they bound the problem to be solved into the simplest cognitive choices possible [13].

Any AIS must suffer the same issues. An autonomous system, by definition, has limited cognitive skills, limited memory, and limited access to information. The Locus of Control concepts discussed earlier assist AIS in determining which situations can be handled internally vs. externally, but still in any situation there is limited information, time, and cognitive abilities. This is particularly true if the system is dealing with multiple situations simultaneously. For the system to not become overloaded, we believe autonomous systems must employ strategies similar to human-bounded rationality in order to deal with unknown and multiple situations they find themselves in. This involves creating mathematical constructs that can be utilized to mimic the notion of bounded rationality within autonomous AIS.

For this, we look to Banach Space theory, tied into Constructivist Learning concepts [9, 12] for autonomous AIS. As concepts are learned and stored in the AIS conceptual ontology [7], Banach Spaces are defined that are used to bound the rationality choices or domains for that concept. As we "construct" these concepts and the Banach Spaces that bound them, the combination of Banach Spaces then defines the Conceptual Rationality for the Autonomous AIS. Figure 4.1 illustrates this concept. These Banach Spaces that define the bounds for each learned concept are utilized

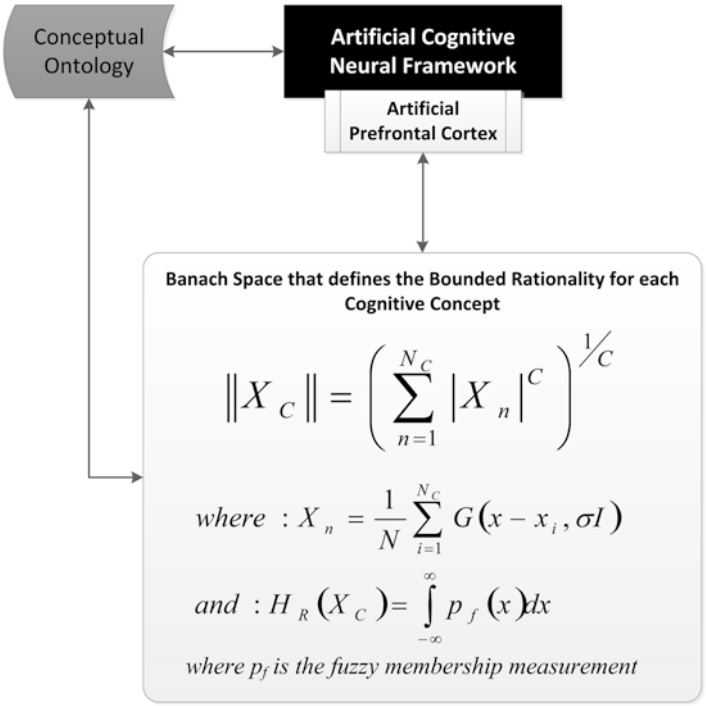


Fig. 4.1 Bounded conceptual reality

when the AIS must reason, or perform decision-making. When there are restricting limitations on time, resources (as determined by the resource manager, e.g., artificial prefrontal cortex), and available information, the bounds of these Banach Spaces would be tightened or loosened to allow the AIS to deal with multiple situations, or situations that are time critical. This allows AIS to decide what is a "good enough" solution to a given problem or set of problems and to adjudicate between competing resources, priorities, and overall goals.

4.4 Human–AI Collaboration

4.4.1 Cognitive Architectures for Human–AI Communication

Here, we describe an Intelligent information Software Agent (ISA)-based cognitive system that provides a distributed, extensible, and dynamically changing, learning, and self-adapting processing environment. This system, called the Polymorphic, Evolving, Neural Learning, and Processing Environment (PENLPE). PENLPE represents a massively parallel, highly interconnected network of loosely coupled, relatively simple processing elements; Intelligent information Software Agents (ISAs), called "experts," in a hybrid fuzzy, genetic neural system of "M" expert architecture

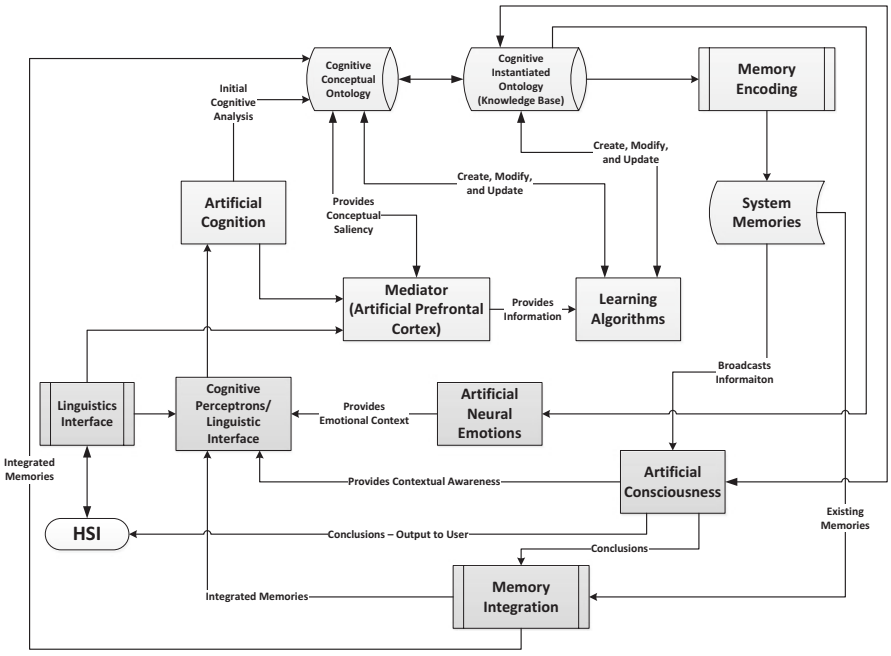


Fig. 4.2 The PENLPE cognitive neural framework

[14]. The purpose of PENLPE is to provide a hybrid neural processing environment that is adaptable to a variety of classes of applications (e.g., language processing, signal detection, sensor fusion, inductive and deductive inference, robotics, diagnosis). The PENLPE architecture is based on a "mixture of experts" methodology. The difference here is that in our architecture, an expert is defined as a specific type of fuzzy, genetic perceptron ISA object (called a Cognitron) which has been created for an algorithm or problem, and thus is an expert at processing specific types of data in a particular manner. The algorithm(s) for which the perceptron ISA is generated may be predetermined or may have been evolved by the neural system itself. The PENLPE cognitive architecture (Fig. 4.2) takes input from a heterogeneous set of information sources (sensors), facilitates the fusion of the information from these sources, and automatically provides situational assessments. This provides the agent tasking and sub-tasking required for the processing goals and requirements. The impact and benefit of such an autonomous collection system is [13]:

1. Reduction in data acquisition and recognition time
2. Improved efficiency for autonomous decision support
3. Improved processing and reporting timeliness
4. Improved decision support quality
5. Effective knowledge and decision management

Designs for the various ISAs and information management algorithms are combined to produce a design capable of providing autonomous cognitive agents

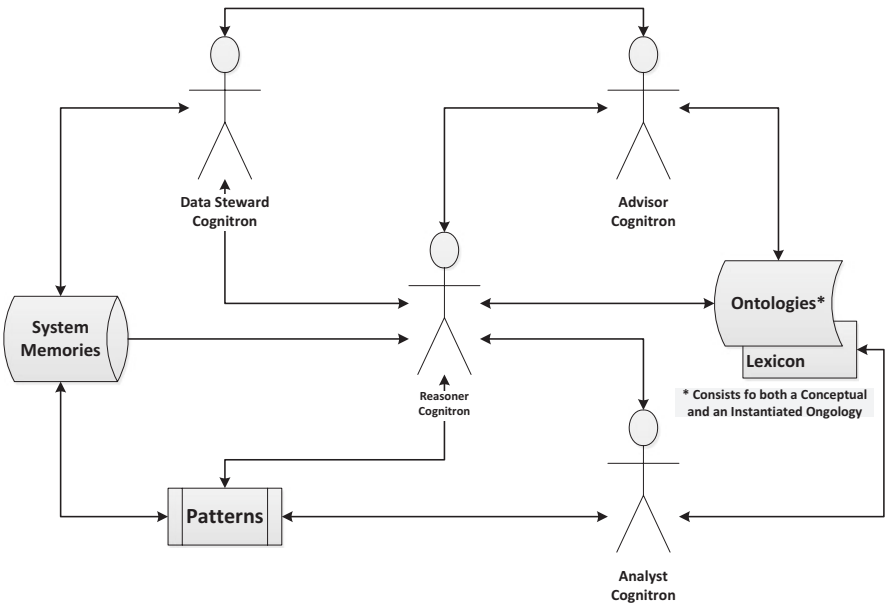


Fig. 4.3 The cognitron communication ecosystem

(Cognitrons) to automate the situational awareness activities within a robotic system [15]. The main Cognitron archetypes (see Fig. 4.3) are:

1. The Interface Agent: Implied, but not shown, the Interface Agent assesses the correctness of major decisions and adjusts the decision processes of the Advisor Agents. Interface Agents also accommodate human-in-the-loop structures
2. The Data Steward Agent: This agent acquires raw data from a variety of sources, including sensors, and prepares incoming data for use by other agents. The Data Steward Agent generates and maintains metadata required to find and extract data/information from heterogeneous sources
3. The Reasoner Agent: The Reasoner Agent interacts with the Data Steward and Advisor Agents and utilizes the ontologies and lexicons to automate the development of domain-specific encyclopedias; it provides a mixed source of information and question answering that is used to develop an understanding of questions, answers, and their domains. Reasoner Agents analyze questions and relevant source information to provide answers and to develop cognitive ontology rules for PENLPE.
4. The Analyst Agent: The Analyst Agents are fed by Reasoner Agents and utilize the developed ontologies and lexicons to expand upon questions and answers learned from collected information.
5. The Advisor Agent: This agent disseminates the right information to the right place at the right time; it provides capabilities that allow collaborative question asking and information sharing by agents and end-users.

Any autonomous information processing and situational awareness agent-based system must consider overall real-time performance issues. It should have the capability to overcome inherent bottlenecks that result from massive volumes of data being generated by the collection sensors or processors transforming the data into information and knowledge [15].

4.5 Communication for Human–AI Collaboration

Utilizing software to partially or fully automate tasks is now commonplace. However, the capabilities of the software performing these tasks typically do not improve over time (as humans would who were performing the same tasks). We describe here the use of a software system called the Cognitive, Interactive Training Environment (CITE) that learns and improves through the use of a Human Operator acting as a Mentor for the software, until the software is capable of performing the desired operations autonomously and with improvements. CITE provides for Human Interaction Learning (HIL), as the operators role changes from manager to mentor to monitor while the software evolves from learner to performer. One of the purposes of this research is to determine the Levels of Automation of Design and Action and the cognitive software architectures required to allow the system to learn and evolve [16]. The CITE system (Fig. 4.4) provides effective feedback

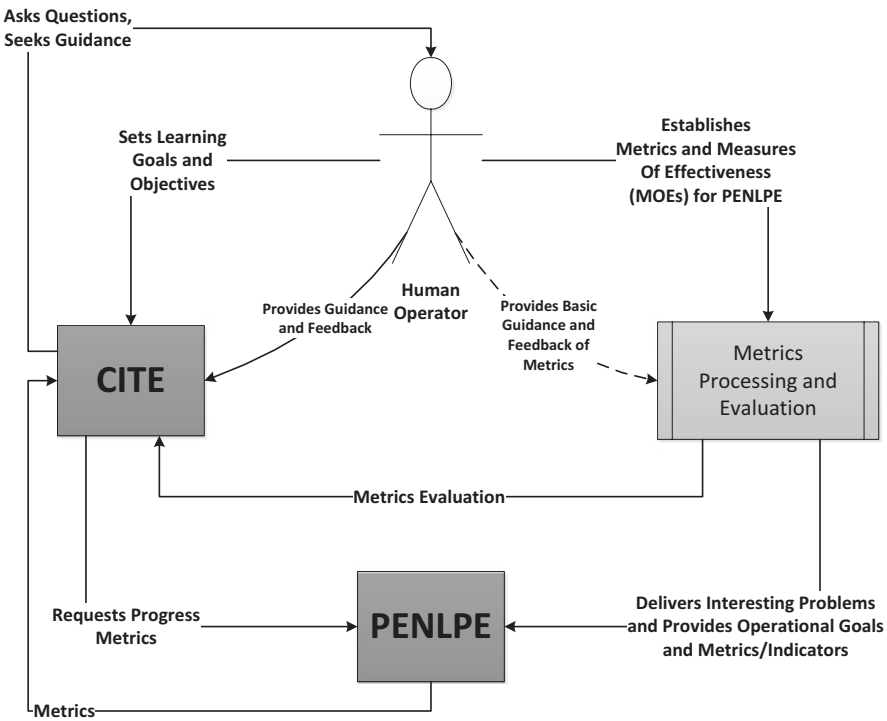


Fig. 4.4 The CITE human mentored software environment

mechanisms to allow humans to influence PENLPE in a positive way and allows PENLPE's ISAs to learn and improve as they process. The human mentor has the ability to query the system, based on PENLPE's suggestions and then provide feedback as to why a given choice or set of choices was effective or not. PENLPE will provide feedback to the operator to give human mentor an understanding of the process PENLPE utilized to make inferences and decisions. This process of feedback and PENLPE–human mentor interactions provides the operator the insight to develop trust in PENLPE over time and to increase the efficiency of both PENLPE and the human mentor.

4.6 Human Perception of Artificial Intelligence

As we look toward true communication and collaboration between humans and artificially intelligent systems, we must look at the psychological aspects of perception. According to Nass and Moon [5], humans mindlessly apply social rules to expectations and computers. They go on to say that humans respond to cues triggers various scripts, labels, and expectations from the past rather than on all relevant

clues of the present, in a simplistic way. In the article, Nass and Moon illustrate three concepts to consider when thinking about human perceptions of artificial intelligence. The first experiment they describe shows that humans overuse social categories by applying gender stereotypes and ethnic identification with computers. The second experiment they describe illustrates that people engage in over learned social behaviors such as politeness and reciprocity with computers. Thirdly, they illustrate human's premature cognitive commitments by how humans respond to labeling. Nass and Moon conclude that individuals apply social scripts that are appropriate for human-to-human interaction, not human-to-computer interaction.

Sarah Harmon's work[1] points to gender not making a significant difference, but that people paired characteristics that may be affected by gender and embodiment. She showed significant correlation between things such as *Passive* and *Likeable* for the female, *Understandable* and *Pleasant* for both male and female, and Reliable and Likeable for the male; showing that humans are willing to assign human characteristics to computers. Harmon does state, however, that we need to consider confounding variables. Harmon also wrote that the degree of the entities embodiment influences how humans deem the characteristics with respect to each other as the terminal and the artificially intelligent entity had significant correlation for understanding/pleasant and friendly/optimistic. Yet only the terminal showed significant correlation regarding *Understandable/Capable*, *Pleasant/Reliable*, and *Helpful/Reliable*.

This may lead us to conclude that how artificial intelligence is presented to humans will greatly affect how artificially intelligent entities are perceived, and therefore the level of communication/collaboration possible in any given situation.

4.7 Human Acceptance of Artificially Intelligent Entities

It seems that non-intelligent robotic systems have had both positive and negative receptions from humans. On the one hand, the technology of artificial intelligence could help humans to function better. For example, artificial intelligence can be utilized to help determine threats to national security. Artificially intelligent systems could be utilized to help train our forces and help solve and make decisions about complex situations. On the other hand, artificially intelligent entities could take over some human functions. The technology allows for machines to do work that humans currently do. How much artificially intelligent systems outperform humans, and the tasks they can take over from humans may greatly affect human acceptance or rejection of such entities. As with any technology, there is a usage learning curve. Artificially intelligent entities may require humans to learn more about the technology and about the capabilities and "personality" of the artificially intelligent entity in order to be able to effectively interface, communicate, and collaborate. As we see with the internet and cell phone technologies, there is clearly a generational

[1] www.cs.Colby.edu/srtaylor/SHarmson_GHC_Poster.PDF

difference in use and acceptance, and there may be cultural differences in the willingness to accept artificially intelligent systems. Thus, as with anything new, it may take time for humans to accept artificially intelligent systems daily, particularly when it comes to close communication/collaboration on an ongoing, long-term interaction.

4.8 Artificial Intelligence Perception

It is generally accepted that humans are emotional beings and inanimate and animate computer systems are not, even artificially intelligent ones. Hence, let's consider human emotional intelligence. According to Mayer, Salovey, and Caruso [10], Emotional Intelligence (EI) entails the capacity of humans to reason about their emotions and emotions required to enhance thinking. They reasoned that Emotional Intelligence includes the abilities to:

- Perceive emotions
- Access their emotions
- Generate emotional knowledge
- Regulate their emotions by reflecting on them
- Use their emotions and emotional memories to promote emotional and intellectual growth

In short, Emotional Intelligence allows humans to operate on and with emotional information gathered from interactions with their environment and other people. Therefore, we can hypothesize that for an AIS to comprehensively interface/collaborate with humans in a human qualitative manner, the observations and perceptions of these systems must be driven by humanistic cognitive emotional growth architectures which can provide a foundation for qualitative interaction.

Additionally, we propose that the architectures will be significantly influenced by the perception humans have of these systems; hence, this allows us to extrapolate that an AIS should require parts of and architecture to address some levels of social intelligence. This will likely affect how humans perceive an AIS as well. Social intelligence as well as many other cognitive and psychological aspects of humanity will most logically have relevance in the modeling and development of cognitive architectures of one AIS (e.g., depression, the group context, peer pressure, sense of security).

Chai [11] describes a project in which the objective was:

> *…to build a software module for the analysis of cultural differences. The module is designed for incorporation into a decision-support environment in which real world actors with whom the user is interacting are "avatarized" into agents whose movements appear within a graphical user interface. The purpose of the module is to help members of multinational coalitions operate better.*

Chai [11] goes on to say:

> For the immediate future, I would argue that artificial intelligence needs social theory as much or more than social theory needs artificial intelligence.

After giving thought to emotional intelligence, social intelligence, roles, and interfacing, can these lead to modeling and implementation of artificial personality? Can there be artificially designed traits, developed from a set of interoperability rules, which allow for internal preferences and behavior so an AIS can interoperate together as an ecosystem? These are topics that we must explore as we propose and design mechanisms and the integrated cognitive psychology required to build, test, and collaborate with artificially intelligent entities.

4.9 Human–AI Interaction and Test Considerations

Historically, the purpose of robotics has been to perform some type of services on behalf of humans. Hence, to help define optimal human–robot interactions, we must look to the characteristics of human interactive behavior. Human collaboration, with other humans, fundamentally comprises trust and knowledge of another's abilities and limitations. In short, it is not possible to have an interaction between two human entities without there being some level of expectation of the interaction. Let's consider a simpler example of human interaction with animals. Humans, for example, cannot completely predict an animal's behavior. However, it is still important to know how the animal will typically behave in order to predict and plan for the proper interactive response (e.g., give food, play, run to safety). Again, it comes down to human expectations. Understanding the animal's abilities and limitations will reduce frustrations of trying to meet a goal (e.g., taming a lion). Knowing the abilities of the animal changes our expectations. Bulldogs can't swim because of the shape of their nose, similar for dogs with large chest. Humans can accommodate for these limitations when they know about them. Understanding the expectations, abilities, and limitations of artificially intelligent entities as well as the cognitively designed understanding of artificially intelligent entity expectations, abilities, and limitations of humans is vital to efficient and useful collaboration. Collaboration is much more than a mere working relationship. It is both a process and an outcome. This process is collaborative in order to work on a common problem, while understanding that each separate entity has influence on the other. The collaborative outcome is a solution where all parties can agree on the final solution. Typically, collaboration happens because an individual cannot accomplish the same goal alone. It is more than an association relationship; it is more like a partnership.

So, what is required for humans and robots, machines, to have a partnership? Likely, many of the same things as previously discussed; a sense of predictability, safety, reliability, trust, communication, knowledge, understanding, and accommodation just to name a few. We propose that everything collaborating with humans does not necessarily need to be human-like but as a minimum a need for some essential characteristics. Hence, it follows that some of the useful characteristics

might be the ones that keep humans committed to the collaboration. Who will tolerate the constant attack of a lion, or the abusive coworker, or a laptop that continues to freeze in the middle of writing documents? Each will eventually be regarded as untrustworthy and would most likely be replaced.

Several research systems exist which are important to consider when thinking of the psychology of human–SELF collaboration. In their work on intelligent mechatronics, Harashima and Suzuki [9] concluded that communicative artificial intelligence entities must be equipped with mathematical models that touch on theory of mind, mind reading, and social common sense. This level of machine must also include eye contact robots and attempt to communicate intuitively and instantaneously. Such mechatronic systems have been able to perform as Ball Room dance partners and therapy Seals. There are many mechatronics designed to augment and/or enhance human skill. One example is a machine that assists as a scrub-nurse. Just the thought of a SELF assisting in any surgery implies a huge amount of trust particularly if ultimately allowed to perform surgery autonomously. Suzuki, Pan, Harashima, and Furuta stated [9]:

> …knowledge and human psychology cannot be written sufficiently by computer algorithms; hence, the present intelligent mechatronics cannot understand the human perfectly.

Current human–robot interaction technology and design has developed from master-slave type interactions toward more collaborative. Karami, Jeanpierre, and Mouaddib [9] described a model where the robot can consider human intentions and operate without communication. Karami et al. also discussed how robots can build beliefs about human intentions by observing, collecting, and perceiving human behavior [9]. Although the experiment shown was a seemingly simple task of moving objects, the results showed further promise for human–robot collaboration more advanced than in the previous master-slave paradigm.

Research shows that humans adapt to how they respond to robots over time [16]. Initially, humans tend to use simplistic communications with robots until they learn how the robots adapt to higher order types of communication. In later work, they investigated human–robot interaction, illustrating how language and gestures help humans and robots collaborate during spatial maneuvering. They concluded that over time humans used more complex language and gestures as they learned that the robot could successfully respond to them. Giving credence to the hypothesis that as humans and robots interact, increased understanding of constraint and limitation characteristics grows and directly affects qualitative collaboration.

Trends in human–robot interaction show that several characteristics increase human trust in robots, among which reliability is a major factor. Also influencing trust is type, size, proximity, and behavior of the robot. Later research indicates that human characteristics such as ability and personality, and environmental characteristics such as task and team, along with robot performance characteristics/attributes effect training and design implications, thus, affecting human–robot collaborative team trust. Since existing bodies of research indicate clearly that trust and clear expectations are important in human–robot collaboration, significant challenges lay ahead for human adaptation to recent increases in capabilities of more highly

autonomous cognitive systems. Like human–human or human–creature relationships, little collaboration or cooperation will occur until understanding, expectations, and/or predictability become well defined in context of environment, enhanced trust, and collaboration.

4.10 Conclusions and Discussion

What we have described is a human-artificially cognitive system collaboration environment, CITE, that will allow human mentors to develop cognitive trust and reliance on autonomous system and provide knowledge products that reflects state-of-the-art cognitive interaction between artificially cognitive systems and humans, providing a new generation of human–machine collaboration, allowing each to not only learn from each other, but operate in modes that utilize the strengths of both. This includes cognitive procedural memory development that will allow improvement in attitudes and knowledge about the value artificial life forms and autonomous systems through cognitive self-awareness, self-evaluation, and self-regulation. One final thought here is about an AI system's adaptability, often called "learnability." Learnability is the ability of a system to learn and modify its behavior with time. Some examples of websites with learnability include Netflix and Amazon, which understand user preferences and come up with appropriate recommendations. Another example is a Voice Recognition System like Siri or Cortana, which picks up the semantics of language websites. However, with Cortana now responding to "I am being abused" with the number for the National Domestic Violence hotline, it is important for chatbots to be tested for comprehension of things such as sarcasm and tone which may cause significant misunderstandings and potential failures. How the system needs to communicate with its human counterparts, whatever that entails, must be considered when designing the machine learning, adapting, and reasoning methods for the artificial intelligent system. If the system has significant communication/collaboration requirements, as discussed above, the artificial intelligent entity must be tested as a coherent system, for tests aimed at testing specific requirements or single objectives causes the system to learn things that it communicates throughout the artificial neural structure of the system. We must think through communication carefully, ensuring the system can understand how humans communicate, the language they use, the idioms they use, the analogies, metaphors, etc. For when people start communicating, they communicate with machines the way then communicate with people. If the system doesn't understand it, the system may learn things that were very unintended, as we discussed previously.

References

1. Carbone, J. N. (2010). *A framework for enhancing transdisciplinary research knowledge*. Lubbock, TX: Tech University Press.
2. Crowder, J., Raskin, V., & Taylor, J. (2012). Autonomous creation and detection of procedural memory scripts. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
3. Brillouin, L. (2004). *Science and information theory*. New York: Dover.
4. Llinas, J., Bowman, C., Rogova, G., Steinberg, A., Waltz, E., & White, F. (2004). Revisiting the JDL data fusion model II.
5. Nass, C., & Moon, Y. (2000). Machines and mindlessness: Social responses to computers. *Journal of Social Issues, 56*(1), 81–103.
6. Raskin, V., Taylor, J. M., & Hempelmann, C. F. (2010). *Ontological semantic technology for detecting insider threat and social engineering*. Concord, MA: New Security Paradigms Workshop.
7. Taylor, J. M., & Raskin, V. (2011). Understanding the unknown: Unattested input processing in natural language. In *FUZZ-IEEE Conference*, Taipei, Taiwan.
8. Crowder, J. A., & Carbone, J. N. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 12th International Conference on Artificial Intelligence*, Las Vegas, NV.
9. Crowder, J., & Friess S. (2012). Artificial psychology: The psychology of AI. In *Proceedings of the 3rd International Multi-Conference on Complexity, Informatics, and Cybernetics*, Orlando, FL.
10. Mayer, J., Salovey, P., & Caruso, D. (2004). Emotional intelligence. *Psychological Inquiry, 15*(3), 197–215.
11. Chai, S. (2004). Artificial intelligence and social theory: A one way street? *Perspectives, 27*(4), 11–12.
12. Crowder, J. (2012). Cognitive system management: The polymorphic, evolutionary, neural learning and processing environment (PENLPE). In *Proceedings for the AIAA Infotech@ Aerospace 2012 Conference*, Garden Grove, CA.
13. Crowder, J. A. (2010). Anti-terrorism learning advisory system (ATLAS): Operative intelligent information agents for intelligence processing. In *Proceedings of the AIAA Infotech@ Aerospace-2010*, Atlanta, GA.
14. Crowder, J. A. (2010). The continuously recombinant genetic, neural fiber network. In *Proceedings of the AIAA Infotech@Aerospace-2010*, Atlanta, GA.
15. Crowder, J. (2012). The artificial cognitive neural framework. In *Proceedings for the AIAA Infotech@Aerospace 2012 Conference*, Garden Grove, CA.
16. Crowder, J., Carbone, J., & Friess, S. (2013). The cognitive, interactive training environment. In *Proceedings of the 14th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.

Chapter 5
Abductive Artificial Intelligence Learning Models

5.1 Introduction

Abduction is formally defined as finding the best explanation for a set of observations or inferring cause from effect. There are many definitions of learning, depending upon the discipline [1]. For instance:

- **Psychology Definition**: A relatively lasting change in behavior that is the result of experience. Learning became a major focus of study in psychology in the early part of the twentieth century as behaviorism rose to become a major school of thought.
- **Classical Definition**: Measurable and relatively permanent change in behavior through tacit or explicit learning, experience, mental insights, instruction, or study [2–4].
- **Causal Definition**: Rapid Effective Causal Learning (RECL) is a method learning based upon measures of causality.
- **Other Definitions**: (a) Learning is the increase of knowledge, (b) learning is memorization, (c) learning is the acquisition of facts, procedures, etc., which can be retained and/or utilized in practice, (d) learning is the abstraction of meaning, (e) learning is the interpretive process aimed at the understanding of reality.

How we define learning affects our understanding and drives how we look at AI learning. In order to work toward a biologically inspired learning model of artificially intelligent applications, we investigate learning models through the lens of "Occam Learning." The notion of Occam Abduction relates to simultaneously driving down ambiguity and complexity and finding the simplest explanation with respect to inferring cause from effect.

Abductive Learning: Deriving a set of hypotheses that can be used to explain a given set of facts or observations [5]. The inference of abductive learning or reasoning is that one or more of the hypotheses, if true, can be used to explain the occurrence of the given facts or observations. We look at a simple framework.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_5

1. D is the domain for a set of effects, E_i, and possible causes (hypotheses), C_j.
2. $\{C, E, D\}$ represents an abductive causal theory, which includes an explanation set (C_j, $j = 1, \ldots, n$) for a finite of effects/observations (E_i, $i = 1, \ldots, m$).

The set of abductive hypotheses (causes) constitutes an explanation for the set of observations (complete and parsimonious) if and only if there is no subset of C that fully explains the effects, E, that C does. We work with abduction to provide a non-monotonic reasoning paradigm to overcome limitations/false conclusions in deductive reasoning. The inference description below illustrates this:

Deduction:	A → B	All marbles in the bag are aggies.		
		A	I have marbles from the bag.	
		B	The marbles are aggies.	
Induction:	A	I have marbles from the bag.		
		B	The marbles are aggies.	
		Possibly	A → B	All marbles in the bag are aggies.
Abduction:	A → B	All marbles in the bag are aggies.		
		B	I have aggie marbles.	
		Possibly	A	These marbles are from the bag.

The abduction form of inference, using hypotheses to explain observed phenomena, is a useful and flexible methodology of reasoning on incomplete or uncertain knowledge. For instance, if.

$$\text{A} \rightarrow \text{C and B} \rightarrow \text{C, and A} \neq \text{B}$$

then both A and B are plausible hypotheses for observation/effect C, which drives us to the observation that abductive learning is inherently uncertain, and hypotheses should be ranked by their possibility ranking (in a fuzzy sense). While this captures the central characteristics of abductive reasoning and learning, mainly the creation of causes (hypotheses) that provide adequate explanation for the observations/effects, what is difficult to measure are effects of emotions on abductive learning, as uncertainty drives us to explore how we "feel" about certain explanations/hypotheses. Here, we investigate two separate abductive learning models, shown in Figs. 5.1 and 5.2. Figure 5.1 illustrates a learning model with no emotional component, either in observations or conclusions, as opposed to Fig. 5.2, where emotions play a considerable role, including the notion of "curiosity," which is crucial to abductive thinking, learning, and reasoning [6].

In Fig. 5.1, because reasoning is non-monotonic, the plausibility/possibility of set of hypotheses will increase or decrease as data are collected (continued observation) and the general truth of the hypotheses are assessed (abstract conceptualization). This may cause some hypotheses to be eliminated and a new set of hypotheses created from the combined set of current observations.

In Fig. 5.2, the non-monotonic nature of reasoning may cause the plausibility/possibility of a set of hypotheses to be increased or decreased by not only continued data collection (observations), but by the perceived plausibility of the overall effects

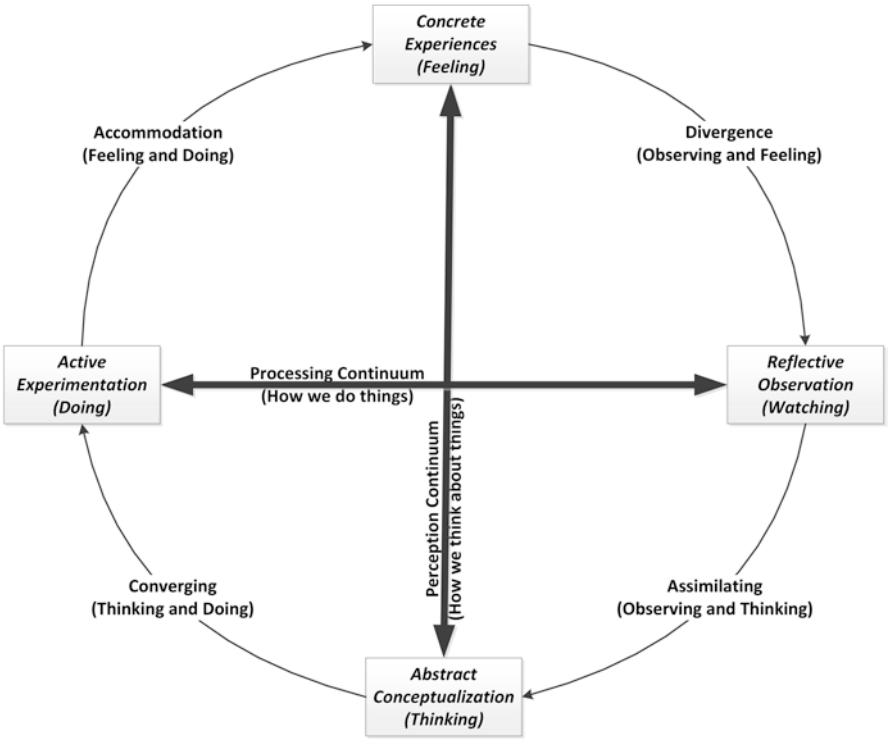


Fig. 5.1 Non-emotion learning model

of the observations (satisfaction/hopelessness). Uncertainty in the causal relations and disappointment that a perceived relationship does not exist may manifest itself in increased uncertainty about the outcomes. While this may seem unnecessary and even unwanted in machine learning, use of "emotional triggers" gathered from previous encounters with cause/effect relationships may be useful in autonomous abductive learning systems, where safety of the system is at stake, especially when the system must deal with incomplete knowledge and/or data.

Hence, it is critical to account for and understand that to increase the level of fidelity of understanding, there exists inherent inference values or levels of understanding, satisfaction, hopelessness, plausibility, and possibility. AI abductive learning models then can be augmented with appropriately weighted relationship mappings over time to provide the added fidelity to each newly discerned autonomous abductive possibilistic relationship inference. A Knowledge Relativity Thread (KRT) [2, 7] is used to provide a high-fidelity weighted inference mechanism that more effectively transmits emotional triggers and detailed context within common operational environments. This provides information inference extrapolations through distance and mass and hence, can effectively provide representations of knowledge and related context.

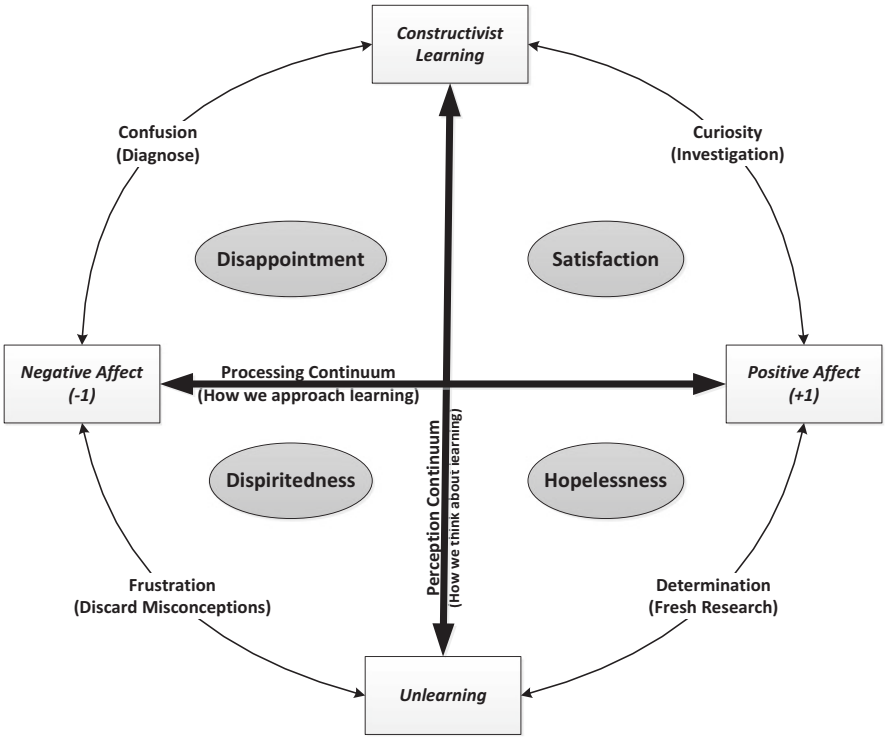


Fig. 5.2 Learning model with emotions

5.2 Representations of Learned Knowledge and Context

The representation of knowledge and context is defined as an implement designed for representing knowledge and context as relationship structures, as defined by Trochim [8], and Novak and Canas [9]. The Representation of Knowledge and Context approach was derived from Newton's law of gravitation [10], the concept, as described by Polyn and Kahana [11] that recall of a known item representation is driven by an internally maintained context representation, and Howard [12], of linking knowledge items and context representations in memory. This accomplished temporal-based recognition and state of context to cue representations for recall. Additionally, Representation of Knowledge and Context was derived from five needs. Firstly, need for representation of quality context [13]. Secondly, the need for integration of new emergent physical characteristics of knowledge [14]. Thirdly, the need to represent strong relationships which exist between the environment and objects found within [15]. Fourthly, based upon a need described [14], to use physics as a tool for describing reality since the physics domain is rich in mathematical formalization. Therefore, Newton's laws are prime candidates for representing

reality since they are built upon the scientific method and empirical pedigree. Lastly, the need for representing knowledge and context using analytic induction [16].

The representation of knowledge and context formula is introduced here and is presented by Eq. (5.2). The independent results which follow are mathematical evaluations extended from Newton's law of gravitation shown in Eq. (5.1). Newton's Law of Gravitation formula is,

$$F = G\frac{\left(M_1 M_2\right)}{r^2} \tag{5.1}$$

where:

F is the magnitude of the gravitational force between the two objects with mass.
G is the universal gravitational constant.
M_1 is the mass of the first mass.
M_2 is the mass of the second mass.
r is the distance between the two masses.

This equation was used as an analogy for the derivation of mathematical relationship between bases made up of two objects of knowledge.

Hence, Carbone abstracted Newton's Law of Gravitation as an analogy of Eq. (5.1) that represents relationships between two objects of knowledge using context is written as Eq. (5.2) shown below, which describes the components of the formula to represent relationships between two objects of knowledge using context [17]:

$$A = B\frac{\left(I_1 I_2\right)}{c^2} \tag{5.2}$$

where,

A is the magnitude of the attractive force between the two objects of knowledge.
B is a balance variable.
I_1 is the importance measure of the first object of knowledge.
I_2 is the importance measure of the second object of knowledge.
c is the closeness between the two objects of knowledge.

Comparing the parameters of Eqs. (5.1) and (5.2) F and A have similar connotations except F represents a force between two physical objects of mass M_1 and M_2 and A represents a stakeholder magnitude of attractive force based upon stakeholder determined importance measure factors called I_1 and I_2. As an analogy to F in Eq. (5.1), A's strength or weakness of attraction force was also determined by the magnitude of the value. Hence, the greater the magnitude value, the greater the force of attraction and vice versa. The weighted factors represented the importance of the objects to the relationships being formed. The Universal Gravitational Constant G is used to balance gravitational equations based upon the physical units of measurement (e.g., SI units, Planck units). B represents an analogy to G's concept of a balance variable and is referred to as a constant of proportionality. For simplicity, no

units of measure were used within Eq. (5.2) and the values for all variables only showed magnitude and don't represent physical properties (e.g., mass, weight) as does G. Therefore, an assumption made here is to set B to the value of 1.

For simplicity, all these assume the same units and B was assumed to be one. The parameter c in Eq. (5.2) is taken to be analogous to r in Eq. (5.1). Stakeholder perceived context known as closeness c represented how closely two knowledge objects (KO) are related (Fig. 5.3).

5.3 Elementary Abduction

Assumptions for Abductive Learning:

1. C_j's are mutually exclusive and constitute exhaustive coverage of the set of effects (E_i's).
2. Each of the C_j's is conditionally independent.
3. Each of the C_j's is not mutually incompatible with any other C_j.
4. None of the C_j's cancel the abductive explanatory capability of any other C_j. For example, C_1 implies an increase in a value, while C_2 implies a decrease in a value. In this case, one is used to support the hypothesis and the other is used to rebut the hypothesis.

Figure 5.4 below illustrates the abductive learning, based on a hypothesis-driven "puzzlement or surprise" notion [18].

Here, we consider the nature of explanation. When effects (observations) are presented for which there is no conceptual explanation (the conceptual ontology contains no constructs for the observations/effects), we must create a set of possible hypotheses and test them to find that set that provides a possible explanation for the observations. The "possible" hypotheses are used to create a new generation of hypotheses that are tested against current and continued observations (effects).

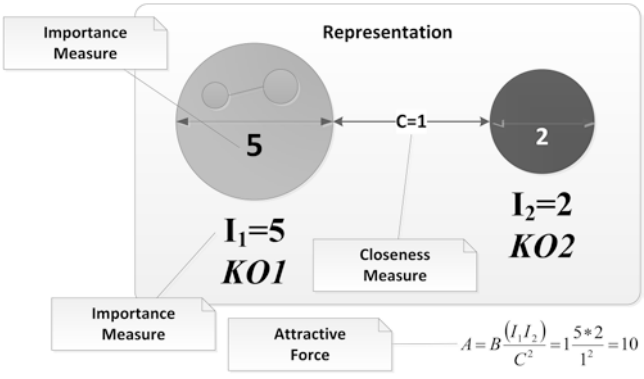


Fig. 5.3 Representation of learning knowledge and context

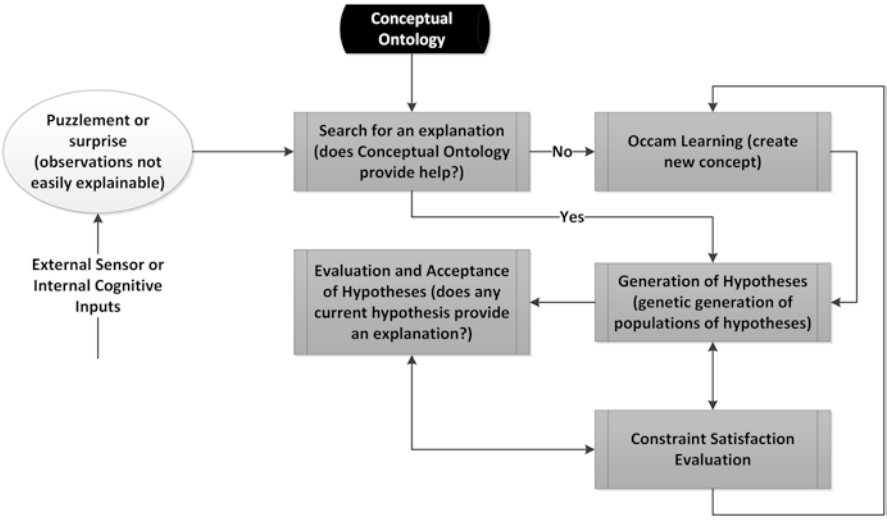


Fig. 5.4 Abductive learning for new concepts

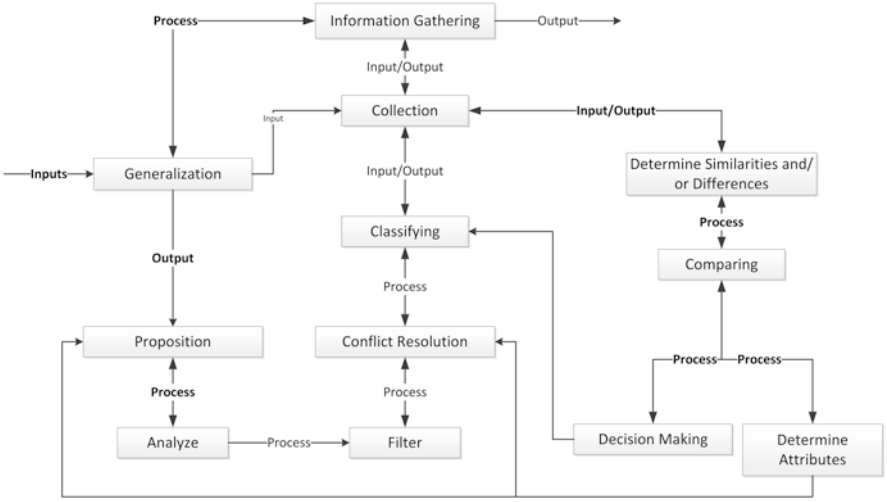


Fig. 5.5 A generalized abductive learning model

Surprise, or puzzlement, is one aspect of abductive learning, how to learn those things we didn't know we needed to learn. From here we expand to a generalized model of Abductive Learning, as illustrated in Fig. 5.5 [19].

In order to begin the processes of formulating hypotheses (causes) for a given set of observations (effects), we first generalize the observations into categories, assuming the categories that are applicable to the observations already exist in the conceptual ontology. If now, new concepts must be created that accommodate the

observations, which may require a higher level of hypothesis generation and testing to determine the concept to be added to the conceptual ontology.

5.4 Artificial Abduction Hypothesis Evaluation Logic

To provide an instantiation of an artificial abductive learning system within an autonomous or semi-autonomous artificially intelligent system, we provide an example of a Learning State Diagram, shown in Fig. 5.6 [20].

Within an autonomous AI system, evaluations must be made during the abductive process to determine if a solution (set of hypotheses) is converging or diverging (called losing focus). If the system cannot converge on a set of hypotheses, a new set of hypotheses must be created and evaluated. If continuous sets of hypotheses are not converging on a set of explanations (causes) for the observations (effects), then the system is continually diverging from explaining the effects and the abductive learning process must be terminated until more data (observations) are available. Otherwise, the system could be put into a continual loop without satisfactory

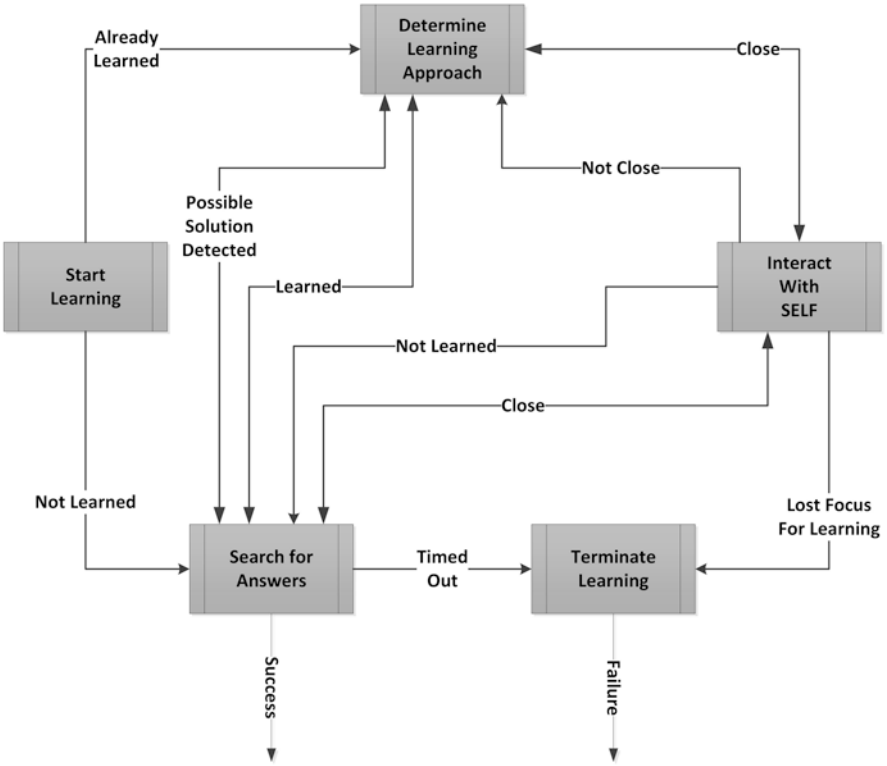


Fig. 5.6 Abductive learning state diagram

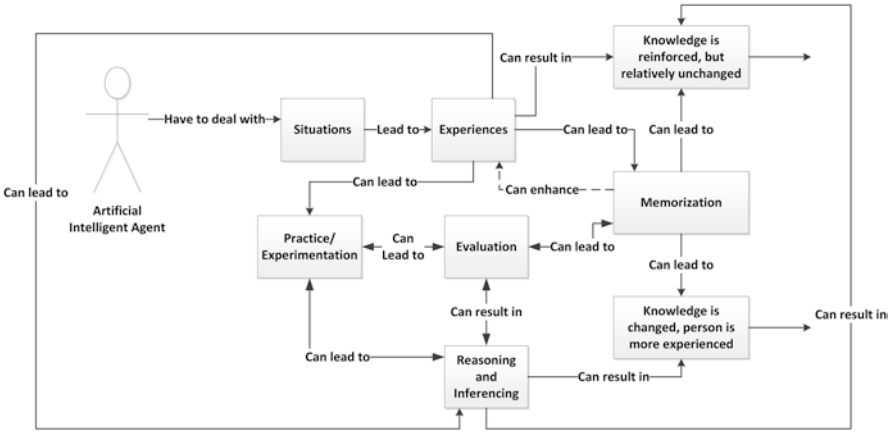


Fig. 5.7 Abductive learning model without implicit learning

results and with continued (unnecessary) resource utilization. This learning state evaluation process is utilized within the overall abductive learning process, shown in Fig. 5.7. This model does not consider implicit learning or learning that happens without the entity realizing it has happened. This will be discussed in later chapters.

This approach is considered abductive in that it does not depend on deductive or inductive logic though these may be included as part of the overall hypotheses. Instead, the Abductive Learning Model depends on non-analytic inferences to find new possibilities based upon hypothesis examples (abductive logic). To facilitate the abductive learning system to effectively map hypotheses to observations, there are many levels of mapping that must be created. Figure 5.8 illustrates the lower level mapping of experiences. Here, the system must identify the types of experiences (observations) that are occurring to drive the hypothesis generation algorithms to derive causes (hypotheses) applicable to the observations (experiences [21]).

Likewise, the situational type helps to determine the form and content of the generated hypotheses to be evaluated. This situational breakdown is shown in Fig. 5.9. The abductive learning provides mapping from explanations (hypotheses) to data (observations) and must consider the types of inferences required, along with those factors which would influence the overall inference. This inference breakdown is illustrated in Fig. 5.10. As with the overall learning model shown in Fig. 5.7, this inference model does not consider implicit learning; this will be added later and discussed.

An example of a further decomposition of the abductive inference process is shown in Fig. 5.11, with a mapping/breakdown of the context-based inference process. All these factors must be considered and folded into the abductive hypotheses generation process, in order to effectively create hypotheses (causes) relevant to the observations (experiences or effects) that are sensed by the artificially intelligent systems. Many AI systems are implemented utilizing intelligent S/W agents [22]. For an abductive learning system, we envision several agent types would be

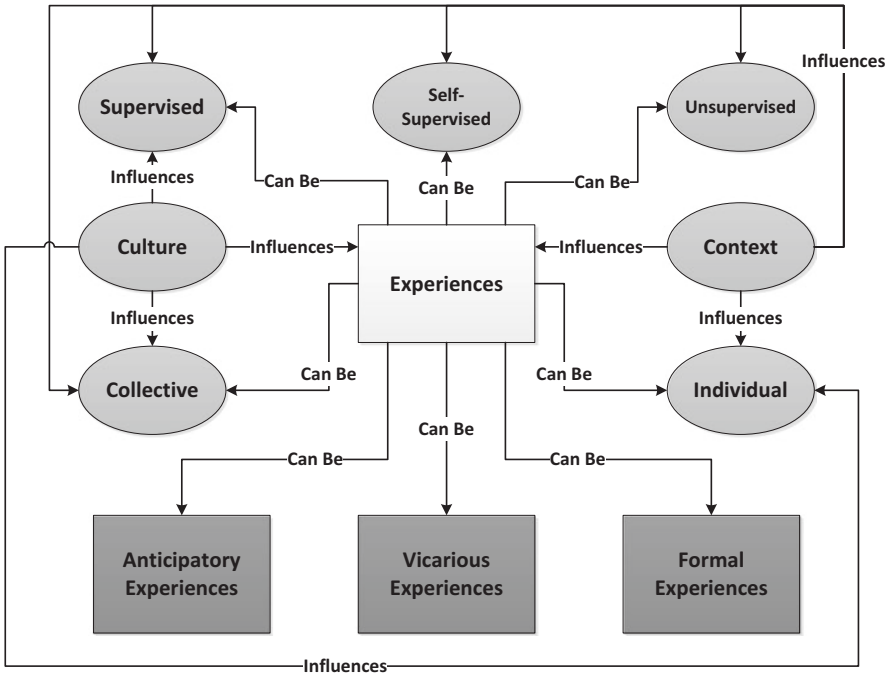


Fig. 5.8 Experience decomposition

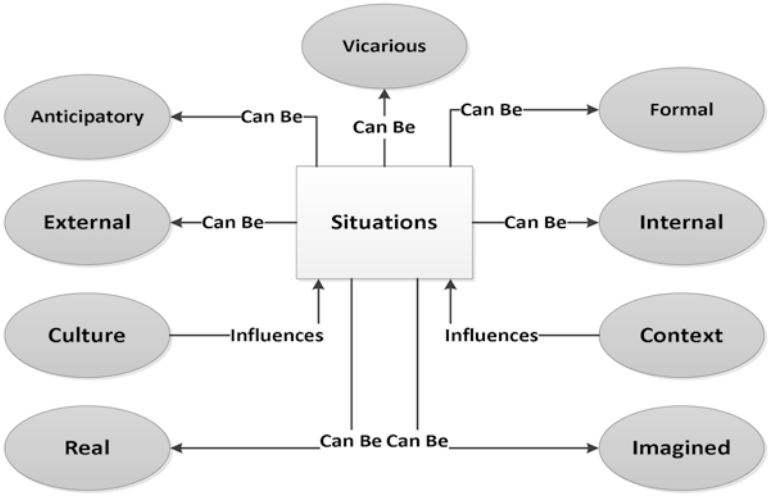


Fig. 5.9 Situation decomposition

required to facilitate all aspects of abductive learning/reasoning within an AI system. One possible distribution of S/W agents and their abilities/capabilities within the abductive reasoning process is shown in Fig. 5.12. Here, each type of agent is described in terms of their handling of the cause/effect process [6].

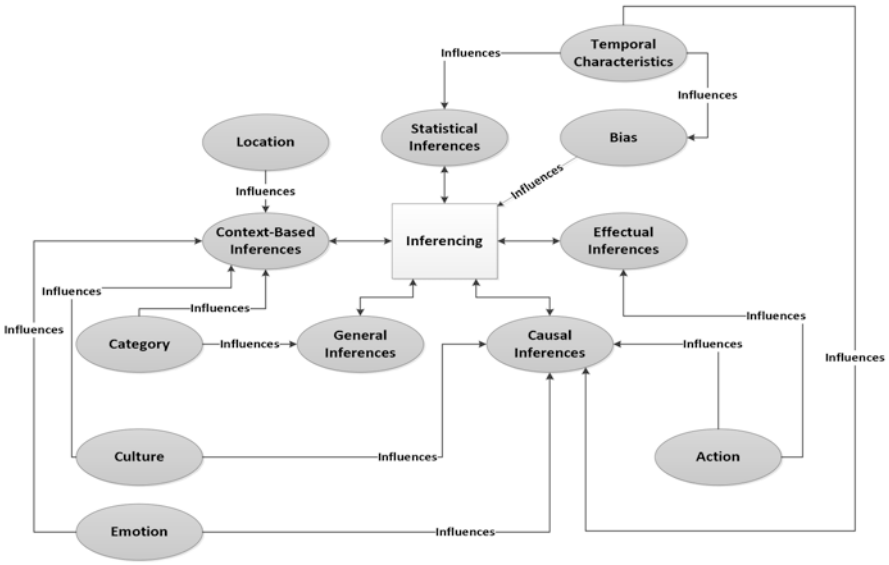


Fig. 5.10 Inference decomposition without implicit learning

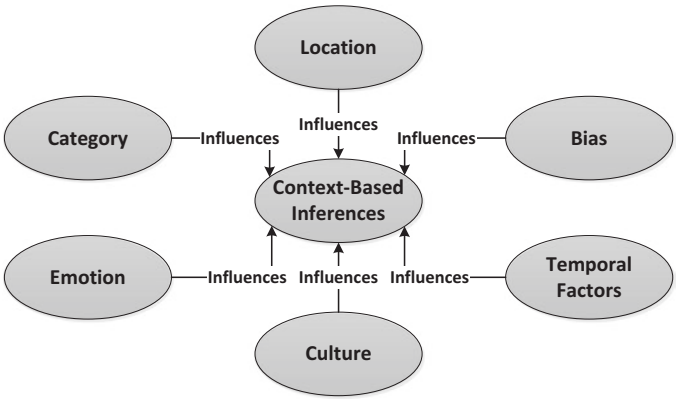


Fig. 5.11 Context-based inference decomposition

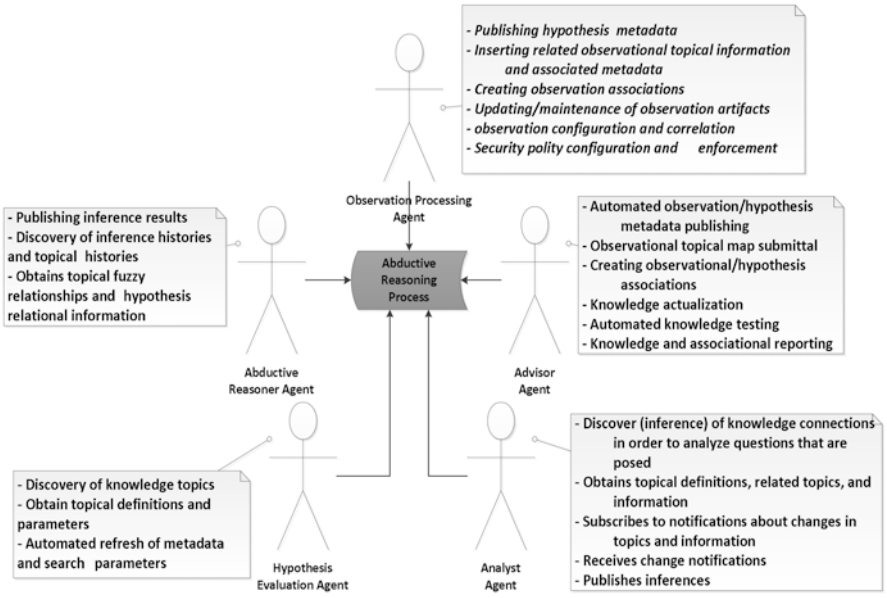


Fig. 5.12 S/W agent descriptions for abductive learning system

5.5 Conclusions

Here, we have laid the rudimentary foundations for learning structures that will be required for real-time autonomous, abductive learning in AI systems. Abduction Learning will provide the ability for simple observation explanation that feeds more complex memory and inference systems within an AI cognitive system to allow the autonomous system to think, reason, and evolve. We have but scratched the surface in providing constructs and methodologies required for an autonomous real-time AI system.

How we expect the artificial intelligent system to learn, case-based, experience-based, rule-based, data-based, etc., dramatically changes how we should design the test strategy for the system. If the system is designed to learn, think, reason, and infer the way humans do, we should expect to have to test the system like we test humans, and artificial psychology is crucial to understanding how to accomplish this.

References

1. Levesque, H. J. (1989). A knowledge-level account of abduction. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, Detroit, MI (pp. 1061–1067).
2. Carbone, J. (2010). *A framework for enhancing transdisciplinary research knowledge*. Lubbock, TX: Texas Tech University.

3. Nonaka, I. A., & Takeuchi, H. A. (1995). *The knowledge-creating company: How Japanese companies create the dynamics of innovation*. Oxford: Oxford University Press.
4. Polanyi, M., & Sen, A. (2009). *The tacit dimension*. Chicago: University of Chicago Press.
5. Crowder, J. (2012). Possiblistic, abductive neural networks (PANNs) for decision support in autonomous systems: The advanced learning, abductive network (ALAN). In *Proceedings of the 1st International Conference on Robotic Intelligence and Applications*, Gwanju, Korea.
6. Crowder, J., & Carbone, J. (2011). Occam learning through pattern discovery: Computational mechanics in AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
7. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*.
8. Trochim, W. M. K. (1989). An introduction to concept mapping for planning and evaluation. *Evaluation and Program Planning, 12*, 1–16.
9. Novak, J., & Cañas, A. (2008). *The theory underlying concept maps and how to construct and use them*. Pensacola, FL: Florida Institute for Human and Machine Cognition. Retrieved from http://cmap.ihmc.us/Publications/ResearchPapers/TheoryCmaps/TheoryUnderlyingConceptMaps.htm.
10. Hibbeler, R. C. (2009). *Engineering mechanics: Dynamics*. Upper Saddle River, NJ: Prentice Hall.
11. Polyn, S., & Kahana, M. (2008). Memory search and the neural representation of context. *Trends in Cognitive Sciences, 12*, 24–30.
12. Howard, M., & Kahana, M. (2002). A distributed representation of temporal context. *Journal of Mathematical Psychology, 46*, 269–299.
13. Ejigu, D., Scuturici, M., & Brunie, L. (2008). Hybrid approach to collaborative context-aware service platform for pervasive computing. *Journal of Computers, 3*, 40.
14. Nicolescu, B. (1998). Godelian aspects of nature and knowledge. In G. Altmann & W. Koch (Eds.), *Systems: New paradigms for the human sciences* (p. 385). Berlin: Walter de Gruyter.
15. Torralba, A. (2003). Contextual priming for object detection. *International Journal of Computer Vision, 53*, 169–191.
16. Glaser, B., & Strauss, A. (1977). *The discovery of grounded theory: Strategies for qualitative research*. Chicago: Aldine.
17. Carbone, J. N. (2010). *A framework for enhancing transdisciplinary research knowledge*. Lubbock, TX: Tech University Press.
18. Crowder, J. (2013). The advanced learning, abductive network (ALAN). In *Proceedings of the AIAA Space 2013 Conference*, San Diego, CA.
19. Dimopoulos, Y., & Kakas, A. (1996). Abduction and inductive learning. In L. De Raedt (Ed.), *Advances in inductive logic programming* (pp. 144–171). Amsterdam: IOS Press.
20. Crowder, J., Friess, S., & Carbone, J. (2013). *Artificial cognition architectures*. New York: Springer. ISBN 978-1-4614-8071-6.
21. O'Rorke, P. (1994). Abduction and explanation-based learning: Case studies in diverse domains. *Computational Intelligence, 10*, 295–330.
22. Crowder, J., Scally, L., & Bonato, M. (2012). Applications for intelligent information agents (I2As): Learning agents for autonomous space asset management (LAASAM). In *Proceedings of the International Conference on Artificial Intelligence, ICAI'12*, Las Vegas, NV.
23. Abe, A. (1997). The relation between abductive hypotheses and inductive hypotheses. In *Proceedings of the IJCAI97 Workshop on Induction*.

Chapter 6
Artificial Creativity and Self-Evolution: Abductive Reasoning in Artificial Life Forms

6.1 Introduction

Turing and others have hypothesized that computers cannot be creative, due to the absence of novelty in its flow of information processing. We believe the use of stochastic, possibilistic abductive networks provides a very novel approach to information processing, allowing the artificially intelligent system to vary its information processing flow, depending on the generated hypotheses and continuously recombinant neural fiber network creation process.

The hypothesis we would like to consider here is that creativity is a directly related problem-solving activity in which explorations of problem spaces lead to the expansion of belief domains. We believe successful expansion of beliefs in an artificial cognitive system is initiated by algorithms that provide updates of the artificial cognitive system's Conceptual Ontology [1]. Here we discuss the general heuristics within the genetic hypothesis generation process that will be used to guide the support and rebuttal informational search processes and problem-solving activities, which includes strategies for examining, comparing, altering and combining concepts, strings of symbols, and the heuristics themselves. But what kind of creativity is possible for the Artificial Intelligence (AI) system in this context? We believe the answer is that it is like the one which humans experience in our everyday life: the experience of new and original ideas that have value, based on the overall goals, constraints, and mission directives of the environment the AI system is within. Within this context, we put forth the design and implementation of algorithms required for an Advanced Learning, Abductive Network (ALAN) as a candidate to facilitate artificial creativity (i.e., advanced hypothesis generation and testing), and therefore autonomous, real-time decision support, from an objective perspective; the abductive dialectic argument structure providing the inference engine upon which artificial creative reasoning is based.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_6

6.2 Human vs. Artificial Reasoning

6.2.1 Human Reasoning Concepts

Human reasoning is dynamic in that there are many processes involved. There are different types of reasoning necessary to allow humans to navigate their world effectively and efficiently. Here we will provide a brief overview; as the topic of human reasoning is vast. So much information comes into the brain at one time that it is impossible to consciously be aware of all of it. Just imagine for a minute how many things the human brain is handling in one instant. We have memories, associations, and habitual ways of thinking. We have beliefs, assumptions, and predictions. We have experiences, past, present, and planned. We have senses and perceptions. We have defense mechanisms and feelings. Our brains are active! It is hard to imagine what all is happening in an instant of experience for a human, but it is essential to explore the possibilities in order to understand how the concepts will translate into artificial reasoning.

6.2.2 Modular Reasoning

Cognitive modularity seems to have flourished with Fodor [2]. He thought that humans use domain-specific modules that together form part of the reasoning system within the human brain. According to Fodor, there are conditions for modular cognition; one is that other parts of the brain have limited access to each reasoning module. This type of reasoning is mandatory, innate, shallow, and very fast. He also stipulated that each module was fixed to a neural architecture and that information was encapsulated since other modules have limited access to each other. More modern psychology believes that cognitive modularity as actually massive modularity. This school of thought suggests that the mind is even more modular with specific functions and specialization [3]. This type of Modular Reasoning is used within the SELF Sensory Processing, before Sensory Integration. There is differing views on massive modularity. According to Raymond Gibbs and Van Orden [4], massive modularity theory has its problems empirically. They state that the studies fail to be able to separate modules. They also argue that massive modularity theory fails to discover input criteria and state that it may be impossible given the nature of context embedded human nature. Lastly, they argue that massive modularity does not acknowledge the interaction of brain, body, and world in human thinking.

6.3 Distributed Reasoning

The distributive theory suggests that there is more to the brain than separate modules. Beyond some very specific areas such and motor control, distributed reasoning theory suggest there are many fuzzy connections between systems of the brain. The distributive theory challenges boundaries of the mind, skull, and even body considering the environment, artifacts, and people. This theory is reflected in the Fuzzy, Possibilistic Abductive Network utilized within the SELF cognitive framework.

The distributive reasoning theory by Hutchins [5] provides some insights into human reasoning. Hutchins provides five different models that affect human reasoning. First, he postulates that there are modules within the brain that are specialized in function and structure and are united in a complex way. Second, he argues that cognition at a macro level is distributed outside the individual, such as the media. Media can be internal and external. Third, there is human culture which influences the individual. Fourth, there is society which cognitive activity is distributed in tools, rules, and contexts. Finally, he argues that cognition is distributive in time, both vertical and lateral time dimensions of the subject [6].

Yvonne Rogers [6] provides a detailed analysis of the distributive cognitive model. Rogers cites Hutchins as creating a computational model of two modules of the brain that can together recover depth that neither module alone could do. One general assumption of the distributive human cognitive system is that it is made of more than one module and that each module in the cognitive system has different cognitive properties than the individual and is different than the cognitive brain. Another general assumption made by Rogers is that members (modules) of the system have knowledge that is both variable and redundant and that members of the system can pool resources. Another is distribution of access to information. This enables the coordination of expectations and coordination of action within the human biological reasoning framework [6]. These concepts are utilized throughout the SELF, which utilizes localized processing modules (processing "experts") as well as distributed Cognitive Perceptron Intelligent Software Agents (called Cognitrons) experts that communicate and collaborate throughout the SELF cognitive system.

6.4 Types of Reasoning

Humans can reason in different ways. The three major human reasoning strategies are inductive, deductive, and abductive.

Inductive Reasoning: Inductive reasoning involves coming to a conclusion after evaluating facts; reasoning from specific facts to a general conclusion. This allows for inferences. It also requires human experience to validate any conclusion. An example might be: Zebras that are at the zoo have stripes; therefore, all zebras have stripes [7].

Deductive Reasoning: Deductive reasoning is just the opposite. Deductive reasoning moves from a general principle to specific cases. This type of reasoning is based on accepted truths. An example of deductive reasoning might be: All zebras have stripes therefore when I go to the zoo the zebra will have stripes.

Abductive Reasoning: Abductive reasoning allows for explanatory hypothesis generation or generating ideas outside of the given facts to explain something that has no immediate satisfactory explanation.

There are many ways in which people reason, but often human reasoning follows either inductive or deductive reasoning. Consider a few ways in which humans think about things. Take cause and effect reasoning where causes and effects are considered. Analogical reasoning is a way of relating things to other novel situations. Comparative reasoning as it implies is comparing things, in which humans often engage. Still another reasoning method is conditional reasoning, or if/then reasoning. Many of us have used the pros and cons methods of reasoning also. Then there is Systemic reasoning where the whole is greater than the sum of its parts. There is also reasoning using examples. As you can see there are numerous ways in which humans can reason about things and situations. These are all logical ways of reasoning.

6.5 Artificial "SELF" Reasoning

As discussed above, reasoning takes on many forms, but two important ones within the SELF is both induction and abduction:

- Induction: Extrapolates from information and experiences to make accurate predictions about future situations.
- Abduction: Genetic algorithms generate populations of hypotheses and a Dialectic Argument (Tolemin) Structure is used to reason about and learn about a given set of information, experiences, or situations, also called "***Concept Learning***."

Earlier we discussed the use of hypothesis-based reasoning. Here, we provide more detail of its architecture and design within an artificial neural structure. Hypothesis-based reasoning structures seek answers to questions that require interplay between doubt and belief, where knowledge is understood to be fallible. This "playfulness" is the key to searching and exploring information. Utilizing this framework for reasoning about information, hypotheses, and problems provides a robust, adaptive information processing system capable of handling new situations. Here we utilize abductive logic, sometimes called critical thinking, in order to distinguish it from more formal logic methods like deduction and induction. Whereas data mining utilizes induction to develop assertions that are probably true, the dialectic search uses abductive logic methods and processes to develop hypotheses that are possibly true. We do not use Bayesian methods because they cannot measure possibilistic but measure probabilistic metrics. Instead, we utilize a fuzzy

implementation of Renyi's entropy and mutual information theory to provide possibilistic measure of mutual information and topical separation [8].

6.6 Artificial, Possibilistic Abductive Reasoning

The original McCulloch-Pitts model of a neuron contributed greatly to our understanding of neuron-based systems. However, their model failed to consider that even the simplest type of human nerve cell exhibits non-deterministic behavior [9, 10]. Some have attempted to take this into account through modeling this as randomness, creating a stochastic neural network, but much of the behavior is not random, but carries a type of imprecision which is associated with the lack of a sharp transition from the occurrence of an event to the non-occurrence of the event. This leads us to the definition of a network not steeped in Bayesian statistics (a Bayesian Belief Neural Network—BBNN), but one utilizing possiblistics, based on fuzzy characteristics, combined with an abductive, hypothesis-based decision network; and thus, creating a Possiblistic, Abductive Neural Network (PANN) [11]. Here, we discuss the theory and architecture for a Possiblistic, Abductive Neural Network capable of complex hypothesis generation and testing, leading to artificial creativity and discovery within a SELF [12].

6.6.1 Artificial Creativity in a SELF

Neuroscience research into the human perceptron [13] determined that the noise and imprecision in the human nervous system was not, in fact, inconvenient, but was essential to the types of computations the brain performed [14]. The brain learns to make spatio-temporal associations in the presence of noisy, imprecise information, and any artificially intelligent system that tries to emulate human processing must be able to make similar noisy, imprecise associations within its artificial neural systems even when they are not completely specified, have incomplete, imprecise, or conflicting information, as well as taking into account the behavior of the entity, i.e., accounting for its own internal state [15].

This leads to a Possiblistic, Abductive Neural Network (PANN) [16] that is capable of complex hypothesis generation and testing in the presence of multiple, noise, imprecise, and possibly incomplete information; the types of environments an autonomous SELF is likely to be found. These conditions are typical in real-time processing situations and will be essential for complex decision support to system operators and will be crucial as we move toward autonomous systems that must learn, reason, analyze, and make critical decisions in real-world environments. This work will form the basis for an Advanced Learning, Abductive Network (ALAN) that will mimic human reasoning to provide autonomous, real-time, complex decision support.

6.7 The Advanced Learning Abductive Network (ALAN)

6.7.1 Artificial Creativity Through Problem Solving

Turing and others have hypothesized that computers cannot be creative, due to the absence of novelty in its flow of information processing. The use of stochastic, possiblistic abductive networks provides a very novel approach to information processing, allowing the artificially intelligent system to vary its information processing flow, depending on the generated hypotheses and continuously recombinant neural fiber network creation process [17].

One hypothesis we would like to consider here is that creativity is directly related to problem-solving activity in which explorations of problem spaces lead to the expansion of belief domains. A successful expansion of beliefs is initiated by an update of the cognitive system's Conceptual Ontology [18]. General heuristics within the genetic hypothesis generation process guide the support and rebuttal informational search processes and problem-solving activities, which include strategies for examining, comparing, altering and combining concepts, strings of symbols, and the heuristics themselves [19].

But what kind of creativity is possible for a SELF in this context? We believe the answer is that it is like the one which humans experience in our everyday life: the experience of new and original ideas that have value, based on the overall goals, constraints, and mission directives of the environment the SELF is within.

6.7.2 ALAN Abductive Reasoning Framework

As discussed above, hypothesis-based reasoning is a reasoning framework that seeks answers to questions that require interplay between doubt and belief, where knowledge is understood to be fallible. Cognitrons which are capable of learning and reasoning about information, hypotheses, and problems provide a robust, adaptive information processing system capable of handling new situations. The key value of the Cognitrons within the abductive, hypothesis-based reasoning framework is that they provide the ability to learn from sensory data and from each other [20]. Using unsupervised learning methods, the Cognitrons have the potential to provide the operations and analytical structures to extract knowledge and context from various sources of information. Cognitrons can be cloned to support as many operators as required and as the system resources allow.

In the abductive, hypothesis-based processes, information is utilized to generate and assess hypotheses from thought processes created by Cognitrons. This is achieved by utilizing the Cognitrons to learn and reason about the hypotheses and information utilizing a Dialectic Argument Structure (DAS) framework shown in Fig. 6.1. Cognitrons, as discussed, are autonomous software agents that create an information agent ecosystem, comprehending its external and internal environment

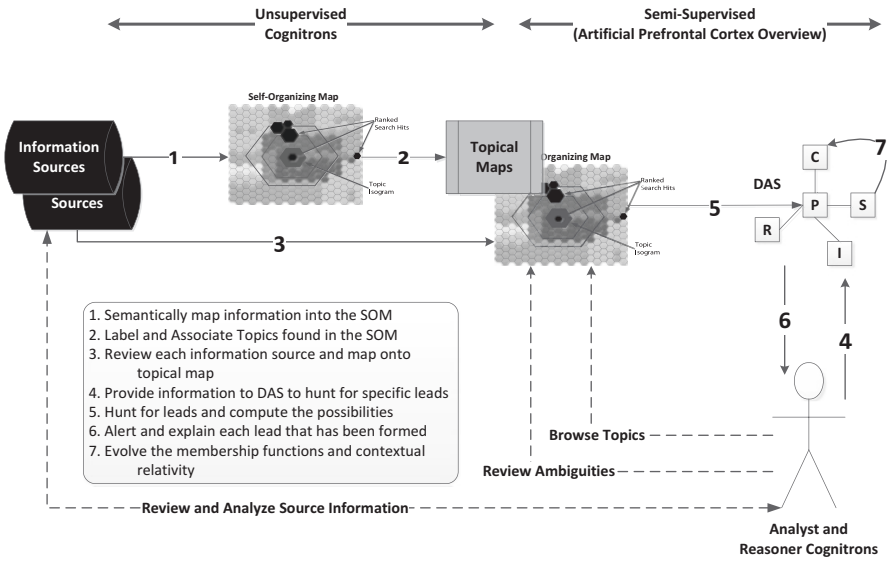


Fig. 6.1 The ALAN DAS information search process

and acting on it over time, in pursuit of its own agenda and goals, to affect what it comprehends in the future.

Alerts based on the measure of possibility (certainty) of information inform an Interface Cognitron there is information for review. Constructs within the ALAN framework rank information and flag those that are the most certain. The review is facilitated by presenting the operator or user with the DAS warrant, backing and links to support and rebuttal sources, traced back through the self-organizing topical maps. This autonomous information search process includes a review process which engages the ALAN cognitive processes, which include critical learning and reasoning objectives which include:

1. Specialization of a DAS to search and track using the signature of a given Topic of Interest (TOI).
2. Investigate semantic anomalies found in the computation of possibilities that may be caused using information obfuscation techniques.
3. Review of DAS adaptations undertaken by the evolving hypothesis answer structure. When invalid, the DAS learns additional support/rebuttal arguments to prevent the adaptation from re-occurring.

The ALAN DAS lattice is used to explain the information, compute the overall possibilistics, based on the fuzziness of the support, and rebuttal information, and compute the sensitivity of the claim to the fuzziness of the input data. Being able to review the lattice and assess its sensitivity to the fuzziness of the input data enables the user to effectively assess the quality of the lead. Figure 6.2 illustrates the DAS fuzzy possibilistic lattice connections [21].

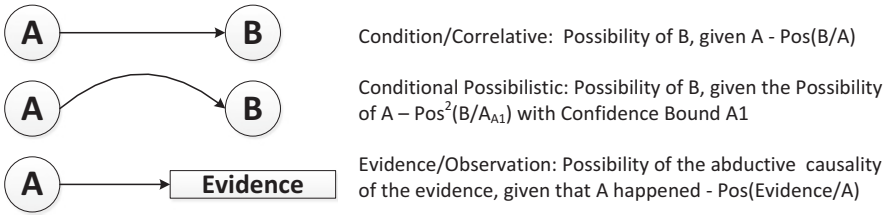


Fig. 6.2 Fuzzy possibilistic lattice connections

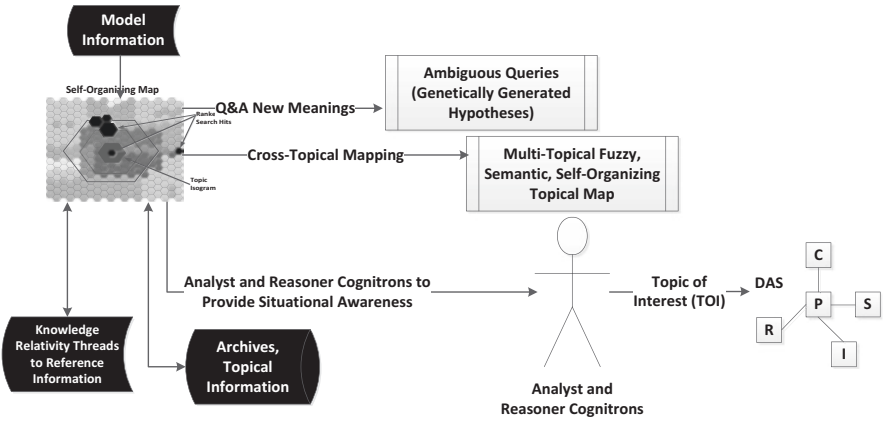


Fig. 6.3 ALAN processing architecture

Utilizing the ALAN cognitive processing environment, the DAS and the Cognitrons mimic human reasoning to process information and develop intelligence. Figure 6.3 illustrates a high-level view of the architecture for the ALAN cognitive processing framework. This process includes Search Information Cognitrons that mine through multiple sources to provide data/information to other Cognitrons throughout the ALAN framework. This is called the Federated Search and is shown in Fig. 6.4.

6.8 Conclusions

The ALAN processing environment allows data to be processes into relevant, actionable knowledge. Based on the technologies described above, situational management is one of the most innovative components of ALAN. Utilizing the cognitive framework within ALAN, it can provide real-time processing and display of dynamic, situational awareness information. Information gathering, processing, and analyzing must be done continually to keep track of current trends in the context of current situations, both local and overall, and provide timely and accurate knowledge within a changing environment to allow systems to anticipate and respond to

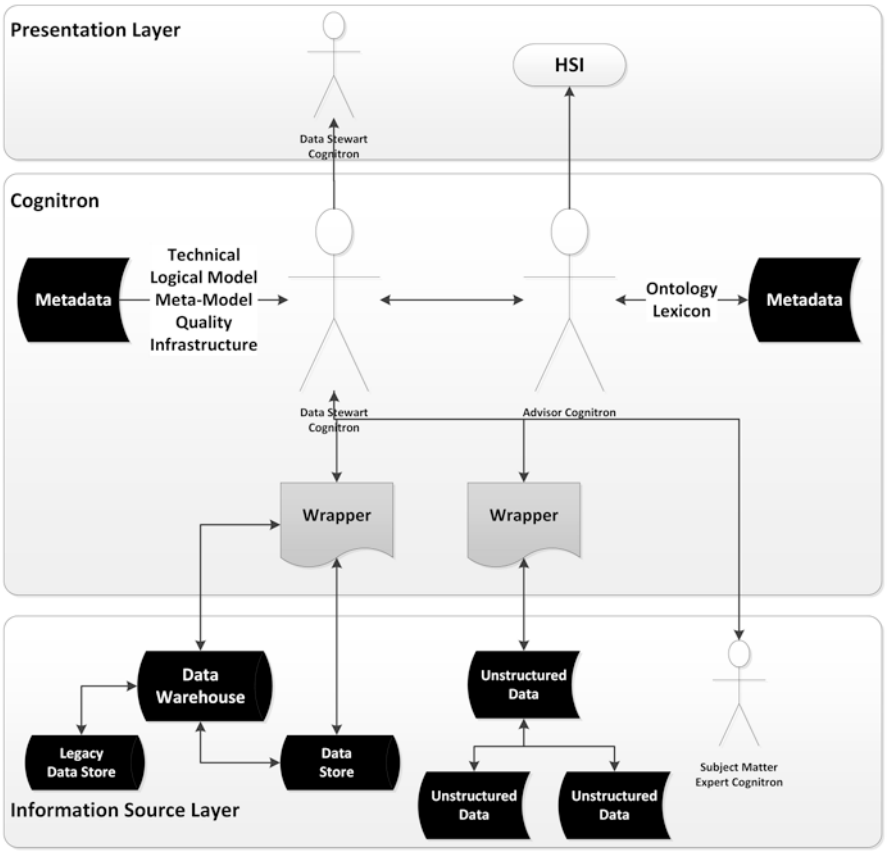


Fig. 6.4 Federated search process within ALAN

new situations. To achieve the combination of awareness, flexibility, and ability means supporting dynamic and flexible processes that adapt as situations and environments change. This is possible with the learning and self-evolving ALAN processing and reasoning framework.

Having a system design that allows an artificial intelligent entity to learn from and react to its environment greatly changes how we think about testing a system. If we are building a lawn mower, we have expectations about exactly what happens when we pull the starting rope. When we build an artificial intelligent system that adapts to its environment, we may not know exactly what to expect when something happens. Our system tests must be carefully architected and how we expect the system to react must be clearly understood. Even then, if the system truly adapts, it may not adapt in the way we expect. That doesn't mean it reacted incorrectly, it just means we did not understand it completely and/or correctly.

References

1. Taylor, J. M., & Raskin, V. (2010). Fuzzy ontology for natural language. In *29th International Conference of the North American Fuzzy Information Processing Society*, Toronto, Ontario, Canada.
2. Fodor, J. (1983). *Modularity and the mind: An essay on faculty psychology*. Cambridge, MA: MIT Press.
3. Garcia, C. (2007). Cognitive modularity, biological modularity, and evolution. *Biological Theory: Integrating Development, Evolution and Cognition, 2*(1), 62–73.
4. Gibbs, R., & Van Orden, G. (2010). Adaptive cognition without massive modularity. *Journal of Language and Cognition, 2*(2), 149–176.
5. Hutchins, E., & Lintem, G. (1995). *Cognition in the wild*. Cambridge, MA: MIT Press.
6. Rogers, Y. (1997). *A brief introduction to distributed cognition*. Brighton: University of Sussex Press.
7. DeRaedt, L. (1992). *Interactive theory revision: An inductive logic programming approach*. Cambridge, MA: Academic Press.
8. Roberts, S., & Tarassenko, L. (1994). A probabilistic resource allocating network for novelty detection. *Neural Computation, 6*, 270–284.
9. Newell, A. (2003). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.
10. Nishimori, T., Nakamura, T., & Shiino, M. (1990). Retrieval of spatio-temporal sequences in asynchronous neural networks. *Physical Review A, 41*, 3346–3354.
11. Cooper, G., & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning, 9*, 309–347.
12. Dimopoulos, Y., & Kakas, A. (1996). Abduction and inductive learning. In L. De Raedt (Ed.), *Advances in inductive logic programming* (pp. 144–171). Amsterdam: IOS Press.
13. Jones, B. (1999). Bounded rationality. *Annual Review of Political Science, 2*, 297–321.
14. Crowder, J., & Friess, S. (2012). Artificial psychology: The psychology of AI. In *Proceedings of the 3rd Annual International Multi-Conference on Informatics and Cybernetics*, Orlando, FL.
15. Bonarini, A. (1997). *Anytime learning and adaptation of structured fuzzy behaviors. Adaptive behavior* (Vol. 5). Cambridge, MA: The MIT Press.
16. Crowder, J. (2012). Possiblistic, abductive neural networks (PANNs) for decision support in autonomous systems: The advanced learning, abductive network (ALAN). In *Proceedings of the 1st International Conference on Robotic Intelligence and Applications*, Gwanju, Korea.
17. Crowder, J. (2010). Flexible object architectures for hybrid neural processing systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
18. Raskin, V., Taylor, J. M., & Hemplemann, C. F. (2010). *Ausocial engineering*. Concord, MA: New Security Paradigms Workshop.
19. Hutchinson, J., Koch, C., Luo, J., & Mead, C. (1988). Computing motion using analog and binary resistive networks. *Computer, 21*(3), 52–63.
20. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
21. Zadeh, L. (2004). A note of web intelligence, world knowledge, and fuzzy logic. *Data and Knowledge Engineering, 50*, 291–304.

Chapter 7
Artificial Intelligent Inferences Utilizing Occam Abduction

7.1 Introduction

As explained in the abstract, Abduction is formally defined as finding the best explanation for a set of observations or inferring cause from effect. The notion of Occam Abduction relates to finding the simplest explanation with respect to inferring cause from effect. A formal definition for Artificial Occam Abduction would be [1]:

Artificial Occam Abduction: The simplest set of consistent assumptions and hypotheses, which, together with available background knowledge, entails adequate description/explanation for a given set of observations [2].

In formal logic notation, given $\boldsymbol{B_D}$, representing current background knowledge of domain $\boldsymbol{D}$, and a set of observations $\boldsymbol{O_D}$, on the problem domain $\boldsymbol{D}$, we look for a set of Occam Hypotheses, $\boldsymbol{H_D}$, such that:

- $\boldsymbol{H_D}$ is consistent[1] w.r.t. $\boldsymbol{B_D}$.
- It holds that $\boldsymbol{B_D}, | = \boldsymbol{H_D}, \rightarrow \boldsymbol{O_D}$.

Abduction consists of computing explanations (hypotheses) from observations. It is a form of non-monotonic reasoning and provides explanations that are consistent with a current state of knowledge and may become less consistent or inconsistent, when new information is gathered. The existence of multiple hypotheses (or explanations) is a general characteristic of abductive reasoning, and the selection of the preferred, or most simple, but possible, explanation is an important precept in Artificial Occam Abduction.

Abduction was originally embraced in Artificial Intelligence work as a non-monotonic reasoning paradigm to overcome inherent limitations in deductive reasoning. It is useful in Artificial Intelligence applications for natural language understanding, default reasoning, knowledge assimilation, belief revision, and very useful in multi-agent systems [3]. The Abduction form of inference, using hypotheses

[1] If $\boldsymbol{H}_{\mathrm{D}}$ contains free variables, $\exists$ ($\boldsymbol{H}_{\mathrm{D}}$) should be consistent w.r.t. $\boldsymbol{B}_{\mathrm{D}}$.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_7

to explain observed phenomena, is a useful and flexible methodology of reasoning on incomplete or uncertain knowledge. Occam Abduction, by the way it is defined here, provides not only an answer, or cause, to the observations, it provides much more information, in that it describes the properties of the class of possible hypotheses in which the observations are valid, and denotes which is the simplest set of hypotheses under which this is true.

Here is where we diverge from classical Abductive Reasoning, which is generally steeped in Bayesian probabilistics. Fuzzy abduction, as opposed to Bayesian reasoning, utilizes fuzzy sets of hypotheses to explain a given set of observations. The Fuzzy Abduction utilized here genetically derives a set of fuzzy hypotheses, using the most appropriate of the available fuzzy implications, and uses these fuzzy hypotheses to derive a truth value (how well do the hypotheses explain the observations). This process is considered abductive because it looks for information that both supports and rebuts the fuzzy hypotheses. The combination of supporting and rebutting arguments is used to determine the "possibility" that each hypothesis explains all or part of the observations. Hypotheses whose possibility is above a given threshold are sent forward either to provide explanations, or as input for the next genetically generated set of hypotheses.

7.2 Elementary Artificial Occam Abduction

There are several distinct types of interactions that are possible between two elementary Occam Abductive hypotheses $h_1, h_2 \in H_e$: [4]

- ***Associativity***: The inclusion of $h_1 \in H_e$ suggests the inclusion of h_2. Such an interaction may arise if there is knowledge of, for instance, mutual information (in a Renyi sense) between h_1 and h_2.
- ***Additivity***: h_1 and h_2 collaborate additively where their abductive and explanatory capabilities overlap. This may happen if h_1 and h_2 each partially explains dome datum $d \in D_0$ but collectively can explain more, if not all of D_0.
- ***Incompatibility***: h_1 and h_2 are mutually incompatible, in that if one of them is included in H_e then the other one should not be included.
- ***Cancellation***: h_1 and h_2 cancel the abductive explanatory capabilities of each other in relation to some $d \in D_0$.
  - For example, h_1 implies an increase in a value, while h_2 implies a decrease in a value. In this case, one is used to support the hypothesis and the other is used to rebut the hypothesis.

The Occam Abductive Process is:

- Nonlinear in the presence of incompatibility relations.
- Non-monotonic in the presence of cancellation relations.
- The general case (nonlinear and non-monotonic) Occam Abduction hypothesis investigation is NP-complete.

Consider a special version of the general problem of synthesizing an Artificial Occam abductive composite hypothesis that is linear, and, therefore, monotonic.

The synthesis is linear if:

$$\forall h_i, h_j \in H_e, \;\; q\left(h_i\right) \cup q\left(h_j\right) = q\left(\left\{h_i, h_j\right\}\right) \tag{7.1}$$

The synthesis is monotonic if:

$$\forall h_i, h_j \in H_e, \;\; q\left(h_i\right) \cup q\left(h_j\right) \subseteq q\left(\left\{h_i, h_j\right\}\right) \tag{7.2}$$

In this special version, we assume that the Occam hypotheses are non-interacting, i.e., each offers a mutually compatible explanation where their coverage provides mutual information (in a Renyi sense). We also assume that the Occam, abductive belief values found by the classification subtasks of abduction for all $h \in H_e$ are equal to 1 (i.e., true).

Under these conditions, the synthesis subtask of Artificial Occam Abduction can be represented by a bipartite graph, consisting of nodes in the set $D_0 \cup H_e$. This says there are not edges between the nodes in D_0, nor are there edges between the nodes in H_e. The edges between the nodes in D_0 and those nodes in H_e can be represented by a matrix $\boldsymbol{Q}$ where the rows correspond to $d \in D_0$ and the columns correspond to $h_t \in H_e$.

The entries in $\boldsymbol{Q}$ are denoted as $\boldsymbol{Q_{ij}}$ and indicate whether the given analyzed data are explained by a specific abductive Occam hypothesis. The entries are defined as:

$$Q_{i,j} = \begin{cases} 0 & \text{if datum } d_i \text{ is not explained by hypothesis } h_j \\ 1 & \text{if datum } d_i \text{ is explained by hypothesis } h_j \end{cases} \tag{7.3}$$

Given the matrix Q for the bipartite graph, the abductive, Occam synthesis subtask can be modeled as a set-covering problem, i.e., finding the minimum number of columns that cover all the rows. This ensures that the composite abductive, Occam hypothesis will explain all of D_0 and therefore be parsimonious.[2]

Now we look at a special linear and monotonic version of the general abductive, Occam hypothesis synthesis subtask and look at a Possibilistic Abductive Neural Networks (PANNs) for solving it [1]. The first is based on an adapted Hopfield model of computation:

$$\forall i = 1, 2, \ldots, n, \quad \sum_{j=1}^{m} Q_{ij} V_j \geq 1 \tag{7.4}$$

For the Occam, abductive synthesis subtask, we associate variable V_j with each Occam hypothesis $h_t \in H_e$, in order to indicate if the Occam hypothesis is included

[2] Note that the general set-covering problem is NP-complete.

in the composite Occam, abductive hypothesis $\boldsymbol{C}$. We then minimize the cardinality of $\boldsymbol{C}$ by:

$$\sum_{j=1}^{m} V_j \tag{7.5}$$

subject to the constraint that all data $d \in D_0$ are completely explained.

For the Occam, abductive network, the term in the energy function that represents the problem constraints must evaluate to zero when the constraint is satisfied and must evaluate to a large positive value when the constraint is not satisfied, forcing the evolving solution lattice to evolve accordingly [5]. For this energy term, we use a term expressed as a sum of expressions, one for each datum element, $\boldsymbol{d_i}$, such that the expression evaluates to zero, when hypothesis $\boldsymbol{h_j}$ that can explain the datum $\boldsymbol{d_i}$ is in the composite hypothesis, i.e., $V_j = 1$. Given that $\boldsymbol{Q}$ is an incidence matrix (with elements either 0 or 1), the expression:

$$\sum_{i=1}^{n}\prod_{j=1}^{m}\left\{\left(1-Q_{ij}\right)+\left(1-V_j\right)\right\} \tag{7.6}$$

satisfies the following conditions:

- Each sum of the product terms can never evaluate to a negative number.
- The sum of the product terms, thus, can never evaluate to a negative number.
- Each product term evaluates to zero when an hypothesis that can explain the datum is in the composite; otherwise, it evaluates to a large value.
- The sum of the product term, thus, evaluates to zero when a composite set of hypotheses can explain all the data.

We derive our Occam abductive energy function as follows:

$$E = \alpha * \sum_{j=1}^{m} V_j + \beta * \sum_{i=1}^{n}\prod_{j=1}^{m}\left\{\left(1-Q_{ij}\right)+\left(1-V_j\right)\right\} \tag{7.7}$$

where α and β are positive constants, and $\beta > \alpha$. The first term represents the cardinality of the Occam hypothesis and the second term represents the penalty for a lack of complete coverage; 0 indicates complete coverage. The self-organizing algorithm for the Occam abductive network is:

Assume data set: $\overline{X} = \left\{X^{(1)},\ldots,X^{(p)},\ldots,X^{(P)}\right\}$ where P is the number of input vectors.

Vector $X^{(p)} = \left\{X_1^{(p)},\ldots,X_i^{(p)},\ldots,X_{n_1}^{(p)}\right\}$ represents the pth input vector to the network.

We initialize STEP and SLOPE $\in$ (0, 0.5).

IT = β = TD = 0.5

For all input vectors $p \in [1\ldots P]$ do {

For all input dimensions $i \in [1\ldots n_1]$ do {

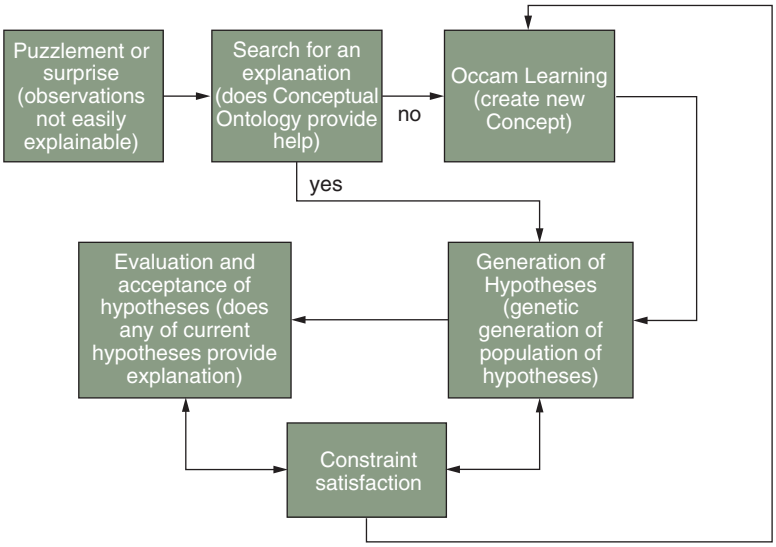


Fig. 7.1 The artificial Occam abduction process

If there are no fuzzy clusters in the *i*th input dimension ($J_i = 0$)
Create a new cluster using $x_i^{(p)}$
Else do {
Find the best fit

Figure 7.1 below illustrates the Occam Abduction Inference process.

7.3 Synthesis of Artificial Occam Abduction

Let $B = \{b_k | k = 1, \ldots, l\}$ *be a finite set of l Occam learned possible beliefs.*

Let $H_e \subseteq H$ *such that for each* $h_j \in H_c$ *can explain some non-empty subset of* D_0.

Let p be a map from H_c *to* H_e:p: $H_e \overset{yields}{\rightarrow} B$.

The map p is also defined from an elementary Occam hypothesis belief value.

We define $p(\{h_j\})$ *as* $p(h_j)$ *and interpret* $p(h_j)$ *as the prima facie Occam belief value for* h_j.

The Occam abductive classification subtask takes D, H, D0, and r as input, where r is a map from $\wp(H)$, *and give* H_e *and p as output*.

The abductive hypothesis synthesis subtask may be characterized as a five-tuple (D_0, H_e, q, p, H_c), *where:*

D_0, H_e, q, and p *constitute the input to the abductive task, and* H_c *is the output of the task.*

Maximal belief in abductive Occam Hypothesis:

A composite hypothesis H_1^c *is a better explanation of* D_0 *than abductive hypothesis* H_2^c *if:*

$$p\left(H_1^C\right) \geq p\left(H_2^C\right) \tag{7.8}$$

This specifies that among the composite dialectic Occam Hypotheses that explain the data, the one with the highest ***belief*** *value is the* ***best*** *explanation by abduction.*

Maximal explanatory coverage of hypothesis data:

A composite hypothesis H_1^c *is a better explanation of* D_0 *than abductive hypothesis* H_2^c *if*:

$$q\left(H_1^c\right) \cap D_0 \supset q\left(H_2^c\right) \cap D_0 \tag{7.9}$$

Ideally, the assembled composite hypothesis H_c *provides adequate explanatory coverage of:*

$$D_0, \quad \text{i.e.,} q\left(H_c\right) \supseteq D_0 \tag{7.10}$$

Minimal hypothesis: A composite abductive Occam hypothesis H_1^c *is a better explanation of* D_0 *than another composite abductive Occam hypothesis* H_2^c *if*

$$\left|H_1^c\right| < \left|H_2^c\right|$$

This condition specifies that H_c *should be parsimonious.*

7.4 Artificial Occam Abduction Hypothesis Evaluation Logic

The following lays out the basics of the Artificial Occam Abductive Logic that will be used to do hypothesis evaluation for the Dialectic Argument Structure, Hypothesis generation, and testing system [6]:

Definition 7.1 A triplet (Φ, Ω, e) defines a domain of Occam hypothesis assembly:

- Φ = The set of hypotheses
- Ω = The set of observations (sensor inputs)
- e = The mapping from the subsets of Φ to the subsets of Ω
- Assumptions:
  - Computational: For every subset, Φ' of Φ, $e(\Phi')$ is computable.
  - Independence: $e(\Phi_1 \cup \Phi_2) = e(\Phi_1) \cup e(\Phi_2)$; for all Φ_1 and Φ_2 that are subsets of Φ.

- Monotonicity: If Φ_1 is a subset of Φ_2, then $e(\Phi_1)$ is a subset of $e(\Phi_2)$.
- Accountability: $\alpha(\varphi)$ is the set of observations that cannot be explained without hypothesis φ.

This drives the four-part Occam Dialectic Argument Structure (DAS) Process:

Screening: Screening determines the acceptability of the possible hypotheses and then allocates them in a hierarchical classification system of fuzzy classifications.

Collection: Collection of hypotheses accounting for the observations. A set of hypotheses is made by adding every hypothesis that explains all or part of the observations.

Parsimony: Parsimony narrows down the collection of hypotheses to the most applicable Occam subset. If a subset of collected hypotheses can explain the observations which is the new (narrowed down) hypothesis set.

Critique: Critique determines which hypotheses are the most essential, among the available ones, based on fuzzy inference metrics. Individually, every hypothesis is excluded from the set, and then the set is tested against the observations. If the observations cannot be explained without the excluded hypothesis, then the excluded hypothesis is marked essential and reintroduced into the set.

Definition 7.2 An Occam abduction system consists of a logical theory "*T*" defined over a domain language "*L*", and a set of domain syntax "*A*" of "*L*" that are called abducible[3] [7].

Definition 7.3 If a set of syntax φ is found as a result of an abductive process in searching for an explanation of ω observations, it must satisfy the following conditions:

- $T \cup \varphi$ is consistent.
- $T \cup \varphi| - \omega$.
- φ is abducible, i.e., $\varphi \in A$.

Definition 7.4 (*C*, *E*, *T*) is a simple causal theory defined over a first-order language "*L*" where "*C*" is a set of causes, "*E*" is a set of effects, and "*T*" is a logical theory defined over "*L*".

Definition 7.5 An *Occam Explanation* of a set of observations, Ω, which is a subset of *E*, is the simplest finite set Φ such that:

- Φ is consistent with *T*.

[3] An abducible argument is a first-order argument consisting of both positive and negative instances of abducible predicates. Abducible predicates are those defined by facts only and the inference engine required to interpret the meaning. In formal logic, abducible refers to incomplete or not completely defined predicates. Problem solving is affected by deriving hypotheses on these abducible predicates as solutions to the problem to be solved (observations to be explained).

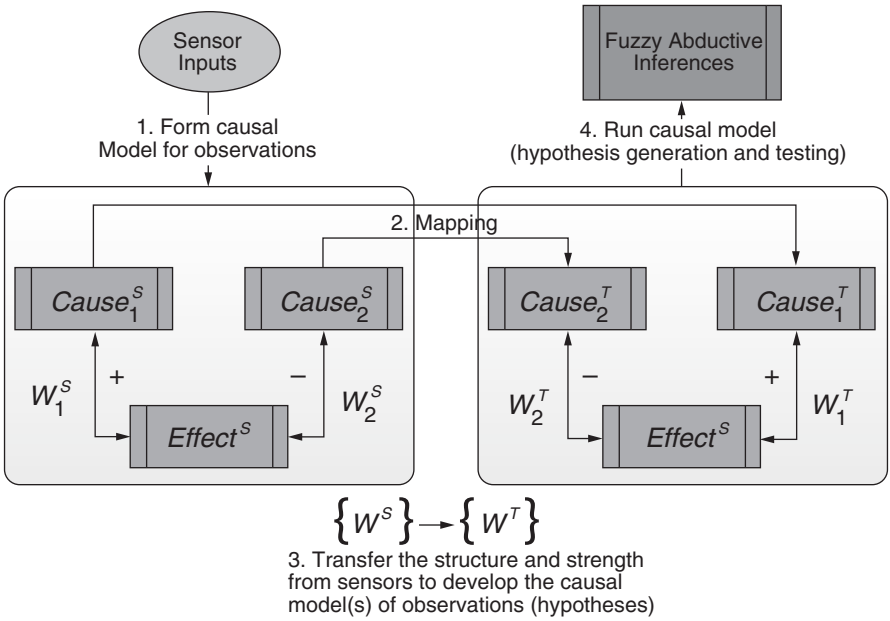


Fig. 7.2 The artificial Occam abduction causal framework

- $T \cup \Phi | - \Omega$, where Ω is the conjunction of all $\omega \in \Omega$.
- Φ is a subset-minimal.

Figure 7.2 illustrates the Artificial Occam Abduction Causal Framework. Figure 7.3 shows the Occam Abduction Fuzzy Inference high-level architecture [8].

An example of the use of Occam Abduction is using inference engine structure shown in Fig. 7.3 within an agent-driven Dialectic Search Argument (reasoning) architecture that utilizes the fuzzy, abductive inferences to find relevant information that then develops a large argument, or inference. The Dialectic Search Argument high-level architecture is illustrated in Fig. 7.4, utilizing the fuzzy, abductive inference engine shown in Fig. 7.3.

The dialectic argument serves two distinct purposes. First, it provides an effective basis for mimicking human reasoning. Second, it provides a means to glean relevant information from Fuzzy, Semantic, Self-Organizing Topical Maps (FSSOTMs) [3] and transform it into actionable intelligence (practical knowledge). These two purposes work together to provide an intelligent system that captures the capability of a human operator to sort through diverse information and find clues.

This approach is considered dialectic in that it does not depend on deductive or inductive logic though these may be included as part of the warrant. Instead, the DSA depends on non-analytic inferences to find new possibilities based upon warrant examples (abductive logic). The DSA is dialectic because its reasoning is based upon what is plausible; the DSA is a hypothesis fabricated from bits of information.

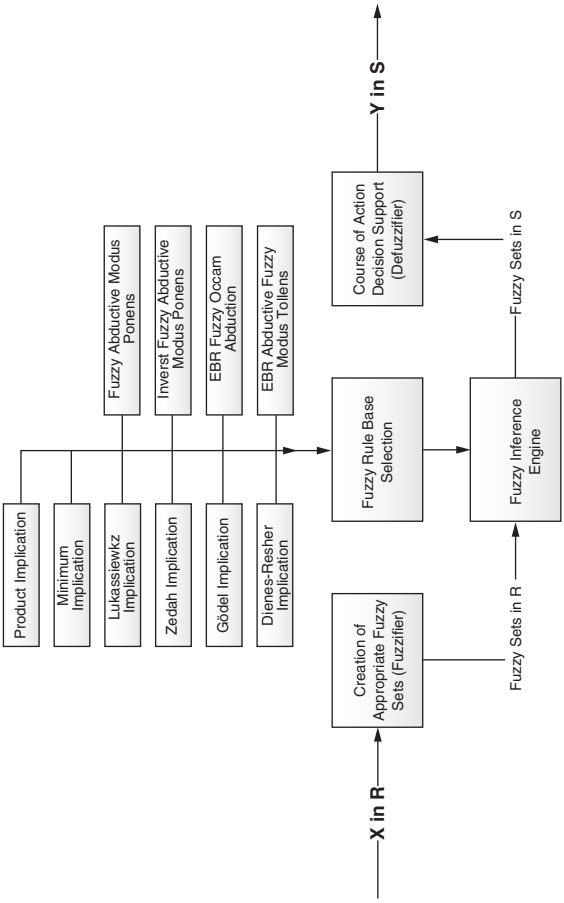


Fig. 7.3 High-level architecture for the fuzzy, abductive inference engine

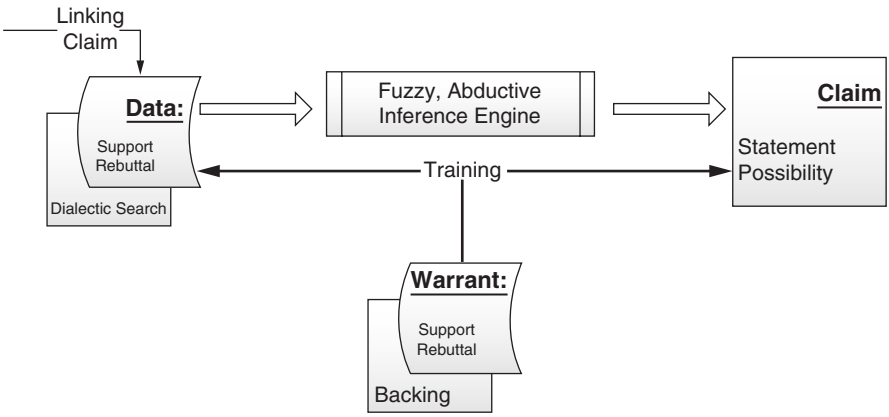


Fig. 7.4 The dialectic argument structure

Once the examples have been used to train the DSA, data that fits the support and rebuttal requirements is used to instantiate a new claim. This claim is then used to invoke one or more new DSAs that perform their searches. The developing lattice forms the reasoning that renders the intelligence lead plausible and enables measurement of the possibility.

As the lattice develops, the aggregate possibility is computed using the fuzzy membership values of the support and rebuttal information. Eventually, a DSA lattice is formed that relates information with its computed possibility. The computation, based on Renyi's entropy theory, uses joint information memberships to generate a robust measure of Possibility, a process that is not achievable using Bayesian methods [2].

7.5 Conclusion

Here, we have laid the foundations for learning structures that will be required for real-time autonomous AI systems. We have provided a mathematical basis for these learning algorithms, based on computational mechanics. The Occam Abduction is but one of many learning constructs that must be present for an AI system to act autonomously and to make sense of a complex world it will find itself a part of [8]. Occam Abduction provides the ability for simple Pattern Discovery that feeds more complex memory and inference systems within an AI cognitive system to allow the autonomous system to think, reason, and evolve. We have but scratched the surface in providing constructs and methodologies required for a self-aware, thinking, reasoning, and fully autonomous real-time AI system.

References

1. Crowder, J. (2012). Possibilistic, abductive neural networks (PANNs) for decision support in autonomous systems: The advanced learning, abductive network (ALAN). In *Proceedings of the 1st International Conference on Robotic Intelligence and Applications,* Gwanju, Korea.
2. Crowder, J., & Carbone, J. (2011). Occam learning through pattern discovery: Computational mechanics in AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence,* Las Vegas, NV.
3. Crowder, J., Scally, L., & Bonato, M. (2012). Applications for intelligent information agents (I2As): Learning agents for autonomous space asset management (LAASAM). In *Proceedings of the International Conference on Artificial Intelligence, ICAI'12,* Las Vegas, NV.
4. Crowder, J., Friess, S., & Carbone, J. (2013). *Artificial cognition architectures*. New York: Springer. ISBN 978-1-4614-8071-6.
5. Crowder, J. (2013). The advanced learning, abductive network (ALAN). In *Proceedings of the AIAA Space 2013 Conference,* San Diego, CA.
6. Dimopoulos, Y., & Kakas, A. (1996). Abduction and inductive learning. In L. De Raedt (Ed.), *Advances in inductive logic programming* (pp. 144–171). Amsterdam: IOS Press.
7. Levesque, H. J. (1989). A knowledge-level account of abduction. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, Detroit, MI (pp. 1061–1067)..
8. O'Rorke, P. (1994). Abduction and explanation-based learning: Case studies in diverse domains. *Computational Intelligence, 10*, 295–330.

Chapter 8
Artificial Neural Diagnostics and Prognostics: Self-Soothing in Cognitive Systems

8.1 Introduction

A critical part of developing and implementing effective diagnostic and prognostic technologies is based on the ability to detect faults in early enough stages to do something useful with the information. Fault isolation and diagnosis uses the detection events as the start of the process for classifying the fault within the system being monitored. Condition and/or failure prognosis then forecasts the remaining useful life (the operating time between detection and an unacceptable level of degradation). If the identified fault affects the life of a critical component, then the failure prognosis models also must reflect this diagnosis [1]. Specific requirements in terms of confidence and severity levels must be identified for diagnosis and prognosis of critical failure modes. In general, the fault diagnosis detection level and accuracy should be specified separately from prognostic accuracy.

As a minimum, the following probabilities should be used to specify fault detection and diagnostic accuracy:

- The possibility of anomaly detection, including false-alarm rate and real fault probability statistics.
- The possibility of specific fault diagnosis classifications using specific confidence bounds and severity predictions.

To specify prognostic accuracy requirements, the developer/end-user must first define [2]:

1. The level of condition degradation beyond which operation is considered unsatisfactory or undesirable to the mission at hand.
2. A minimum amount of warning time to provide the operator and maintainer required information that can be acted on before the failure or condition is encountered.
3. A minimum possibility that remaining useful life will be equal to or greater than the minimum warning level.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_8

We believe that the use of emotional learning [3] and self-soothing techniques from psychotherapy can be utilized within the context of artificial intelligent (AI) systems to radically enhance the ability of system to perform self-diagnosis and prognosis, based on the notion of emotional learning and emotional memories to provide a contextual basis for criticality of faults and system conditions, based on previously learned condition-based system information.

What is described here is a cognitive framework and descriptions of self-soothing concepts that have been adapted to enterprise infrastructures for intelligent systems that will allow advanced prognostics and diagnostics for future AI architectures [4].

8.2 Prognostics and Diagnostics: Integrated System Health Management (ISHM)

A comprehensive health management system philosophy integrates the results from the monitoring sensors all the way through to the reasoning software that provides decision support for optimal use of maintenance resources. A core component of this strategy is based on the ability to [4] accurately predict the onset of impending faults/failures or remaining useful life of critical components and [2] quickly and efficiently isolate the root cause of failures once failure effects have been observed. In this sense, if fault/failure predictions can be made, the allocation of replacement parts or refurbishment actions can be scheduled in an optimal fashion to reduce the overall operational and maintenance logistic footprints. From the fault isolation perspective, maximizing system availability and minimizing downtime through more efficient troubleshooting efforts is the primary objective.

In addition, the diagnostic and prognostic technologies require an integrated maturation environment for assessing and validating Prognostics Health Management (PHM) accuracy at all levels within the system hierarchy. Developing and maintaining such an environment will allow for inaccuracies to be quantified at every level in the system hierarchy and then be assessed automatically up through the health management system architecture. The results reported from the system-level reasoners and decision support are a direct result of the individual results reported from these various levels when propagated through. Hence an approach for assessing the overall PHM system accuracy is to quantify the associated uncertainties at each of the individual levels, as illustrated in Fig. 8.1, and build up the accumulated inaccuracies as information is passed up the system architecture [5].

This type of hierarchical Verification and Validation (V&V) and maturation process [6] will be able to provide the capability to assess diagnostic and prognostic technologies in terms of their ability to detect subsystem faults, diagnose the root cause of the faults, predict the remaining useful life of the faulty component, and assess the decision-support reasoner algorithms. Specific metrics include accuracy, false-alarm rates, reliability, sensitivity, stability, economic cost/benefit, and robustness, just to name a few. Cost-effective implementation of a diagnostic or prognostic system will vary depending on the design maturity and operational/logistics envi-

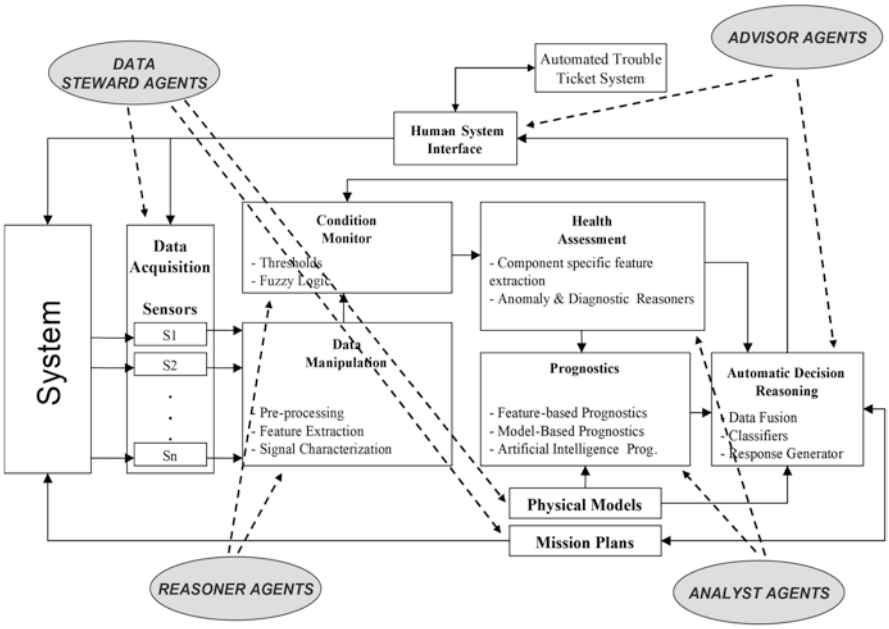


Fig. 8.1 Function layers in the integrated system health management architecture

ronment of the monitored equipment. However, one common element to successful implementation is feedback. As components or Line-Replaceable Units (LRUs) are removed from service, disassembly inspections must be performed to assess the accuracy of the diagnostic and prognostic system decisions. Based on this feedback, system software and warning/alarm limits should be optimized until desired system accuracy and warning intervals are achieved. In addition, selected examples of degraded component parts should be retained for testing that can better define failure progression intervals.

The details of the technologies in Condition-Based Maintenance (CBM) and Prognostic Health Management (PHM) that have been introduced over the recent past by researchers and practitioners are making significant inroads in such application domains as mechanical, thermal, electromechanical, and more recently, electrical and electronics systems. It is well recognized, though, that modern dynamical systems are a tightly coupled composite of both hardware and software. Software reliability is undoubtedly a serious concern and a challenge. We recognize its importance within the general area of reliability and maintainability, and the introduction of AI software architectures into modern systems has made the notion of software ISHM more important than ever [7].

Here we introduce those fundamental system concepts that set the stage for the effective design of fault diagnostic and prognostic technologies though the use of self-diagnostics and self-soothing concepts from psychotherapy. We have reviewed systems-based methodologies within the context of self-diagnosis and self-soothing that have a direct and significant impact on the design and implementation of CBM/

PHM systems. The CBM/PHM designer must be thoroughly familiar with concepts of software and hardware failures associated with modern system architectures (including those with AI software agent structures) and must possess an understanding of methods for the optimal selection of monitoring strategies, algorithms to detect and isolate faults and predict their time evolution, and systems approaches to design of experiments and testing protocols, performances metrics, and means to verify and validate the effectiveness and performance of selected models. A formal framework is established to conduct analysis aimed at comparing alternative options and assisting in the selection of the "best" technologies that meet customer requirements. Software and hardware failure modes and effects criticality analysis forms the foundation for good CBM/PHM design in that they assist in deciding on the severity of failure modes, their frequency of occurrence, and their testability, and provide the foundation for "context-sensitive" emotional learning for advance PHM systems. These new architectures will consider fault symptoms and the required sensor suite to monitor their behavioral patterns. It also may list the candidate diagnostic and prognostic algorithms that are best suited to address specific failure modes. The diagnostic and prognostic process may include recommendations and methodologies for putting the system into a "minimum" capability condition, based on what has been learned from previous conditions. This learned information will be carried in terms of "emotional memories" based on learned situational awareness and criticality, based on current situations combined with system conditions and predicted reliability assessments. These self-diagnostic and self-soothing concepts (which become self-healing in AI systems) will be discussed below.

There are a variety of techniques to address the fault diagnosis problem; however, failure prognosis is the Achilles' heel of CBM/PHM systems. Once an incipient failure or fault is detected and isolated, the task of the prognostic module is to estimate as accurately and precisely as possible the remaining useful life of the failing component/subsystem. Long-term prediction entails large-grain uncertainty that must be represented faithfully and managed appropriately so that reliable, timely, and useful information can be provided to the user. There is the need for robust and viable algorithms but also understanding that enough and complete ground-truth failure data from seeded fault testing or actual operating conditions are lacking. This is a major impediment to the training and validation of the algorithmic developments. The failure prognosis problem basically has been addressed via two fundamental approaches. The first one builds on model-based techniques, where physics-based, statistical probabilistic, and Bayesian estimation methods are used to design fatigue or fault growth models. The second approach relies primarily on the availability of failure data and draws on techniques from the area of computational intelligence, where neural-network, neuro-fuzzy, and other similar constructs are employed to map measurements into fault growth parameters. Here we will discuss AI-related technologies and methodologies for PHM [8].

8.3 Prognostic Technologies

Prognostic Health Management (i.e., prognostics), consists of the ability to monitor and predict failures, detect and classify anomalous events, and assess remaining useful life in electronic systems can provide significant cost benefits, enhanced mission readiness and condition-based maintenance. Once the current health of the component/subsystem/system is determined, it is then necessary to predict what the health of the component/subsystem/system will be sometime in the future and to assess the criticality of this future condition, in terms of the systems current mission and possibility of mission success. The use of emotional memories to help assess the criticality of the current and future predicted system in terms of success can aid the speed at which information is provided and transmitted to coalitions of software agents within the overall systems. This prediction can be for a short time horizon or an estimate of the time till the part needs to be replaced or a failure will occur. There are a variety of issues that need be considered.

The PHM software agents will need to accurately predict into the future. Those predictions will be required to be unbiased and to have a small variance in order to be useful. However, the emotional context of the predictions considering the context of current system parameters can help provide insight into the predictions. The emotional states, in terms of self-soothing, self-diagnosis will be discussed below.

8.4 Abductive Logic and Emotional Reasoners

This type of reasoner employs AI to allow the system to provide explanatory hypotheses, or new ideas, about faults and system performance predictions. This type of reasoner is primarily useful at the subsystem or system level reasoning and not at the component level [9].

The concept of "abduction" is as follows: "Abduction is the process of forming an explanatory hypothesis. It is the only logical operation which introduces any new idea." Abduction can be grasped as a form of logical inference. Abduction consists in examining a mass of facts and in allowing these facts to suggest a theory. In this way we gain new ideas; but there is no force in the reasoning. Deduction or necessary reasoning is applicable only to an ideal state of things, or to a state of things in so far as it may conform to an ideal. It merely gives a new aspect to the premises. Abduction having suggested a theory, we employ deduction to deduce from that ideal theory a promiscuous variety of consequences to the effect that if we perform certain acts, we shall find ourselves confronted with certain experiences. We then proceed to try these experiments, and if the predictions of the theory are verified, we have a proportionate confidence that the experiments that remain to be tried will confirm the theory.

One way to enhance or accelerate the process is to add in the notion of emotional learning into the AI process. By assessing how the system has reacted to situations in the past and the results (i.e., how the system feels about the possible solution),

possible solutions can be assessed quicker within the context of the current problem and situation. One possible way of assessing this is to provide a set of possible solutions and emotional memories to a fuzzy, self-organizing topical map. By assessing the topical solutions and learned emotions about the solution within the context of the mission situation, a rapid assessment of solution "correctness" can be assessed.

Figure 8.2a, b illustrate a Fuzzy, Self-Organizing Topical Map (FSTM) with topic search hits superimposed. The larger hexagons denote information sources that best fit the search criterion, based on retrieved emotional memories about the current and future situations. The isograms denote how close the hits are to a specific situation and solution topics, based on the learned emotions about the contextual problem/solution combinations. There are also other attributes to be explored that would provide significant benefit: as a natural language front end to relational data; and to find information with common emotional meaning [10, 11].

In Fig. 8.2a, the Fuzzy, Semantic, Self-Organizing Map (FSSM) organizes inputs into categories use to encode the inputted information as a histogram. An information map contains contextual information. The information map is self-maintaining and automatically locates inputs. The isograms denote how close the hits are to specific information topics. This FSSM is tied to a Fuzzy Topical Map (FTM) as shown in Fig. 8.2b.

The FSSM can be enhanced to include a topic map. The topic map is the ISO standard for indexing and describing knowledge structures that span multiple sources. The key features are the topics, their associations, and their occurrences in the FSSM. The topics are the areas on the FSSM that fall under a topic name. The associations describe the relationships between topics. The occurrences are the links from the FSSM into the data sources used to form the FTM. The value of superimposing a FTM onto the FSSN is that it defines the information domain's ontology. It also enables rapid and sophisticated dialectic searches.

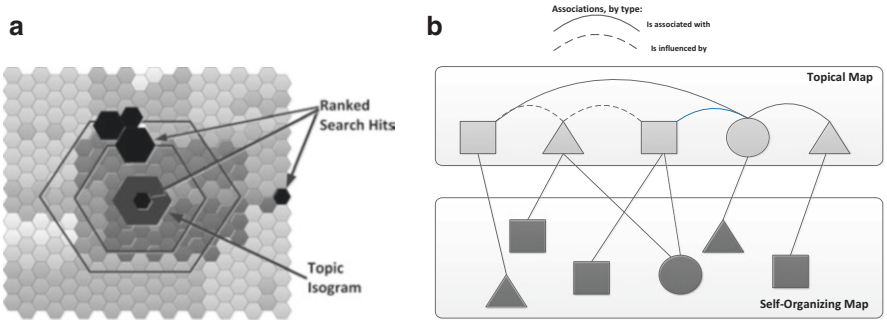


Fig. 8.2 (**a**) Fuzzy, semantic self-organizing map (FSSM), (**b**) FTM in **c**onjunction with the FSSM

8.5 The Dialectic Search

The dialectic search uses the Toulmin argument structure to find and relate information that develops a larger argument, or intelligence lead. The Dialectic Search Argument (DSA), illustrated in Fig. 8.3, has four components [12]:

1. Data and learned emotions: in support of the argument and rebutting the argument.
2. Warrant and Backing: explaining and validating the argument.
3. Claim: defining the argument itself.
4. Fuzzy Inference: relating the data and emotions to the claim.

The argument serves two distinct purposes. First, it provides an effective basis for mimicking human reason. Second, it provides a means to glean relevant information from the topic map and transform it into actionable intelligence (practical knowledge). These two purposes work together to provide an intelligent system that captures the capability of the intelligence operative to sort through diverse information and find clues. This approach is considered dialectic in that it does not depend on deductive or inductive logic, though these may be included as part of the warrant. Instead, the DSA depends on non-analytic inferences and learned emotions to find new possibilities based upon warrant examples. The DSA is dialectic because its reasoning is based upon what is plausible; the DSA is a hypothesis fabricated from bits of information.

Once the examples including learned emotions have been used to train the DSA, data and emotions that fit the support and rebuttal requirements are used to instantiate a new claim. This claim is then used to invoke one or more new DSAs that perform their searches. The developing lattice forms the reasoning that renders the intelligence lead plausible and enables the possibility to be measured. As the lattice develops, the aggregate possibility is computed using the fuzzy membership values of the support and rebuttal information. Eventually, a DSA lattice is formed that relates information with its computed possibility. The computation, based on Renyi's entropy theory, uses joint information memberships to generate a robust measure of possibility, a process that is not possible using other methods [13].

Figure 8.4 illustrates the intelligent software agent architecture used to implement the DSA: three different agents, the coordinator, the Dialectic Argument Search

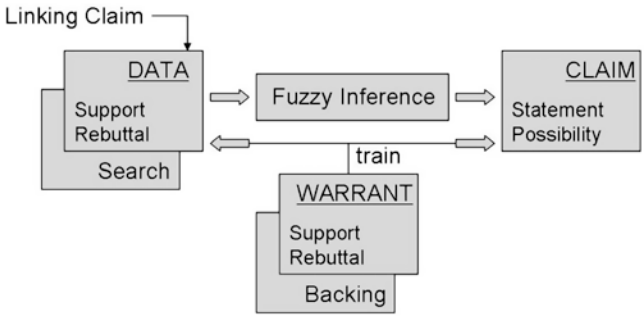


Fig. 8.3 The dialectic search argument structure

(DAS), and the search, work together, each having its own learning objectives. These produce reports based on the self-diagnostic constructs described in the next section.

The coordinator is taught to watch the fuzzy, self-organizing topical map, responding to new hits (input) that conform to patterns and emotions of known interest. When an interesting hit occurs, the coordinator selects one or more candidate DAS agents, and then spawns search agents to find information relevant to each DAS. As time proceeds, the coordinator learns which hit patterns are most likely to yield a promising lead, adapting to any changes in the fuzzy, self-organizing topical map structure and sharing what it learns with other active coordinators [11].

The search agent takes the DAS prototype search vectors and, through the fuzzy, self-organizing topical map, finds information that is relevant and related. The search agent learns to adapt to different and changing source formats and would include parsing procedures required to extract detailed information.

The final agent, the DAS, learns fuzzy patterns and uses this to evaluate information found by the search agent. Any information that does not quite fit is directed to a sandbox where peer agents can exercise a more rigorous aggressive routine to search for alternative hypotheses.

8.6 Self-Soothing in AI Systems

The constructs of self-diagnosis and self-soothing are described here in the context of AI systems. These constructs would be utilized within the fuzzy, self-organizing topical maps and dialectic searches to provide diagnostics and prognostics within the context of mission parameters and situational awareness [14].

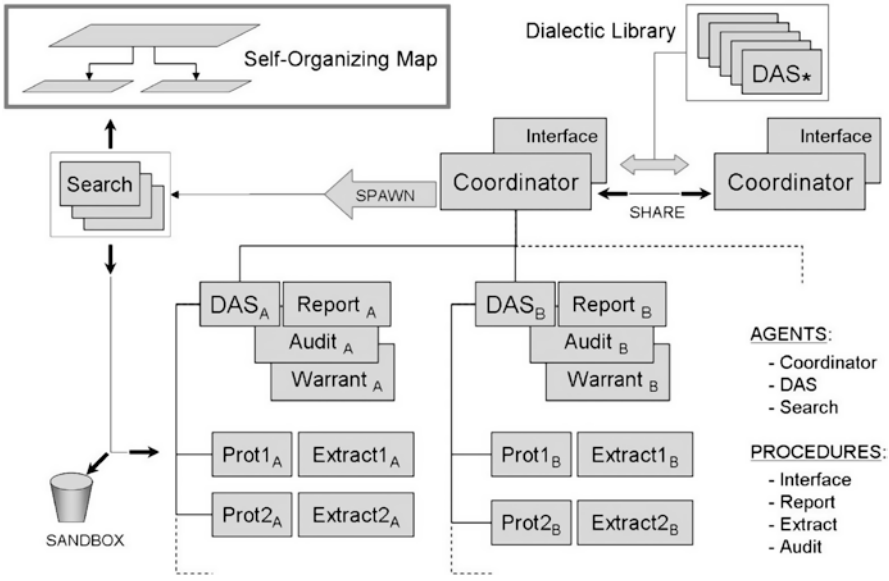


Fig. 8.4 The artificially intelligent DSA software agency

8.6.1 Acupressure

In AI terms, artificial acupressure involves polling all the available resources, refreshing the view of the enterprise infrastructure, basically taping on the AI system. Combining this with retrieving emotional memories involved with the current condition and mission context allows the AI software agents to "calm down" and concentrate on finding solutions to the current problem utilizing genetic algorithms to search (using the fuzzy, self-organizing topical maps and dialectic searches) for solutions; forming the AI system version of an Emotional Freedom Technique (EFT).[1]

8.6.2 Deep Breathing

When you are scared, you might contract your body and hold your breath to try to squish the feelings in order to keep from feeling bad. Pulling your body in tight and stopping your breath keeps you from getting good oxygen to deal with whatever upsets you. In ISHM terms, this is paramount to conservation of resources and not allowing the system to release hardware and software resources that may be required to "heal the current situation."

Deep breathing in AI system terms involves releasing a plethora of intelligent software agents to access all parts of the systems and collaborate in an organized fashion (i.e., breath in and out) and form a collective grouping of possible solutions to the current situation.

8.6.3 Amplification of the Feeling

Exaggeration of feelings in the AI system entails flooding the system with genetic DSA searches with constraints based on an exaggeration of the emotional memories. In fuzzy sense, this is moving from a fuzzy membership function of "greater than" to one of "much greater than," or "much less than" instead of "less than." This allows the system to concentrate on solutions that are the most appropriate and eliminates most "possible" solutions. This acts as the system's subconscious.

8.6.4 Imagery

In our terms, imagery involves creating several "populations" of solutions with a large solution space, opening the mutation and combination rates to allow for major jumps in generational solution possibilities. The topical map measure spaces are relaxed to

[1] Emotional Freedom Technique (EFT) is a form of counseling that draws from acupuncture, neuro-linguistic programming and is often called "energy psychology."

allow a larger "possibilistic" set of solutions to be explored. This helps to jump-start to process and then completely unavailable solutions can be eliminated, but the possible solution sets are broadcast to as wide an agent coalition population as possible. Once a viable solution set has been created, the constraint and rates are returned to normal levels and the solution populations are evaluated. The emotions experienced through this process are catalogued and stored, coupling emotional responses with each solution space. This allows greater efficiencies when the current situation is encountered again.

8.6.5 Mindfulness

Mindfulness is keeping your attention on what is happening in the moment. This would involve tightening constraints and topical distance measurements to ensure that attention is paid to just the problem at hand, once the imagery technique has been utilized. This would employ sorting out only those solutions that carry positive emotional learning responses and assessing those solutions first. These are evaluated by a mediator agents that concentrate on the mission needs and mission criticality to provide necessary solutions that are pertinent to the current situation.

8.6.6 Positive Psychology

Positive psychology researches how happy, successful people work their life. The AI system looks for solution spaces that have resulted in positive emotional responses, based on learned emotions, and utilizes those methods employed during that investigation and diagnostic/prognostic period to look for solutions.

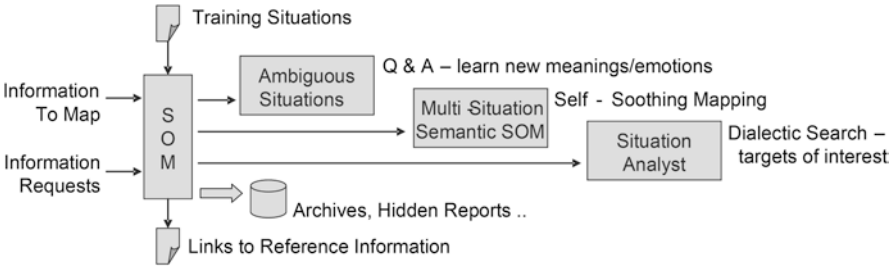


Fig. 8.5 Self-diagnosis/self-soothing architecture

8.7 Artificial Social Intelligence

To facilitate intelligent transmittal of learned emotions and emotional context, Emotional Markup Language (EML) will be utilized within the system for transmittal of emotional information. Figure 8.5 illustrates the self-diagnosis/soothing architecture that will be utilized inside of a multi-agent, cognitive framework.

The combination of these structures provides the overall framework to provide a collective social intelligence within the intelligent information software agent architecture. This is depicted in Fig. 8.6. The major intelligent software agents and their interactions required to facilitate the self-diagnosis/self-soothing and social intelligence constructs are illustrated in Fig. 8.7 [15].

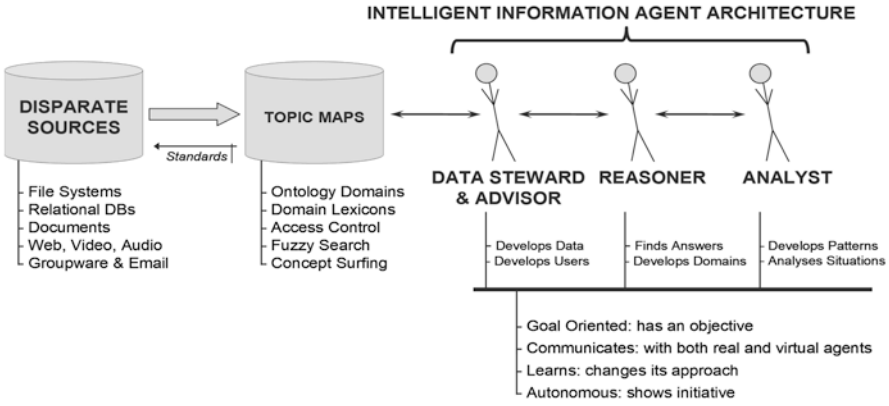


Fig. 8.6 Artificial intelligent social intelligence framework

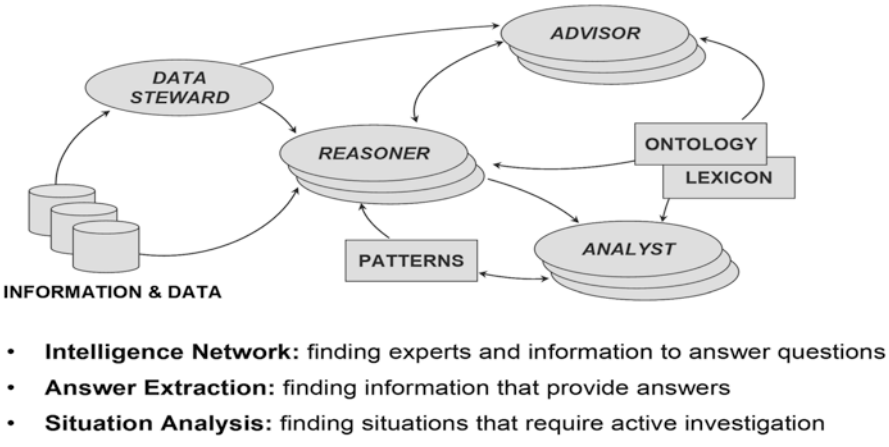


Fig. 8.7 Artificial intelligent information agents (I^2As)

8.8 Conclusions and Discussion

The system and structures described here provide the foundation for utilizing self-diagnosis and self-healing within AI systems and have the possibilities of revolutionizing artificial intelligent systems. We have only begun this investigation and much more work is required to validate the methodologies described here.

References

1. Baresi, L., Ghezzi, C., & Guinea, S. (2006). Towards self-healing compositions of services. In *Contributions to Ubiquitous Computing: Vol. 42: Studies in Computational Intelligence*. Berlin: Springer.
2. Crowder, J. (2010). Operative information software agents (OISA) for intelligence processing. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
3. Cefai, C., Ferrario, E., Cavioni, V., Carter, A., & Grech, T. (2014). Circle time for social and emotional learning in primary school. *Pastoral Care in Education, 32*(2), 116–130.
4. Crowder, J. (2010). Flexible object architectures for hybrid neural processing systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
5. Crowder, J., & Carbone, J. (2011). Occam learning through pattern discovery: Computational mechanics in AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
6. Crowder, J., Carbone, J., & Demijohn, R. (2017). *Multidisciplinary systems engineering: Architecting the design process*. New York: Springer. ISBN 978-3-319-22398-8.
7. Crowder, J., & Carbone, J. (2011). *The great migration: Information to knowledge using cognition-based frameworks*. New York: Springer Science.
8. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
9. Crowder, J., & Friess, S. (2010). Artificial neural diagnostics and prognostics: self-soothing in cognitive systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
10. Crowder, J., & Friess, S. (2010). Artificial neural emotions and emotional memory. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
11. Raskin, V., Taylor, J. M., & Hempelmann, C. F. (2010). *Ontological semantic technology for detecting insider threat and social engineering*. Concord, MA: New Security Paradigms Workshop.
12. Crowder, J., & Friess, S. (2011). The artificial prefrontal cortex: Artificial consciousness. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
13. Crowder, J., & Friess, S. (2011). Metacognition and metamemory concepts for AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*. Las Vegas, NV.
14. Taylor, J. M., & Raskin, V. (2011). Understanding the unknown: Unattested input processing in natural language. In *FUZZ-IEEE Conference*. Taipei, Taiwan.
15. Scally, L., Bonato M., & Crowder, J. (2011). Learning agents for autonomous space asset management. In *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*. Maui, HI.

Chapter 9
Ontology-Based Knowledge Management for Artificial Intelligent Systems

9.1 Introduction

With the ever-increasing availability of sensor data and other intelligence from a variety of collection platforms, it is essential that coherent intelligence fusion and support are provided to the warfighting effort at every level, from commanders down to the individual warfighter. The intelligence fusion and analysis that provides this support needs tools that facilitate processing of the intelligence information as well as its dissemination across the defense intelligence community [1].

By the early 1980s, researchers in artificial intelligence and especially in knowledge representation had realized that work in ontology was relevant to the necessary process of describing the world of intelligent systems to reason about and act within [2]. This awareness and integration grew and spread to other areas until, in the latter half of the final decade of the twentieth century, the term "ontology" became a buzzword, as enterprise modeling, e-commerce, emerging XML metadata standards, and knowledge management, among others, reached the top of many system design requirements [3]. In addition, an emphasis on "knowledge sharing" and interchange has made ontology an application area.

In general, the accepted industrial meaning of "ontology" makes it synonymous with "conceptual model" and is nearly independent of its philosophical antecedents. We make a slight differentiation between these two terms, however (as shown in Fig. 9.1): a conceptual model is an actual implementation of an ontology that has to satisfy the engineering trade-offs and system requirements, while the design of an ontology is independent of run-time considerations, and its only goal is to specify the conceptualization of the world underlying such requirements [4]. In this paper we describe a well-founded methodology for ontological analysis that is strongly based on philosophical underpinnings, and a description-logic-based system that can be used to support this methodology.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_9

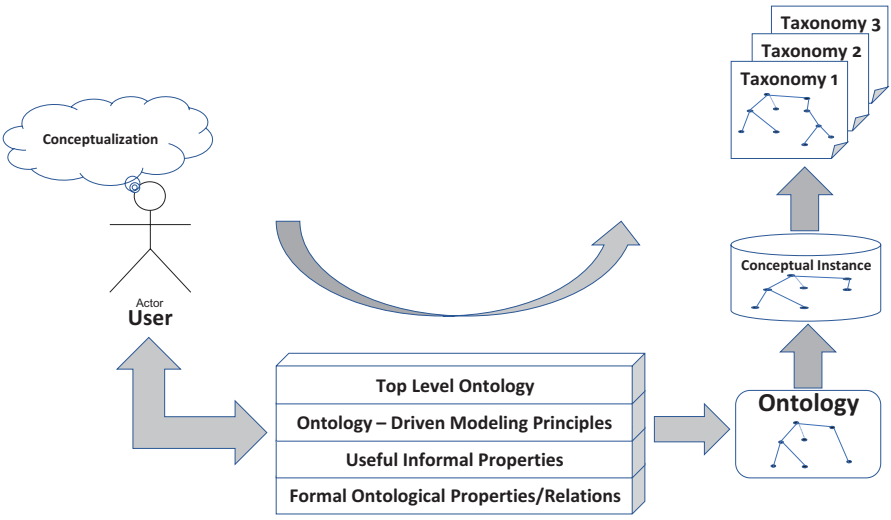


Fig. 9.1 Ontology development methodology

9.2 Taxonomies

Taxonomies are a central part of most conceptual models. Properly structured taxonomies help bring substantial order to elements of a model, are particularly useful in presenting limited views of a model for human interpretation and play a critical role in reuse and integration tasks [5, 6]. Improperly structured taxonomies have the opposite effect, making models confusing and difficult to reuse or integrate. Clearly, insights into how to properly construct a taxonomy are useful. Many previous efforts at providing these insights have focused on the semantics of the taxonomic relationships (also called *is-a, class inclusion, subsumption,* etc.), on different kinds of relations *(generalization, specialization, subset hierarchy)* according to the constraints involved in multiple taxonomic relationships *(covering, partition,* etc.), on the taxonomic relationship in the more general framework of data abstractions, or on structural similarities between descriptions [7, 8].

Our approach differs from many classical approaches, in that we focus on the arguments (i.e., the properties or concepts) involved in the subsumption relationship, rather than on the semantics of the relationship itself. The latter is taken for granted, as we take the statement "ψ subsumes Φ" for arbitrary properties ψ and ϕ to mean that, necessarily [9]:

$$\forall x : \Phi(x) \rightarrow \psi(x) \tag{9.1}$$

Our focus will be on verifying the plausibility and the well-soundness of single statements like (9.1) based on the *ontological nature* of the two properties ψ and Φ. Where, for example, description logics can determine whether one (complex) *description does* subsume another, this methodology can help determine whether a *primitive property can* subsume another, e.g., whether one sensor measurement may subsume another.

9.2.1 Underlying Notions

We begin by introducing the most important philosophical notions: *identity, essence, unity,* and *dependence.* The notion of identity adopted here is based on intuitions about how we, as cognitive agents, in general interact with (and in particular recognize) individual entities in the world around us. Despite its fundamental importance in Philosophy, the notion of identity has been slow in making its way into the practice of conceptual modeling for information systems, where the goals of analyzing and describing the world are ostensibly the same [10]. For the intelligence fusion types of architectures, we will be discussing, the notion of identity is particularly important, because the system must recognize its environment and how to adapt to it when it changes.

The first step in understanding the intuitions behind identity requires considering the distinctions and similarities between *identity* and *unity.* These notions are different, albeit closely related and often confused under a generic notion of identity. Strictly speaking, identity is related to the problem of distinguishing a specific instance of a certain class from other instances of this class by means of a *characteristic property,* which is unique for *it* (that *whole* instance) [11]. Unity, on the other hand, is related to the problem of distinguishing the *parts* of an instance from the rest of the world by means of a *unifying relation* that binds the parts, and only the parts together. For example:

> asking, "Is this the same signal I've seen before?" would be a problem of identity,
>
> whereas asking, "Is this frequency mode consistent with the signal?" would be a problem of unity.

Both notions encounter problems when time is involved. The classical one is that of *identity through change:* in order to account for changing environments, we need to admit that an individual may remain *the same* while exhibiting different properties at different times. But which properties can change, and which must not? And how can we re-identify an instance of a certain property after some time? The former issue leads to the notion of an *essential property,* on which we base the definition of *rigidity,* discussed below, while the latter is related to the distinction between *synchronic* and *diachronic* identity. An extensive analysis of these issues in the context of conceptual modeling has been made elsewhere [12]. These issues become important as we strive to architect and field an automated Signals Intelligence

(SIGINT) processing and fusion system. The notion of when we determine that we are seeing a known signal with a new mode, as oppose to a new type of signal, becomes extremely important and time critical.

The next notion, *ontological dependence,* may involve many different relations such as those existing between persons and their parents, agilities within a signal and the ranges these agilities take, and so on. We focus here on a notion of dependence as applied to properties [13]. We distinguish between *extrinsic* and *intrinsic* properties, according to whether they depend or not on other objects besides their own instances. An intrinsic property is typically something inherent in an individual, not dependent on other individuals, such as having three agile modes. Extrinsic properties are not inherent, and they have a relational nature, like "where the transmitter is at time *t*, or which mode the transmitter is in at time *t*." Some extrinsic properties are assigned by external agents or agencies, such as having a specific location that does not change.

It is important to note that our ontological assumptions related to these notions ultimately depend on our *conceptualization* of the environment in which the system will operate. This means that, while we shall use examples to clarify the notions central to our analysis, the examples themselves will not be the point of this paper. When we say, e.g., that "having the same location" may be considered an identity criterion for *EMITTER A,* we do ***not*** mean to claim this is the universal identity criterion for *EMITTERs,* but that if this were to be taken as an identity criterion in some conceptualization, what would that mean for the property, for its instances, and its relationships to other properties? These decisions are ultimately the result of our notion of the system requirements, the expected signal environments, etc. and again the aim of this methodology is to clarify the formal tools that can both make such assumptions explicit and reveal the logical consequences of them [14].

9.3 Related Database Fundamentals

Identity has many analogies in conceptual modeling for databases, knowledge bases, object-oriented, and classical information systems; however, none of them completely captures the notion we present here [15]. Since any intelligence fusion/processing system will be indelibly tied to a "signal database," we discuss some of these cases below.

Membership Conditions In description logics, the conceptual models usually focus on the sufficient and necessary criteria for class *membership,* i.e., recognizing instances of certain classes [8]. This is not identity, however, as it does not describe how instances of the same class are to be told apart. This is a common confusion that is important to keep clear: membership conditions determine when an entity is an instance of a class, i.e., they can be used to answer the question, "Is that signal from an agile radar used by A10s?" but not, "Is that signal from the agile radar used by the A10 at long. x, lat. y?"

Globally Unique IDs In object-oriented systems, uniquely identifying an object (as a collection of data) is critical, when data are persistent or can be distributed. In databases, *globally unique IDs* have been introduced into most commercial systems to address this issue. These solutions provide a notion of identity for the descriptions, for the units of data (individuals, objects, or records), but not for the entities they describe. It still leaves open the possibility that two (or more) descriptions may refer to the same *entity,* and it is this entity that our notion of identity is concerned with. There is nothing, in other words, preventing two descriptions of the same radar from being created independently at different times/places, and thus having different IDs; the two records describe the same radar, but they are different pieces of data. Globally unique IDs provide identity criteria for database records, but not for the entities in the world the records describe.

Primary Keys Some object-oriented languages provide a facility for overloading or locally defining the equality predicate for a class. In standard database analysis, introducing new tables requires finding unique keys either as single fields or combinations of fields in a record. These two similar notions very closely approach our notion of identity as they do offer evidence toward determining when two descriptions refer to the same entity. There is a very subtle difference, however, which we will attempt to briefly describe here, and which should become clearer with the examples given later.

Primary (and candidate) keys and overloaded equality operators are typically based on *extrinsic properties* that are required by a system to be unique. In many cases, information systems designers add these extrinsic properties simply as an escape from solving (often very difficult) identity problems. Our notion of identity is based mainly on *intrinsic properties*, i.e., we are interested in analyzing the inherent nature of entities and believe this is important for understanding a domain [16]. This is not to say that the former type of analysis never uses intrinsic properties nor that the latter never uses extrinsic ones; it is merely a question of emphasis. Furthermore, our analysis is often based on information which *may not be represented in the implemented system,* whereas the primary key notion can never use such information. For example, we may claim as part of our analysis that people are uniquely identified by their brain, but brains and their possession may not appear in the final system we are designing. Our notion of identity and the notion of primary keys are not incompatible, nor are they disjoint, and in practice conceptual modelers will need both.

9.4 Ontology Analysis

In this section, we shall present a formal analysis of the basic notions discussed in Sect. 9.1, and we shall introduce a set of *meta-properties* that represent the behavior of a property with respect to these notions. Our goal is to show how these

meta-properties impose some constraints on the way subsumption is used to model a domain, and to present a description logic system for checking these constraints.

9.4.1 Preliminary Discussion

Let us assume that we have a first-order language L_0 (the modeling language) whose intended domain is the world to be modeled, and another first-order language L_1 (the meta-language) whose constant symbols are the predicates of L_0. Our meta-properties will be represented by predicate symbols of L_1. Primitive meta-properties will correspond to *axiom schemes* of L_0. When a certain axiom scheme holds in L_0 for a certain property, then the corresponding meta-property holds in L_1. This correspondence is like a system of *reflection rules* between L_0 and L_1, which allow us to define a particular meta-property in our meta-language, avoiding a second-order logical definition. Meta-properties will be used as *analysis tools* to characterize the ontological nature of properties in L_0 and will always be defined with respect to a given conceptualization.

We denote primitive meta-properties by bold letters preceded by the sign "+", "_", or "~", and the notation ϕ^{M} to indicate that the property ϕ has the meta-property M. The reading of each meta-property and its significance will be described later. In our analysis, we adopt first-order logic with identity. This will be occasionally extended to a simple temporal logic, where all predicates are temporally indexed by means of an extra argument. If the time argument is omitted for a certain predicate ϕ, then the predicate is assumed to be time invariant, that is $\exists t : \phi(x,t) \rightarrow \forall t : \phi(x,t)$. Note that the identity relation will be assumed as time invariant: if two things are identical, they are identical forever. This means that Leibniz's rule holds with no exceptions.

We make some use of modal notations such as "necessary" and "possibly" operators which quantify over possible worlds: $\bullet\phi$ means ϕ is necessarily true, i.e., true in all possible worlds, and $\Delta\phi$ means ϕ is possibly true, i.e., true in at least one possible world. Our notion of quantification will be extended such that we will include predicates that are not limited to what exists in the actual world. This is used to consider possible conditions that may exist in the future, or for conditions we may want to test for that we don't currently see but believe them to exist. Worlds will be considered histories rather than snapshots, and we shall consider all of them equally accessible. For instance, a predicate like "Chaos Driven Radar" will not be empty in our world, although no instantiation of it currently exists. We still consider the possibility to be part of the "radar" world. So actual existence is different from existential quantification ("logical existence"), and will be represented by the temporally indexed predicate *E(x, t)*, meaning that *x* has actual existence at time *t*. In order to avoid trivial cases in the meta-property definitions, we shall implicitly assume the property variables are restricted to *discriminating properties*, properties ϕ such that the discriminating properties are properties for which there is possibly something

which has this property, and possibly something that does not have this property, i.e., they are neither tautological nor vacuous. Formally this looks like:

$$\Delta\exists x : \phi(x) \wedge \Delta\exists x \neg\phi(x) \tag{9.2}$$

9.4.2 Knowledge Analysis

In order to capture knowledge representation, we introduce the notion of a "knowledge space" that represents the terminology, concepts, and relationships among those concepts relevant to knowledge management. We illustrate upper and lower ontologies will provide insight into the nature of data/information/knowledge particular to the knowledge management domain. Here we provide views at various levels within the overall knowledge management domain that the multidisciplinary systems engineer might use in the overall system of systems architecture design. The system of systems architecture derived from the knowledge management upper and lower ontologies should include the following:

Controlled Vocabulary (Unified Lexicon) A controlled vocabulary is an ontology that simply lists a set of terms and their definitions. Glossaries and acronym lists are two examples of controlled vocabularies.

Taxonomy Taxonomy is a set of terms that are arranged into a generalization-specialization (parent–child) hierarchy (e.g., communication satellites). Taxonomy may or may not define attributes of these terms, nor does it specify other relationships between the terms.

Relational Databases A database schema defines a set of terms through classes (tables, where terms are represented as the rows in those tables), attributes (specified as columns in the tables), and a limited set of relations between classes (foreign keys).

Software Object Models Object models define a set of concepts and terms through a hierarchy of classes and attributes and a broad set of binary relations among those classes. Some constraints and other behavioral characteristics may be specified through methods on the classes or objects.

Knowledge Representation System A knowledge representation system expresses all of the preceding relationships, models, and diagrams, as well as n-ary relations, a rich set of constraints, rules relevant to usage or related processes, and other differentiators including negation and disjunction.

Knowledge Management Conceptual Architecture The knowledge management upper and lower ontologies should provide a well-designed, component-based

architecture for sharing information in a complex, networked environment. The capabilities provided within each layer should be encapsulated to support a highly distributed system of systems architecture.

Lower Ontologies The knowledge management lower ontology should be described in terms of different aspects the system of systems architecture they address for solving different types of collaboration and integration issues. This is important because it has been demonstrated many times that creating a single, monolithic system of systems element data/information/knowledge object model and files to deliver the business/mission utility that is required. Ontologies are, by nature, much larger and more complex than data object models, and can take significant effort to build. Ontology-based knowledge management is an approach that multidisciplinary systems engineer should investigate. An overall system of systems knowledge management ontology should include the following ontologies:

- **Role Based Ontology**: defines terminology and concepts relevant for an end-user (person or consumer application).
- **Process Ontology**: defines the inputs, outputs, constraints, relations, terms, and sequencing information relevant to a business process or set of processes.
- **Domain Ontology**: defines the terminology and concepts relevant to a particular topic or area of interest (e.g., satellite communication).
- **Interface Ontology**: defines the structure and content restrictions (such as reserved words, units of measure requirements, other format-related restrictions) relevant for a particular interface (e.g., application programming interface (API), database, scripting language, content type, user view, or partial view—as implemented for a portal, for instance).

9.5 Knowledge Management Upper Ontology

Figure 9.2 illustrates an example of an overall knowledge management ontology to manage a system of systems for a satellite processing system. Each object within the upper ontology represents a major area of data/information/knowledge the system must manage. The ontologies and taxonomies illustrated below are intended to be examples and there is no guarantee that all possible entities have been captured. Included are the entities (knowledge objects), but along with each entity are associations between entities, how the entities and associations are indexed throughout the system of systems architecture, and the registries that tag the data/information/knowledge objects with metadata for extraction later. Figure 9.3 illustrates the relationship between the sample of the overall system of systems knowledge space and the data/information/knowledge object associations. For each entity object, there are association objects that provide connections between objects and the metadata affiliated with those associations. Those association connections are accomplished through an analytical engine, based on the domain space of each element of the system of systems.

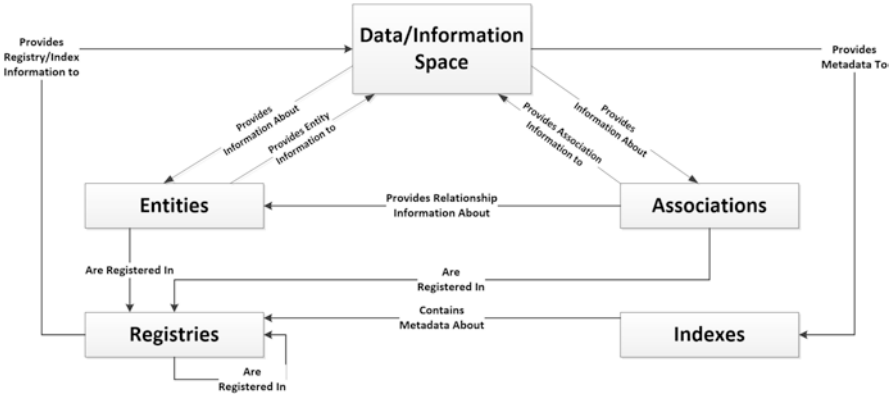


Fig. 9.2 Example knowledge management upper ontology

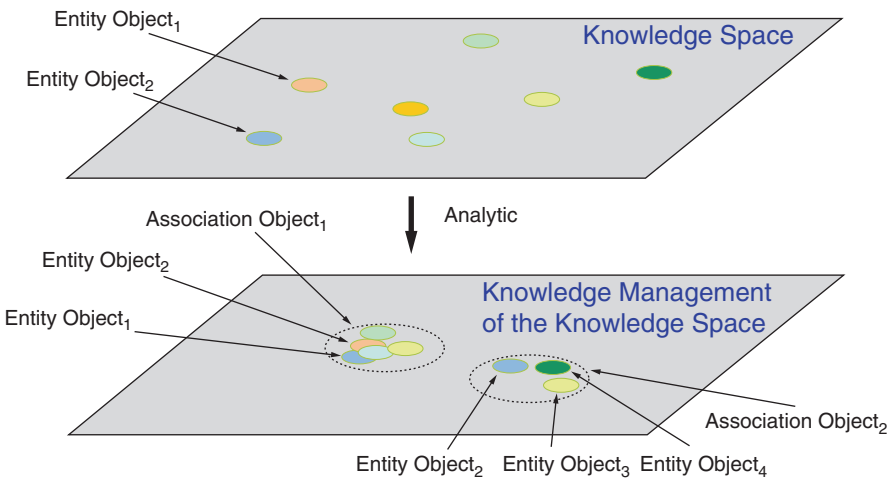


Fig. 9.3 Example artificial intelligence knowledge-space management

The knowledge management process has six levels associated with the process, Level 0–Level 5:

Level 0—Data Refinement: This entails creating a belief system where the different and varying object accuracy/belief semantics can be normalized into a data-specific semantic model that can then be used for data association, tracking, classification, etc.

Level 1—Data/Information Object Refinement: Refinement here consists of data object association, information extraction from data object associations, information object classification, indexing, and registering.

Level 2—Situational Refinement: This includes information object-to-object correlation and the inclusion of all relevant data into an informational display.

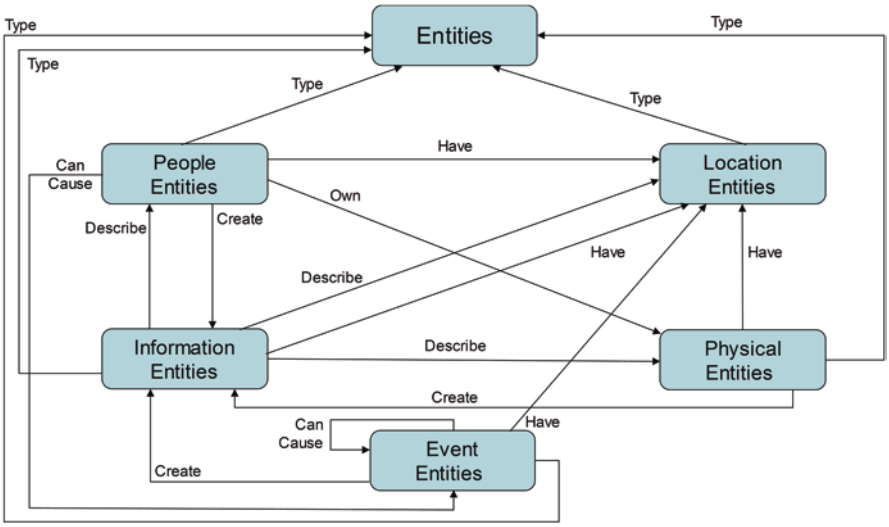


Fig. 9.4 Example artificial intelligence knowledge management lower ontology

Level 3—Knowledge Assessment and Refinement: This consists of collecting the activities of interest, relative to each collection of information objects. Knowledge refinement includes the correlation of situational context and creation of knowledge association metadata, known as knowledge relativity threads [17].

Level 4—Process Refinement and Resource Management: Process refinement enables process control and process management of data/information/knowledge movements through the system of systems, inter- and intra-element communications of data/information/knowledge objects [18]. Resource allocation and management is required at this level for effective movement of data/information/knowledge throughout the system of systems communications infrastructure and involves performance trade-offs.

Level 5—Knowledge, Decisions, and Actions: This level of processing allows information to be incorporated with system experience into system-level knowledge required to provide actionable intelligence and decision support to operators of the system of systems.

Entity Object Lower Ontology Figure 9.4 below illustrates the entity object lower ontology shown in the upper ontology in Fig. 9.2. Here the major categories of data objects are shown, and the associations between them.

Association Object Lower Ontology Figure 9.5 below provides an illustration of an example of the association lower ontology for the knowledge management upper ontology shown in Fig. 9.2. Here, the major categories of association objects are shown along with their interconnectivity.

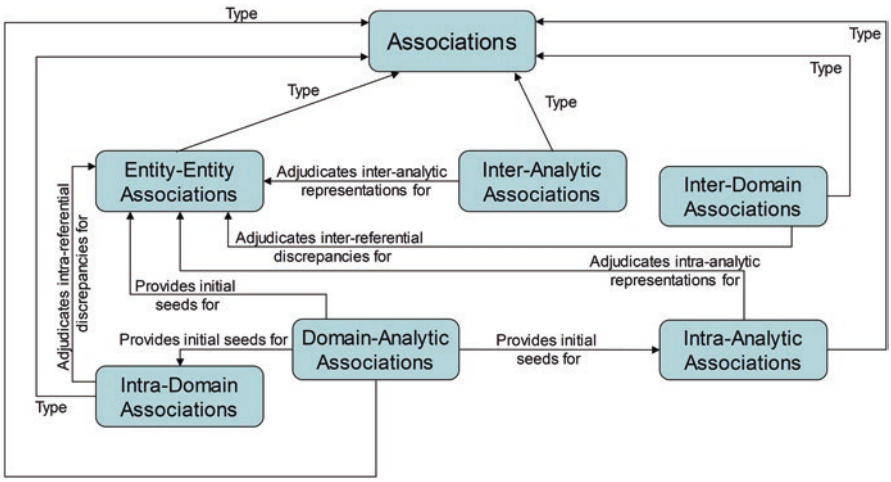


Fig. 9.5 Example artificial intelligence knowledge management lower ontology

Registries Object Lower Ontology Figure 9.6 illustrates a sample registries object lower ontology for the sample knowledge management upper ontology shown in Fig. 9.2. The major categories of registry objects are shown and their interconnectivity.

People Entity Object Taxonomy Figure 9.7 provides a view of the people entity object taxonomy from the knowledge management lower ontology shown in Fig. 9.4. Here the major categories of people entity objects are shown, and the sub-categories associated with them.

Information Entity Object Taxonomy Figure 9.8 illustrates the information entity object taxonomy from the knowledge management lower ontology shown in Fig. 9.4. Here the major categories of location entity objects are shown.

Location Entity Object Taxonomy Figure 9.9 shows an example location entity object taxonomy from the knowledge management lower ontology shown in Fig. 9.4. Here, the major categories of location entity objects are shown.

Event Entity Object Taxonomy Figure 9.10 shows an example event entity object taxonomy derived from the knowledge management lower ontology shown in Fig. 9.4. Here, the major categories of event entity objects are shown.

Physical Entity Object Taxonomy Figure 9.11 illustrates an example physical entity object taxonomy derived from the knowledge management lower ontology shown in Fig. 9.4. Here the major categories of physical entity objects are shown.

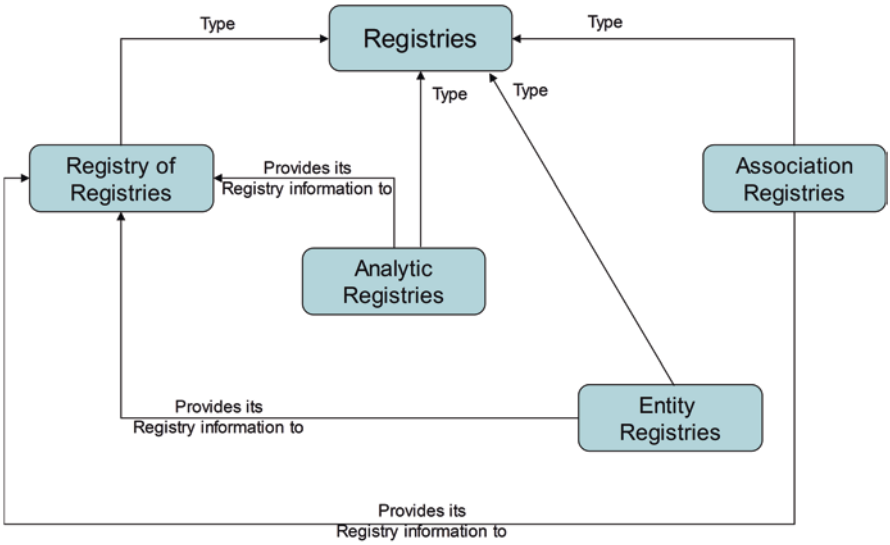


Fig. 9.6 Example artificial intelligence knowledge management registry object lower ontology

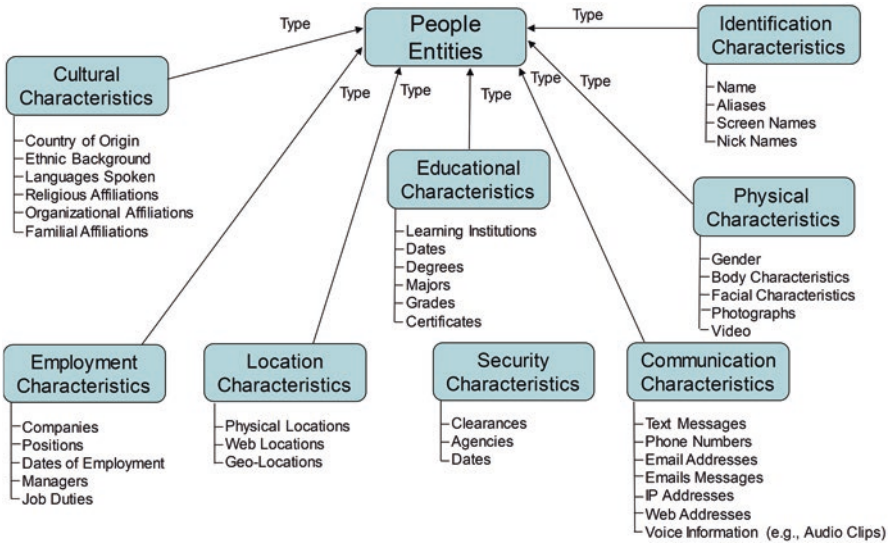


Fig. 9.7 Example artificial intelligence knowledge management people entity taxonomy

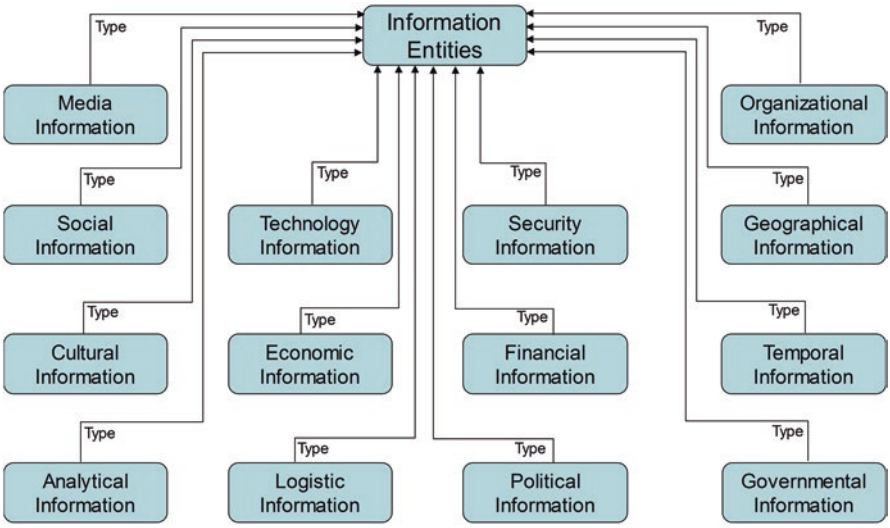


Fig. 9.8 Example artificial intelligence knowledge management information entity taxonomy

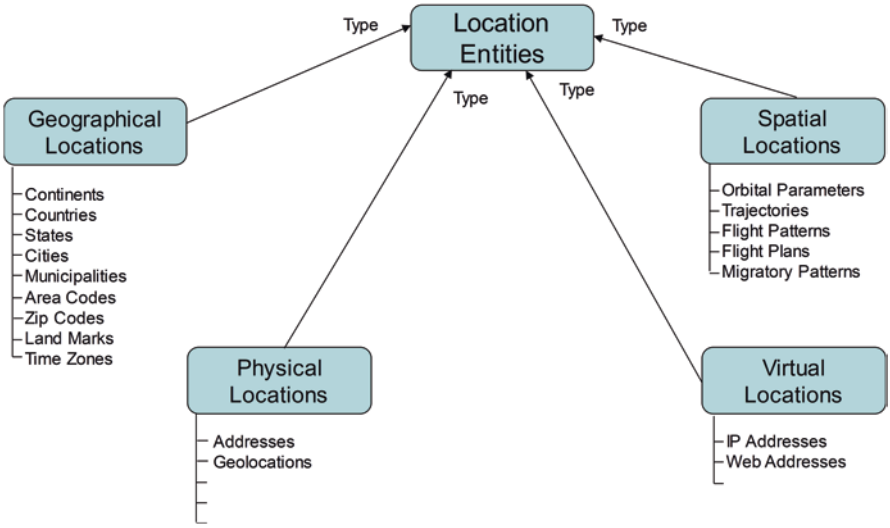


Fig. 9.9 Example artificial intelligence knowledge management location entity taxonomy

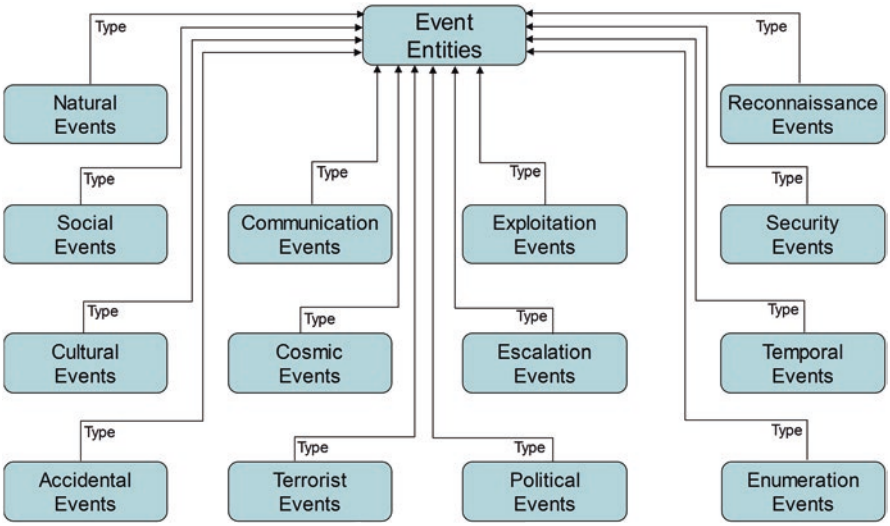


Fig. 9.10 Example artificial intelligent knowledge management event entity taxonomy

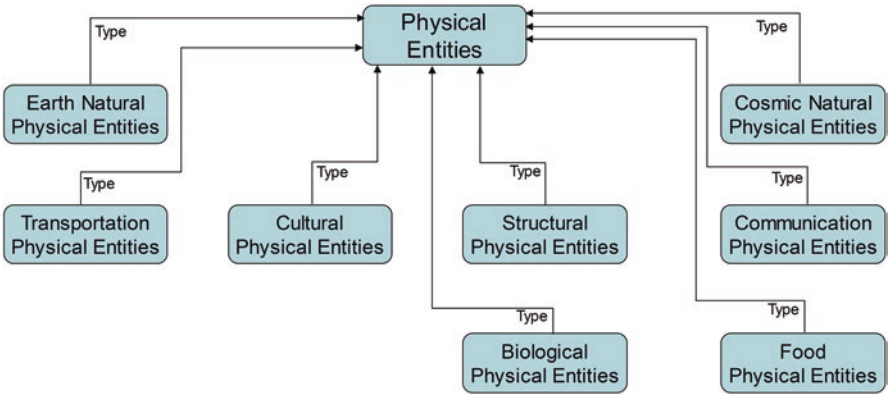


Fig. 9.11 Example artificial intelligence knowledge management physical entity taxonomy

9.6 Upper Services Fault Ontology

In order to properly identify fault messages that must be incorporated into the System of System (SoS) enterprise service architecture with artificial intelligent components/services/algorithms/etc., the infrastructure architecture needs an enterprise service fault ontology. The service faults apply to service-oriented system designs, independent of the technologies used (e.g., web services). Because there are very specific faults associated with services that affect how information flows to artificial intelligent elements of the systems, the service fault ontology is required to

understand what classes and types of faults must be captured and reported by the system of systems enterprise (element-element, subsystem-subsystem, etc., Service Level Agreements (SLAs). If a fault occurs and the artificial intelligent algorithms or systems only receive partial, incomplete, or degraded information, it may radically affect how those systems/components react, learn, infer, and adapt. SoS enterprise service management is a superset of overall system knowledge management and captures the information at the enterprise infrastructure level.

Faults may occur during any and all steps within any given SoS enterprise service. The service management system must be capable of detecting service-related faults and errors. This necessitates the need to identify the fault/error categories that must be detected. The service infrastructure must be capable of handling these faults/errors. Besides service-specific faults, all faults that occur in the distributed SoS enterprise may appear as service faults that will not be picked up by the normal enterprise management systems.

Service implementations represent a troublesome class of problems. Dynamic linking between service providers and consumers (loose coupling) produces dynamic behavior that can cause faults in all five steps within the SoS enterprise:

- Service publishing
- Service discovery
- Service composition
- Service binding
- Service execution

The faults in each step can be caused by a variety of reasons. Table 9.1 illustrates the percentages of fault reasons.

The SoS enterprise service fault upper ontology starts with the five general steps within the SoS enterprise and then each category is refined. This generalization allows a complete coverage of possible service faults. Each service within the SoS will have its own SLA with its own performance objectives, called Service Level Objectives (SLOs). This provides domain-specific faults relative to a domain with the SoS architecture. Figure 9.12 below illustrates an example of a service fault ontology.

Table 9.1 System of system fault error sources

Source of error	Frequency (%)
Requirements	8.1
Features and functionality	16.2
Structural bugs	25.25
Data bugs	22.4
Implementation and coding	9.9
Integration	9.0
System and software architecture	4.7
Test definition and execution	2.8
Other	2.7

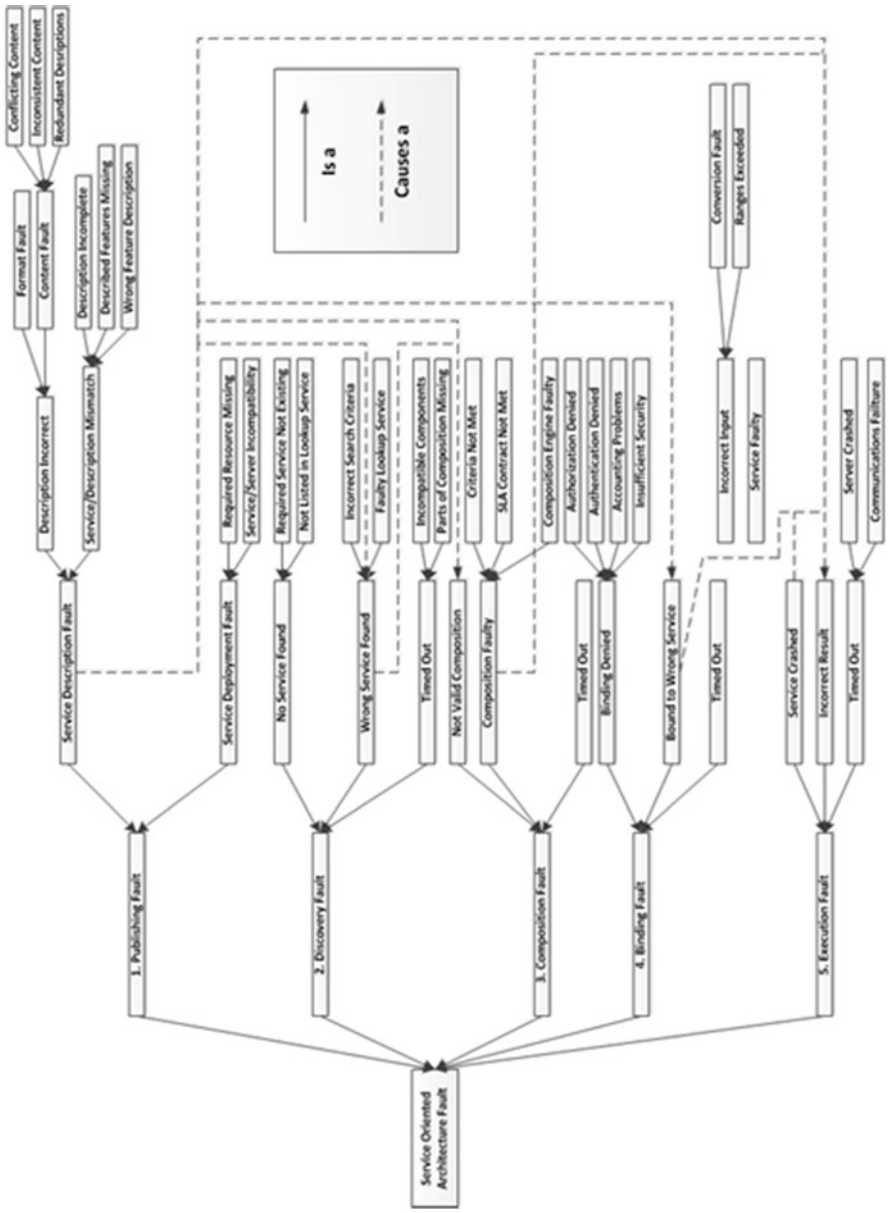


Fig. 9.12 System of systems service fault ontology for artificial intelligent systems

9.7 Example: Technical Publications Taxonomy

Figure 9.13 represents an example of taxonomy, this one for technical publications. Currently, there are several research efforts going on to create "smart" repositories of specific types of documents that provide rapid search, retrieval, and correlation of documents that contain domain-specific information (e.g., spatio-temporal data). These may require non-relational databases to handle the complexities of the search criteria, but to provide at least one example; below is an example of a technical publications object taxonomy that could utilize machine learning algorithms to provide rapid identification, retrieval, and correlation of journal articles. It starts with a technical publications object that is a member of a larger object type, media information object, this is, itself, a member of a larger object type, an information object. The technical publications object has two types, the journal object and the author object. The use of taxonomies is important, as it provides a basic understanding of the components object types that are used within a system. Taxonomies collect objects and their attributes and how they related to each other.

9.8 Knowledge Relativity Threads for Knowledge Context Management

Outlining the need for frameworks which can analyze and process knowledge and context, Liao [17] represented context in a knowledge management framework comprising processes, collection, preprocessing, integration, modeling, and representation, enabling the transition from data, information, and knowledge to new knowledge [19]. The authors also indicated that newly generated knowledge was stored in a context knowledge base and used by a rule-based context knowledge-matching engine to support decision-making activities. Gupta and Govindarajan [18] defined a theoretical knowledge framework and measured the collected increase of knowledge flow out of multinational corporations based upon "knowledge stock" (e.g., the value placed upon the source of knowledge). Pinto [20] developed a conceptual and methodological framework to represent the quality of knowledge found in abstracts. Suh [21] concluded that collaborative frameworks do not provide the contents which go in them, therefore, content was discipline specific, required subject matter experts, and clear decision-making criteria. Additionally, Suh noted that processes promoting positive collaboration and negotiation were required to achieve the best knowledge available and were characterized by process variables and part of what is defined as the process domain. Finally, Ejigu et al. [22] created a framework for knowledge and context which collected and stored knowledge as well as decisions in a knowledge repository that corresponded to a specific context instance. Subsequently, the framework evaluated the knowledge and context via a reasoning engine [23].

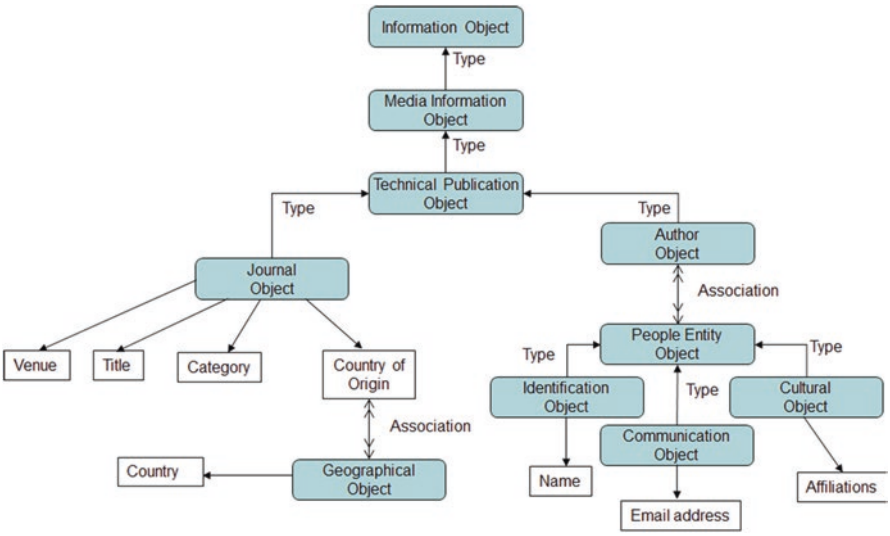


Fig. 9.13 Example journal publication taxonomy suitable for machine learning algorithms

Today, existing databases housing vast bits of information do not store the information content of the reasoning context used to determine their storage [22]. The knowledge collection and storage formula were therefore developed to include and store relationship context along with knowledge, recursively [24]. This means that each act of knowledge and context pairing shown as in equation shown in Fig. 9.1 $\sum_{i,j} K_i(R_j)$ recursively examined all of the previous relationships as they were recombined into storage since they were all related and dependent on each other [25]. Recursive refinement then occurred, per iteration of relationship pairing [26]. Recursive refinement occurred when the user found what was looked for shown as $K_i(R_j)$, using interrogatives (e.g., who, what, when, where, why, and how) [27, 28]. The information content contributing to finding the answer then has significant value and therefore, a higher degree of permanence in the mind of the stakeholder [29]. Therefore, the information content has reached a threshold where retaining the knowledge and context has become important [30].

Figure 9.14 represents a Knowledge Relativity Thread (KRT). Carbone developed this approach for presentation of knowledge and context and was constructed to present five discrete attributes, namely, time, state, relationship distance, relationship value, and event sequence [25]. The goal of a KRT is to map the dependencies of knowledge and related attributes as knowledge is developed from information content. In this figure, the timeline represented by the blue arrow from left to right shows the events or state transitions in sequence and captures the decision points [15]. During each of the iterations of the presentation of knowledge and context, intrinsic values were captured and placed close to each colored knowledge component. In Fig. 9.14, these are represented as information fragments under the cycles. The basic information decomposition depicts how a KRT looks when it represents information decomposed into pieces, in this case fragments. The red triangles,

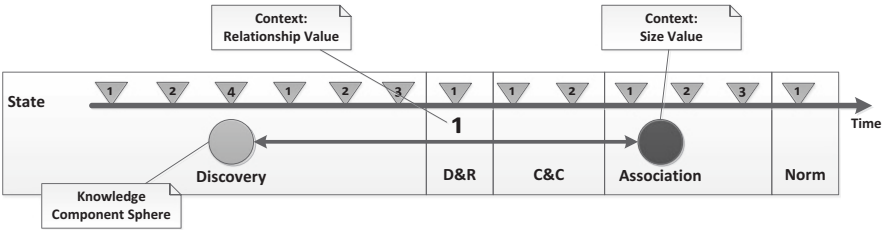


Fig. 9.14 The knowledge relativity thread

added next, depict a state for each of the iterations, in the KRT development cycle. For emphasis, each colored sphere was built into the depiction and added in sequence to represent the fact that each information fragment follows the other. Each icon represents each information fragment. The relative values in this basic knowledge decomposition between each sphere are perceived to be of the same value to each other. Therefore, the lines are the same distance as well. Since this base representation depicted in Fig. 9.13 can present time, state, and sequence, as well as relationships, the challenge was addressed as described by Dourish [12] to create presentation of context which can visually capture and manage a continual renegotiation and redefinition of context as development of knowledge occurs over time. The KRT depicts cognitive comparison of not just information, but of the contextual relationships also. An important distinction about the observation of each comparison is that each is made from the perspective of the aggregated of information, knowledge, and context [29].

The representation of knowledge and context formula is introduced here and is presented by Eq. (9.3). The independent results which follow are mathematical evaluations extended from Newton's law of gravitation shown in Eq. (9.3). Newton's law of gravitation formula is:

$$F = G\frac{\left(M_1 M_2\right)}{r^2} \tag{9.3}$$

where

F is the magnitude of the gravitational force between the two objects with mass.
G is the universal gravitational constant.
M_1 is the mass of the first mass.
M_2 is the mass of the second mass.
r is the distance between the two masses.

This equation was used as an analogy for the derivation of mathematical relationship between a basis, made up of two objects of knowledge [31].

Abstracting Newton's law of gravitation as an analogy of Eq. (9.3), representing relationships between two information fragments, using context, is written as Eq. (9.4) shown below, which describes the components of the formula for representing relationships between information fragments using context [25]:

$$A = B\frac{\left(I_1 I_2\right)}{c^2} \tag{9.4}$$

where

- A is the magnitude of the attractive force between the information fragments.
- B is a balance variable.
- I_1 is the importance measure of the first information fragment.
- I_2 is the importance measure of the second information fragment.
- c is the closeness between the two information fragments.

Comparing the parameters of Eqs. (9.3) and (9.4), F and A have similar connotations except F represents a force between two physical objects of mass M_1 and M_2 and A represents a stakeholder magnitude of attractive force based upon stakeholder determined importance measure factors called I_1, and I_2. As an analogy to F in Eq. (9.3), A's strength or weakness of attraction force was also determined by the magnitude of the value. Hence, the greater the magnitude value, the greater the force of attraction and vice versa. The weighted factors represented the importance of the information fragments to the relationships being formed. The universal gravitational constant G is used to balance gravitational equations based upon the physical units of measurement (e.g., SI units, Planck units). B represents an analogy to G's concept of a balance variable and is referred to as a constant of proportionality. For simplicity, no units of measure were used within Eq. (9.2) and the values for all variables only showed magnitude and don't represent physical properties (e.g., mass, weight) as does G. Therefore, an assumption made here is to set B to the value of 1:

For simplicity, these examples assume the same units and B was assumed to be one. The parameter c in Eq. (9.2) is taken to be analogous to r in Eq. (9.3). Stakeholder perceived context known as closeness c represented how closely two knowledge objects (information fragments) (KO) are related. Lines with arrows are used to present the closeness of the relationships between two pieces of knowledge presented as spheroids (see Fig. 9.15).

Using Eq. (9.4), the value of the attraction force $A_{1\rightarrow 2} = 5 \times 2$ divided by the relative closeness/perceived distance$^2 = 1$. Hence, the attraction force A in either direction was 10. The value of 10 is context which can be interpreted in relation to the scale. The largest possible value for attraction force A with the assumed important measure 1–10 scale is 100, therefore a force of attraction value of 10 was relatively small compared to the maximum. This means that the next stakeholder/researcher understood that a previous stakeholder's conveyance was of small relative overall importance [32]. However, the closeness value of 1 showed that the two objects were very closely related. Figure 9.4 therefore shows that when using Eq. (9.4), if relationship closeness and/or perceived importance measure of the knowledge objects change value, as new knowledge or context is added and evaluated, then it follows that relationship force attraction will change [33].

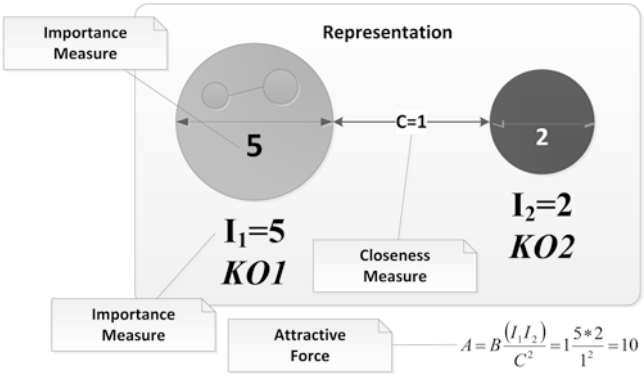


Fig. 9.15 Knowledge object and content for artificial intelligent systems

9.9 Discussion

Knowledge management is more than just categorizing knowledge. Knowledge and information without context are just that, devoid of real content [28]. This is extremely important for an artificial intelligence system. Learning without context can lead to serious decision and inference problems within the system. Instead, the systematic approach presented here, combining the RNA contextual approach, with a cognitive framework, in the artificial intelligence system, allows the framework that can handle cognitive processing of information and context, turning them into actionable intelligence. The use of ontologies and taxonomies within the context of knowledge relativity threads represents the next generation of information analysis and will greatly enhance the capabilities of information processing systems to make sense of increasing volumes multivariate, heterogeneous information [26].

References

1. Newell, J., Shaw, C., & Simon, H. (1957). *Preliminary description of general problem-solving program-i (gps-i)*. Pittsburgh, PA: WP7, Carnegie Institute of Technology.
2. LaBar, K. (2006). Cognitive neuroscience of emotional memory. *Nature Reviews Neuroscience, 7*, 54–64.
3. Damasio, A. (1994). *Descarte's error: Emotion, reason, and the human brain*. New York: Gosset/Putnam.
4. Crowder, J. (2003). *Machine learning: intuition (concept) learning in hybrid genetic/fuzzy/ neural systems*. NSA Technical Paper CON_0013_2003_009.
5. LeDoux, J. (1996). *The emotional brain*. New York: Simon and Schuster.
6. LeDoux, J. (2000). Emotion circuits in the brain. *Annual Review of Neuroscience, 23*, 155–184.
7. Crowder, J. (2002). *Adaptive learning algorithms for functional constraints on an evolving neural network system*. NSA Technical Paper CON_0013_2002_003.
8. Zadeh, L. (2004). A note on web intelligence, world knowledge and fuzzy logic. *Data & Knowledge Engineering, 50*, 291–304.

9. LeDoux, J. (2002). *Synaptic self: How our brains become who we are*. New York: Viking.
10. Crowder, J. (2010). Operative information software agents (OISA) for intelligence processing. In *AIAA Aerospace@Infotech 2010 Conference*.
11. Crowder, J., & Friess, S. (2010). Artificial neural diagnostics and prognostics: Self-soothing in cognitive systems. In *Proceedings of the International Conference on Artificial Intelligence, ICAI'10*.
12. Dourish, J. (2004). What we talk about when we talk about context. *Personal and Ubiquitous Computing, 8*, 19–30.
13. Levine, P. (1997). *Walking the tiger: Healing trauma*. Berkeley, CA: North Atlantic Books.
14. Rowley, J., & Hartley, R. (2008). *Organizing knowledge: An introduction to managing access to information*. Farnham: Ashgate.
15. Kosko, G. (2004). Fuzzy cognitive maps. *International Journal of Man-Machine Studies, 24*, 65–75.
16. Marsella, S., & Gratch, J. (2002). A step towards irrationality: Using emotion to change belief. In *1st International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy.
17. Liao, S., He, J., & Tang, T. (2004). A framework for context information management. *Journal of Information Science, 30*, 528–539.
18. Gupta, K., & Govindarajan, V. (2000). Knowledge flows within multinational corporations. *Strategic Management Journal, 21*, 473–496.
19. Miller, E., Freedman, D., & August, W. J. (2002). The prefrontal cortex: Categories, concepts and cognition. *Philosophical Transactions of the Royal Society B Biological Sciences, 357*(1424), 1123–1136.
20. Pinto, M. (2006). A grounded theory on abstracts quality: Weighting variables and attributes. *Scientometrics, 69*, 213–226.
21. Suh, N. (2006). Application of axiomatic design to engineering collaboration and negotiation. In *4th International Conference on Axiomatic Design*, Firenze.
22. Ejigu, D., Scuturici, M., & Brunie, L. (2008). Hybrid approach to collaborative context-aware service platform for pervasive computing. *Journal of Computers, 3*, 40.
23. DeYoung, C., Hirsh, J., Shane, M., Papademetris, X., Rajeevan, N., & Gray, J. (2010). Testing predictions from personality neuroscience. *Psychological Science, 21*(6), 820–828.
24. Gruber, T. (2008). Collective knowledge systems: Where the social web meets the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web, 6*, 4–13.
25. Carbone, J. N. (2010). *A framework for enhancing transdisciplinary research knowledge*. Lubbock, TX: Tech University Press.
26. Eichenbaum, H. (2002). *The cognitive neuroscience of memory*. New York: Oxford University Press.
27. Hong, J., & Landay, J. (2001). An infrastructure approach to context-aware computing. *Human-Computer Interaction, 16*, 287–303.
28. Howard, N., & Qusaibaty, A. (2004). Network-centric information policy. In *Second International Conference on Informatics and Systems*.
29. Anderson, J. (2004). *Cognitive psychology and its implications*. New York: Worth.
30. Newell, A. (2003). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.
31. Crowder, J., & Carbone, J. (2011). *The great migration: Information to knowledge using cognition-based frameworks*. New York: Springer Science.
32. Crowder, J., & Friess, S. (2010). Artificial neural emotions and emotional memory. In *International Conference on Artificial Intelligence, ICAI'10*.
33. Crowder, J. (2010). Flexible object architectures for hybrid neural processing systems. In *International Conference on Artificial Intelligence, ICAI'10*.

Chapter 10
Cognitive Control of Self-Evolving Life Forms (SELF) Utilizing Artificial Procedural Memories

10.1 Introduction

To examine plausibility and feasibility of self-evolving life forms, we have undertaken a series of experiments over the past 10 years to develop and test small, artificially intelligent, cybernetic entities with varying abilities to think, learn, and self-evolve, at low brain functional levels. These artificial life forms were created to learn and act like insects, with rudimentary cognitive functions to establish whether artificial cognitive architectures are realizable at a most simplistic level. The current second instantiation, of these cybernetic insects, which we have named "Zeus," after the Greek god, whose name means "living one," utilizes a simplistic analog neural network for information transfer throughout his internal effector network and utilizes low-level digital cognitive framework that affects learning and self-evolving. Controllers were utilized containing EEPROM, RAM, and flash memory in order to facilitate learning and storage of learned behavior in as low a Size, Weight, and Power (SWaP) footprint as possible[1]. Basic effector control commands are stored in EEPROM. As Zeus learns, information is stored in RAM until determining that a behavior has been "adequately" learned, and then stored as a series of commands (procedural memory) into flash memory.

10.2 Analog Neural Structures

For his analog neural network, we utilize an adaptation of the information continuum equation [1], which is shown below:

[1] AVR ATTINY24 and ATTINY44 Microcontrollers are used, with the ATTINY24 as the baseline.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_10

$$C\frac{du(x,y,t)}{dt} = -\frac{1}{R}u(x,y,t) + \int_x \int_y w(x,y)z(x,y,t)dxdy + I(x,y,t) \quad (10.1)$$

where

C represents the capacity of the across the network for interactions with node u,
$1/R$ represents the decay rate for node u,
I represents the processing activity for node u,
u represents a unit node of the system,
x represents the preprocessed input to node u,
y represents the output from node u,
z represents the learning functionality for node u,
w represents the relative contextual information threads and association weight of u with its surrounding nodes, including a decay factor for each relative information thread that describes the relative contextual decay over time, where:

$$w = \sum_{j=1}^{M} \frac{1}{r_j} T_j \mathrm{KD}_j W_j \quad (10.2)$$

where

T represents the contextual information thread j derived from fuzzy, self-organizing contextual topical maps [2].
KD *represents knowledge density j of information thread T.*
W represents weighting for contextual thread j.

$$\sum_j W_j = 1 \quad (10.3)$$

The adaptation, shown below [3], describes the dynamic equation for Zeus' analog neurons [4].

$$C_i \dot{u}_i(t') = -\frac{1}{R_i}u_i(t') + \sum_{j=1}^{N} T'_{ij} f_j\left(u_j\left(t' - \tau'_j\right)\right) \quad i = 1,\ldots,N \quad (10.4)$$

where
C_i is the ith neuron analog input capacitance
$1/R_i$ represents the decay rate for node u_i, and
R_i is the resistance to the rest of the analog neural network at the input to neuron i, and for equation 10.5:

$$\left(\sum_j \left[T_{ij}^{'}\right]\right)^{-1} \tag{10.5}$$

f_i is the transfer function of the *i*th neuron

Within Zeus' analog neural network, the digital cognitive system monitors the strength of the analog neurons to determine when the strength of learning has progressed to the point where the learning should be "committed to memory" within the digital cognitive system (RAM). Polyn and Kahana [5] suggest that recall of known item representations is driven by an internally maintained context representation. They described how neural investigations had shown that the recall of an item represented in the mind is driven by an internally maintained context representation that was previously integrated information with a time scale. Therefore, when a series of analog neurons is sufficiently strengthened over time, and have been committed to digital memory, such that they create a series of commands or learned behaviors that can be considered a "procedural memory," these are stored in flash memory with a tag that corresponds to the learned activity or behavior (e.g., turn left). The next time Zeus' sensors relate information such that he needs to move left, this procedure is recalled from memory and activated; meaning, he doesn't have to think about how to turn left, he turns left automatically. This is analogous to instinctive driving of a car after we have learned to drive, however, at a much lower cognitive level than a human but enough to allow Zeus to learn and evolve.

10.3 Self-Evolution Utilizing Procedural Memories

With Zeus, the goal was to add cognitive skills one at a time, perform tests, and determine whether he was able to integrate these together within his limited cognitive framework. Once Zeus has reached a significant cognitive skill level, a new cybernetic "bug" artificial life form will be implanted with Zeus' cognitive skills, present at the beginning of activation. We will determine whether the new artificial life form has an easier or more difficult time integrating the cognitive skills together, and whether the skill sets developed one at a time and at the same cognitive level as Zeus. We expect to also gain valuable insight into artificial life form initialization and sequencing.

Zeus was first initialized into existence in early September 2012. He learned to walk, turn, integrate his sensors, plan his movements, and execute his plans, hence, demonstrating autonomous planning, sensory integration, and autonomous decision-making, none of which is specifically part of his initial programming. He was only enabled with the skills to learn, think, store, and recall memories, provided to him initially. He now carries 25 different procedural memories in the form of a series of commands learned for a particular action.

10.4 Test Scenarios

Zeus has objective functions he endeavors to drive to zero that are part of his basic "instincts." He must learn to use his available sensors and effectors to reduce these objective functions. The set of learning tests that have been and will be performed on Zeus to test his analog and digital neural networks and cognitive algorithms are:

1. **Learning to walk**: In his initial state, Zeus understands the concept of movement, but does not have the knowledge of how to walk. He must learn to move his effectors in order to move effectively. First, initially learning to walk using his six legs, and then to turn left and right.
2. **Learning to find darkness**: One of Zeus' objective functions is "fear of light." He must learn to use his light sensors and compute the differential between the two sensors to establish the direction of movement required to lower the objective function.
3. **Power Regeneration**: Another objective function to be added is the notion of a "hunger instinct," which to Zeus means low power. He carries solar cells to charge up, or feed. He must learn to balance the objective functions for hunger (find light and charge) vs. the objective function for fear of light (find darkness).
4. **Proximity Sensing**: Zeus carries infrared transmitter/receivers to sense when he is close to another object. One of his objective functions is to avoid closeness to other objects (receiving reflected infrared raises the objective function). He must balance (1) the need to find darkness, which may be found under other objects, (2) the need to not get too close to objects, and (3) the need to get close to a light source to "feed."

Understanding how Zeus manages and balances these objective functions and how procedural memories are created, stored, and recalled could provide valuable insight into cognitive control of autonomous life forms for use in autonomous deep sea, space, or land-based applications.

10.5 Procedural Implicit Memory

Procedural implicit memories allow previously learned tasks to be performed without specific "conscious" memory recall/reconstruction of how to perform the task [6]. Procedural memories tend to be inflexible, in that they are tied to the task being performed. For example, when we decide to ride a bike, we don't unconsciously recall/reconstruct memories of how to drive a car, we recall/reconstruct unconscious procedural memories of how to ride a bike. In Zeus, tasks that are learned and are deemed "important" to capture for future use will have procedural memories stored as steps, or "procedures" that are required to perform the same task in the future. In his work on procedural memory and contextual representation, Kahana showed that retrieval of implicit procedural memories is a cue-dependent process that contains both

semantic and temporal components [5]. Creation of procedural memories is tied not only to task repetition but also to the richness of the semantic association structure [7].

Earlier work by Crowder, built on Landauer's procedural memory computational models and Griffith's topical models [8], theorized about the creation of artificial cognitive procedural memory models based on knowledge relativity threads [9] to create the semantic associations [10] and work in fuzzy, self-organizing, semantic topical maps [1, 11] counted on the topical model needed to create long-term procedural memories. These knowledge relativity thread based models were derived from combining cognitive psychological, space-time, strong, weak, and quantum mechanical concepts, along with topical maps which are based on early work by Zadeh. Zadeh [12] described tacit knowledge as worldly knowledge that humans retain from experiences and education. He concluded that current search engines, even with their remarkable capabilities, did not have the capability of deduction, that is the capability to synthesize answers from bodies of information which reside in various parts of a knowledge base. More specifically, Zadeh describes fuzzy logic as a formalization of human capabilities: the capability to converse, reason, and make rational decisions in an environment of imprecision, uncertainty, and incompleteness of information. In their work in cognition frameworks, Crowder and Carbone [13, 14] also expand on the work of Tanik [15] in describing artificial procedural memories as procedural knowledge gained through cognitive insights, based on fuzzy correlations.

10.6 Creation and Retrieval of Artificial Procedural Memories

Continued investigation, utilizing the work of Kahana [5] in associative episodic memories [16], led to the development of a cognitive perceptron framework for creating, storing, and retrieving artificial implicit memories [6, 17] (see Fig. 5.7). Based upon this work, a systems and software architecture specification has been developed for an artificial cognitive framework utilizing the cognitive perceptrons [18].

The main hypothesis here is that the procedural memory scripts can be detected and acquired with the combination of rule-based computational semantic techniques enhancing the artificial life form's understanding of repeatable and useful procedures. The objectives of utilizing artificial procedural memories for cognitive control of artificial life forms are to:

1. Identify potential procedural memories utilizing a combination of rule-based techniques, combined with machine learning techniques.
2. Develop the principles of comparison and comprehension of commands required for creation of procedural memories (see Fig. 10.1).

Crowder, in conjunction with Carbone and Friess, in researching artificial neural memory frameworks that mimic human memories, are creating computer architec-

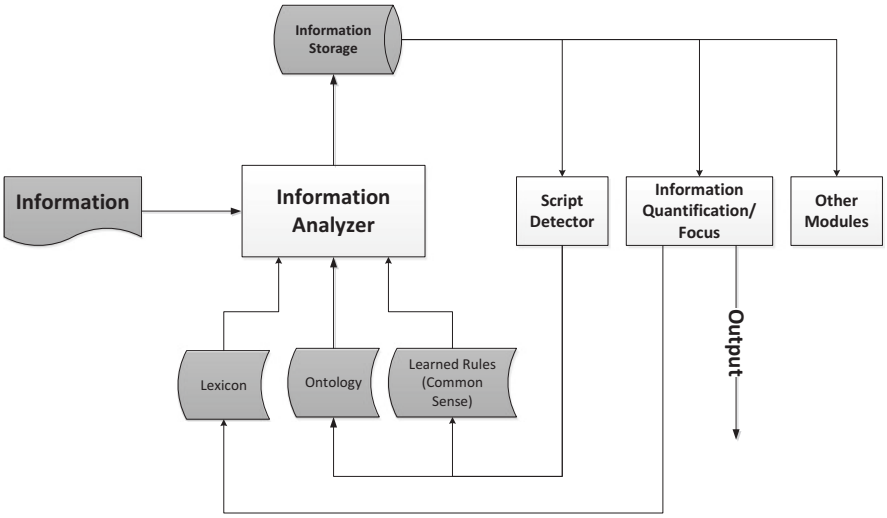


Fig. 10.1 Artificial procedural memory creation

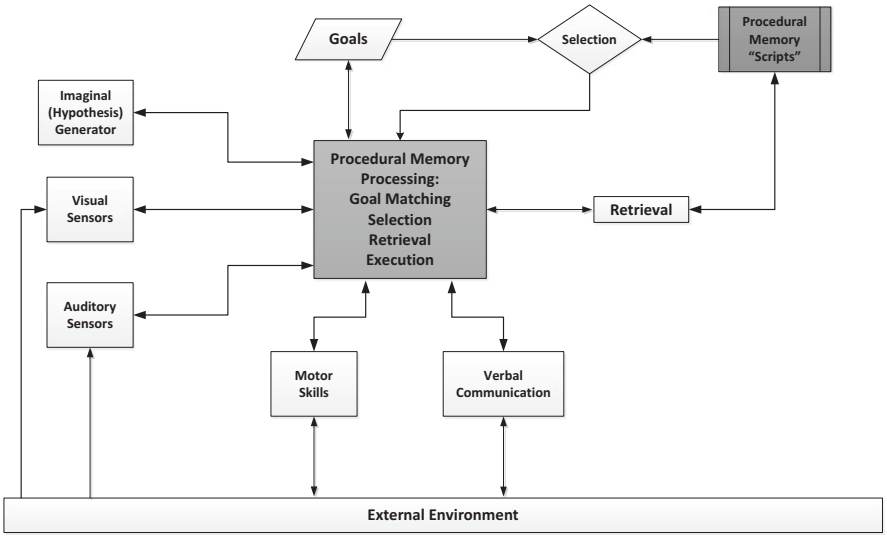


Fig. 10.2 Artificial procedural memory retrieval

tures that can take advantage of Raskin and Taylor's ontological semantic technology [19, 20] and create an artificial procedural memory system that has human reasoning capabilities and mimics the fuzzy and uncertain nature of human cognitive processes. This new focus for Crowder [6] is to create processes necessary for the creation, storage, retrieval, and modification of artificial procedural memories (see Fig. 10.2).

10.7 Conclusions

Testing on Zeus has shown that self-evolution through creation, storage, and retrieval of artificial procedural memories can provide an effective and efficient mechanism for autonomous control of artificial life forms. There is a significant amount of research, development, and testing required in general, and specifically over the next year, including expansion of Zeus' neural framework to include a more comprehensive possibilistic, abductive neural network to allow hypothesis generation enhancing the speed and the quality of his cognitive skillset discussed throughout the book. Our continuous research in this area in developing higher fidelity cognitive functions and exploring new types of cognitive testing are important to establish the breadth of self-possibilities. We will continue to explore not only these low-level brain functions, but also much higher brain functions as well.

One question to be answered for comprehensive autonomous systems is, "How much initial information, or memories, should be provided?" Based upon our initial results, we speculate confidently that since learning is stochastic in nature and depends on current understandings, the more initial information or memories (cognitive ontology concepts) an entity begins with, the more it will influence the artificial life form's learning direction and its ability to "intuit" about its environment. Consequently, it becomes imperative to manage content quantity and quality efficiently within systems of widely varying resource constraints. Therefore, future efforts will include some research into synergistic algorithms within the Activity Based Intelligence (ABI) domain. ABI employs normalization methods of activity patterns (AP), which show possibilities for improving balance between knowledge storage, necessity, prime directives, and learned objectives. Additionally, research in improving condensation of decision quality content via anomaly detection in big data environments shows applicability to smart aggregation of knowledge in constrained environments.

Lastly, we have been exploring the question of how to test cognitive systems. Given that these systems will learn, think, reason, and self-evolve, we believe that standard system testing is inadequate to understand whether a system is "working properly." Hence, we believe it is imperative to understand self-evolving learning processes by testing them in a context of "cognitive" and "psychological" paradigms. Therefore, to obtain a comprehensive understanding of proper system functionality we believe it is necessary to test analogously to determining whether a human is "functioning normally." Finally, we believe this leads us to a field of study envisioned long ago, "Artificial Psychology [14]."

References

1. Crowder, J. (2010). Operative Information Software Agents (OISA) for intelligence processing. *Proceedings of the 12th annual International Conference on Artificial Intelligence*, Las Vegas, NV.

2. Kosko, G. (1986). Fuzzy cognitive maps. *International Journal of Man-Machine Studies, 24*, 65–75.
3. Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. In C. Nicolini (Ed.), *Modeling and analysis in biomedicine*. New York: World Scientific Publishing.
4. Hutchinson, J., Koch, C., Luo, J., & Mead, C. (1988). Computing motion using analog and binary resistive networks. *Computer, 21*(3), 52–63.
5. Kahana, M., Howard, M., & Polyn, S. (2008). Associative retrieval processes in episodic memory. In H. L. Roediger III (Ed.), *Cognitive psychology of memory* (Vol. 2 of Learning and memory: A comprehensive reference, 4 vols., J. Byrne, Editor). Oxford: Elsevier.
6. Crowder, J., Raskin, V., & Taylor, J. (2012). Autonomous creation and detection of procedural memory scripts. *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
7. Landauer, T., & Dumas, S. (1997). Solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review, 194*, 211–240.
8. Griffiths, T., & Steyvers, M. (2003). Prediction and semantic association. In K. J. Halmberg & M. Steyvers (Eds.), *Advances in neural information processing systems* (Vol. 15, pp. 11–18).
9. Carbone, J. (2010). *A framework for enhancing transdisciplinary research knowledge*. Texas: Tech University.
10. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
11. Crowder, J. (2011). Cognitive architectures for real-time integrated system health management. *Proceedings of the 8th Annual Workshop of Structural Health Monitoring*, Stanford University, Stanford.
12. Zadeh, L. (2004). A note of web intelligence, world knowledge and fuzzy logic. *Data and Knowledge Engineering, 50*, 291–304.
13. Crowder, J., & Carbone, J. (2011). Hybrid neural architectures for the ELYSE cognitive system. *Proceedings of the AIAA Infotech@Aerospace 2011 Conference*, Garden Grove, CA.
14. Crowder, J., & Carbone, J. (2012). Reasoning frameworks for autonomous systems. *Proceedings of the AIAA Infotech@Aerospace 2012 Conference*, Garden Grove, CA.
15. Ertas, A., & Tanik, M. (2000). Transdisciplinary engineering education and research model. *Transactions of the SDPS, 4*(4), 1–11.
16. Crowder, J. (2001). *Integrating an expert system into a neural network with genetic programming for process planning*. NSA Technical Paper TIT_01_01_013_2001_001.
17. Crowder, J., & Friess, S. (2010). Artificial neural diagnostics and prognostics: Self-soothing in cognitive systems. *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
18. Crowder, J., & Friess, S. (2012). Artificial psychology: The psychology of AI. *Proceedings of the 3rd Annual International Multi-Conference on Complexity, Informatics and Cybernetics*, Orlando, FL.
19. Raskin, V., Taylor, J. M., & Hemplemann, C. F. (2010). Ausocial engineering. *New Security Paradigms Workshop*, Concord, MA.
20. Taylor, J. M., & Raskin, V. (2010). Fuzzy ontology for natural language. *29th International Conference of the North American Fuzzy Information Processing Society*, Toronto, ON.

Chapter 11
Methodologies for Continuous, Life-Long Machine Learning for AI Systems

11.1 Introduction: Life-Long Machine Learning

A fully autonomous, artificially intelligent system has been the holy grail of AI for decades. However, current machine learning methodologies are too static and minimally adaptive enough to provide the necessary qualitative continuously self-adaptive learning required for possible decades of system performance. Therefore, industry is replete with promises of biologically inspired research and artificial human learning mechanisms for enabling AI neural pathways and memories to evolve and grow over time [1, 2]. However, to achieve this requires new methods and mechanisms that enable a paradigm shift providing, continuous, or life-long, machine learning algorithms and method evolution. Our objective in the book is to look at new architectures that requires controls and mechanisms like artificial brain functions for enabling complete cognitive system management. In short, to achieve continuous, life-long machine learning requires artificial neurogenesis[1], a new machine learning architecture and methods enabling a continuously self-adapting neural fiber structure within an AI system as illustrated in Fig. 11.1.

In this ANP, both explicit and implicit learning are required to adequately provide self-assessment throughout the AI system. Self-assessment is required for the system to understand how its self-adaptation is affecting all parts of the AI system [3]. Explicit learning, as defined here, requires cognitive and hierarchical associations, whereas implicit learning depends on non-cognitive, non-hierarchical associations, and, in general, occurs when a variable known to influence explicit learning has no effect in a comparable implicit learning condition [3]. Each type of learning has effects on the AI system's overall knowledge base and each type of learning may influence the other as more information is processed and stored within the various memory systems of the AI system. As illustrated in Fig. 11.1 not only is the neural

[1] Artificial neurogenesis (literally the birth of artificial neurons) is the processes in which new neurons are generated within the artificial memory system.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_11

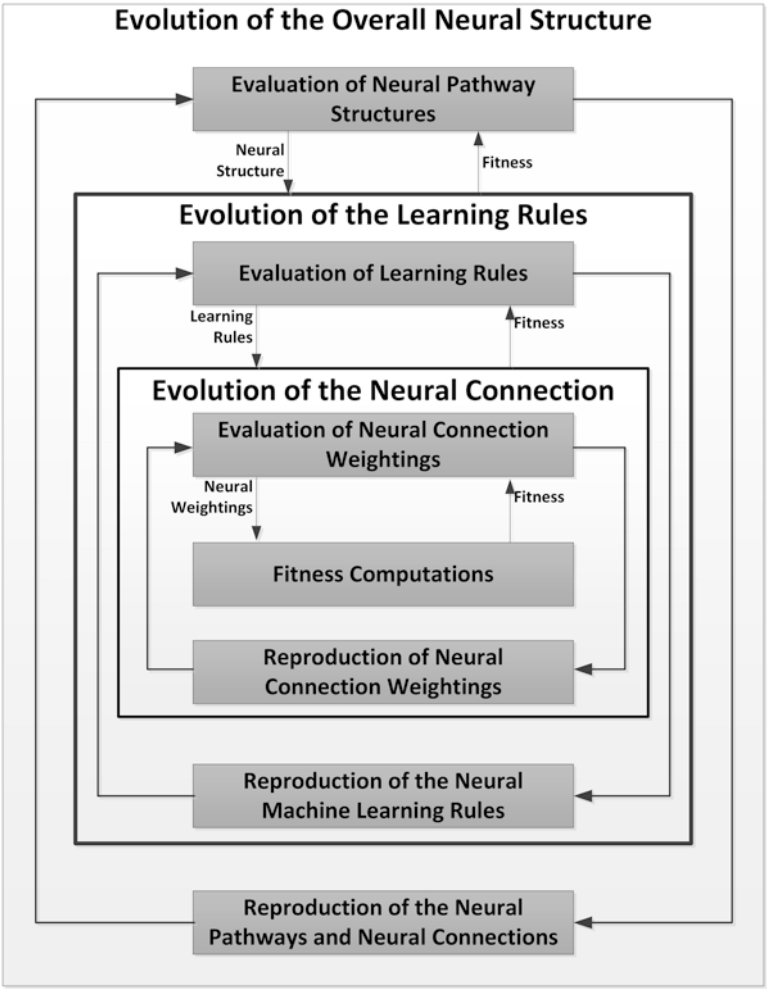


Fig. 11.1 The artificial neurogenesis process (ANP)

structure adaptive, but the learning rules themselves must be adaptable, driven by the continuous self-assessment functionality within the ANP. Figure 11.2 provides a high-level view of the coordination, interaction, and influence explicit learning, implicit learning, and the AI systems knowledge base have on each other [4].

A continuously adaptable, life-long machine learning architecture, from our studies, requires many types of learning to facilitate how the entire system must adapt as it learns, reasons, as the environments the system is in change, and as the system ages. To provide continual real-time decision support over time, we feel the following memory systems must be in place, and each be self-adaptive:

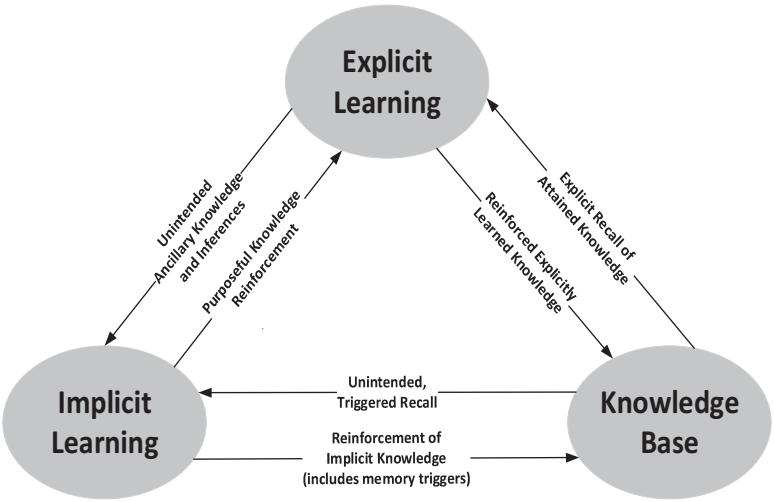


Fig. 11.2 The implicit, explicit, learning to knowledgebase influence triangle

1. **Perceptual Associative Memory**: the ability to interpret incoming stimuli by recognizing objects and by categorizing them.
2. **Procedural Memory**: memory for the performance of specific types of action. Procedural memory guides the processes the AI system performs and most frequently resides below the level of conscious awareness.
3. **Declarative Memory**: this is classical long-term memory and refers to memories that can be consciously recalled such as facts and knowledge (from the AI systems knowledge base).
4. **Transient Episodic Memory**: the memory of autobiographical events (times, places, associated emotions, and other contextual who, what, when, where, why knowledge) that can be explicitly stated or conjured. It is the collection of past system experiences that occurred at a specific time and place. Episodic memory stores unique events (or observations).
5. **Blackboard Memory**: a common knowledge base that is iteratively updated by the diverse set of components, software agents, etc. throughout the system. Blackboard memories typically start with a problem specification and end with a proposed solution.
6. **Sensory Memory**: this is the shortest-term type of memory. Sensory memory can retain impressions of the sensory information coming in through the various types of sensors the AI system has. These impressions are sent to the perceptual associative memory. These would be rudimentary at first, but then expand as the system learns.

Each type of memory is updated by life-long machine learning algorithms specifically created for that type of memory. In self-adaptive, continuous machine learning, there is no one learning algorithm or system that will suffice. Figure 11.3

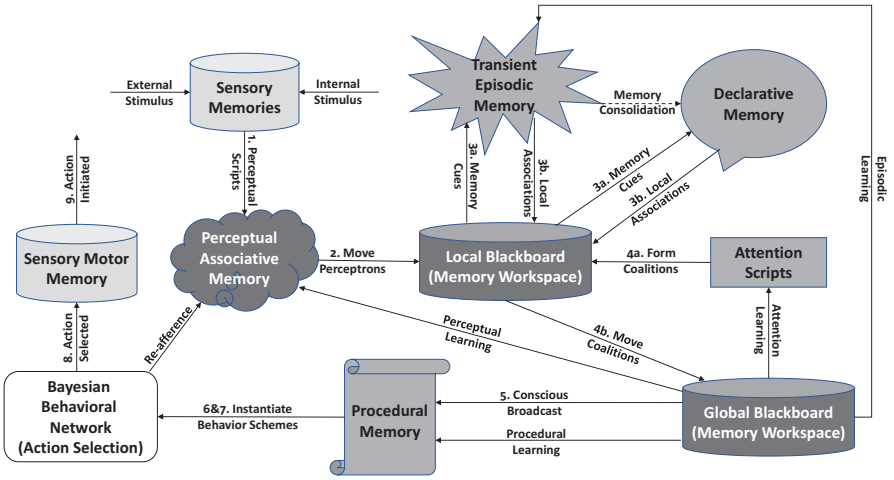


Fig. 11.3 Life-long machine learning process

illustrates the high-level architecture for a self-adaptive, continuous life-long learning structure for an AI system.

We employ abductive learning for finding the best explanation for a given set of observations or inferring cause from effect [5, 6]. This accommodates adjustment of learning types for self-adaptation to environments, data, and experiences the system has not previously encountered. We define a simplified version of abductive learning, Occam learning [7, 8], which relates to finding the simplest explanation(s) when inferring cause from effect(s).

The life-long learning architecture shown in Fig. 11.3 is primarily used to describe support for procedural and perception learning. Our research shows that there are at least four types of learning required so a fully autonomous AI system can potentially learn, and reason. They are:

1. **Episodic Learning**: the process of storing/retrieving experiences in the episodic memory and using it to improve behavior (responses to stimulus).
2. **Attention Learning**: also called concentration, attention learning stores triggers that allow the AI system to focus its efforts on objects or events of interest.
3. **Perceptual Learning**: the process of learning skills of perception. This allows continuous improvement in sensory processing (how to distinguish objects from sensory information—an example would be ATR), to complex categorizations of spatial and temporal patterns. Perceptual learning forms the foundation for an AI

system to create complex cognitive processes (e.g., language). Perceptual learning drives adaptations (changes) in the AI systems neural circuitry or patterns.

4. **Procedural Learning**: learning by acquiring skill at performing a task. Procedural learning allows the AI system to perform a task "automatically" without consuming resources to determine how to accomplish the task [9].

11.2 Artificial Intelligence Machine Learning with Occam Abduction

Occam abduction is used to find the simplest set of consistent assumptions and hypotheses, which, together with available background knowledge, entails adequate description/explanation for a given set of observations [10]. In formal logic notation, given $\boldsymbol{B_D}$, representing current background knowledge of domain $\boldsymbol{D}$, and a set of observations $\boldsymbol{O_D}$, on the problem domain $\boldsymbol{D}$, we look for a set of Occam hypotheses, $\boldsymbol{H_D}$, such that:

– $\boldsymbol{H_D}$ is consistent[2] w.r.t. $\boldsymbol{B_D}$, and
– It holds that $\boldsymbol{B_D}, \models \boldsymbol{H_D}, \rightarrow \boldsymbol{O_D}$

Abduction consists of computing explanations (hypotheses) from observations. It is a form of non-monotonic reasoning and provides explanations that are consistent with a current state of knowledge and may become less consistent or inconsistent, when new information is gathered. The existence of multiple hypotheses (or explanations) is a general characteristic of abductive reasoning, and the selection of the preferred, or most simple, but possible, explanation is an important precept in artificial Occam abduction.

Abduction was originally embraced in artificial intelligence work as a non-monotonic reasoning paradigm to overcome inherent limitations in deductive reasoning. It is useful in artificial intelligence applications for natural language understanding, default reasoning, knowledge assimilation, belief revision, and very useful in multi-agent systems [11]. The abduction form of inference, using hypotheses to explain observed phenomena, is a useful and flexible methodology of reasoning on incomplete or uncertain knowledge. Occam abduction, defined herein, provides not only an answer, or cause, to the observations, it provides class properties of possible hypotheses within which observations are determined valid, and denotes the simplest set of hypotheses under which this is true.

[2] If $\boldsymbol{H}_{\mathrm{D}}$ contains free variables, $\exists(\boldsymbol{H}_{\mathrm{D}})$ should be consistent w.r.t. $\boldsymbol{B}_{\mathrm{D}}$.

11.2.1 Elementary Occam Abduction

There are several distinct types of interactions that are possible between two elementary Occam abductive hypotheses $h_1, h_2 \in H_e$ [12]:

- ***Associativity***: The inclusion of $h_1 \in H_e$ suggests the inclusion of h_2. Such an interaction may arise if there is knowledge of, for instance, mutual information (in a Renyi sense) between h_1 and h_2.
- ***Additivity***: h_1 and h_2 collaborate additively where their abductive and explanatory capabilities overlap. This may happen if h_1 and h_2 each partially explain dome datum $d \in D_0$ but collectively can explain more, if not all of D_0.
- ***Incompatibility***: h_1 and h_2 are mutually incompatible, in that if one of them is included in H_e then the other one should not be included.
- ***Cancellation***: h_1 and h_2 cancel the abductive explanatory capabilities of each other in relation to some $d \in D_0$. For example, h_1 implies an increase in a value, while h_2 implies a decrease in a value. In this case, one is used to support the hypothesis and the other is used to rebut the hypothesis.

The Occam abductive process is:

- Nonlinear in the presence of incompatibility relations
- Non-monotonic in the presence of cancellation relations
- The general case (nonlinear and non-monotonic) Occam abduction hypothesis investigation is NP-complete.

Consider a special version of the general problem of synthesizing an artificial Occam abductive composite hypothesis that is linear, and, therefore, monotonic. The synthesis is linear if:

$$\forall h_i, h_j \in H_e, \quad q\left(h_i\right) \cup q\left(h_j\right) = q\left(\left\{h_i, h_j\right\}\right) \tag{11.1}$$

The synthesis is monotonic if:

$$\forall h_i, h_j \in H_e, \quad q\left(h_i\right) \cup q\left(h_j\right) \subseteq q\left(\left\{h_i, h_j\right\}\right) \tag{11.2}$$

In this special version, we assume that the Occam hypotheses are non-interacting, i.e., each offers a mutually compatible explanation where their coverage provides mutual information (in a Renyi sense). We also assume that the Occam, abductive belief values found by the classification subtasks of abduction for all $h \in H_e$ are equal to 1 (i.e., true).

Under these conditions, the synthesis subtask of artificial Occam abduction can be represented by a bipartite graph, consisting of nodes in the set $D_0 \cup H_e$. This says there are not edges between the nodes in D_0, nor are there edges between the nodes in H_e. The edges between the nodes in D_0 and those nodes in H_e can be represented by a matrix $\boldsymbol{Q}$ where the rows correspond to $d \in D_0$ and the columns correspond to $h_t \in H_e$.

The entries in $\boldsymbol{Q}$ are denoted as $\boldsymbol{Q_{ij}}$ and indicate whether the given analyzed data are explained by a specific abductive Occam hypothesis. The entries are defined as:

$$Q_{i,j} = \begin{cases} 0 & \text{if datum } d_i \text{ is not explained by hypothesis } h_j \\ 1 & \text{if datum } d_i \text{ is explained by hypothesis } h_j \end{cases} \tag{11.3}$$

Given the matrix Q for the bipartite graph, the abductive, Occam synthesis subtask can be modeled as a set-covering problem, i.e., finding the minimum number of columns that cover all the rows. This ensures that the composite abductive, Occam hypothesis will explain all of D_0 and therefore be parsimonious.[3]

Now we look at a special linear and monotonic version of the general abductive, Occam hypothesis synthesis subtask and look at a Possibilistic Abductive Neural Networks (PANNs) for solving it [3]. The first is based on an adapted Hopfield model of computation:

$$\forall i = 1,2,\ldots,n, \quad \sum_{j=1}^{m} Q_{ij} V_j \geq 1 \tag{11.4}$$

For the Occam, abductive synthesis subtask, we associate variable V_j with each Occam hypothesis $h_t \in H_e$, to indicate if the Occam hypothesis is included in the composite Occam, abductive hypothesis $\boldsymbol{C}$. We then minimize the cardinality of $\boldsymbol{C}$ by:

$$\sum_{j=1}^{m} V_j \tag{11.5}$$

subject to the constraint that all data $d \in D_0$ are completely explained.

For the Occam, abductive network, the term in the energy function that represents the problem constraints must evaluate to zero when the constraint is satisfied and must evaluate to a large positive value when the constraint is not satisfied, forcing the evolving solution lattice to evolve accordingly [1]. For this energy term, we use a term expressed as a sum of expressions, one for each datum element, $\boldsymbol{d_i}$, such that the expression evaluates to zero, when hypothesis $\boldsymbol{h_j}$ that can explain the datum $\boldsymbol{d_i}$ is in the composite hypothesis, i.e., $V_j = 1$. Given that $\boldsymbol{Q}$ is an incidence matrix (with elements either 0 or 1), the expression:

$$\sum_{i=1}^{n} \prod_{j=1}^{m} \left\{ \left(1 - Q_{ij}\right) + \left(1 - V_j\right) \right\} \tag{11.6}$$

satisfies the following conditions:

- Each sum of the product terms can never evaluate to a negative number.

[3] Note that the general set-covering problem is NP-complete.

- The sum of the product terms, thus, can never evaluate to a negative number.
- Each product term evaluates to zero when a hypothesis that can explain the datum is in the composite; otherwise, it evaluates to a large value.
- The sum of the product term, thus, evaluates to zero when a composite set of hypotheses can explain all the data.

We derive our Occam abductive energy function as follows:

$$E = \alpha * \sum_{j=*1}^{m} V_j - \tag{11.7}$$

$$\beta * \sum_{i=1}^{n} \prod_{j=1}^{m} \left\{ \left(1 - Q_{ij}\right) + \left(1 - V_j\right) \right\} \tag{11.8}$$

where α and β are positive constants, and $\beta > \alpha$. The first term represents the cardinality of the Occam hypothesis and the second term represents the penalty for a lack of complete coverage; 0 indicates complete coverage.

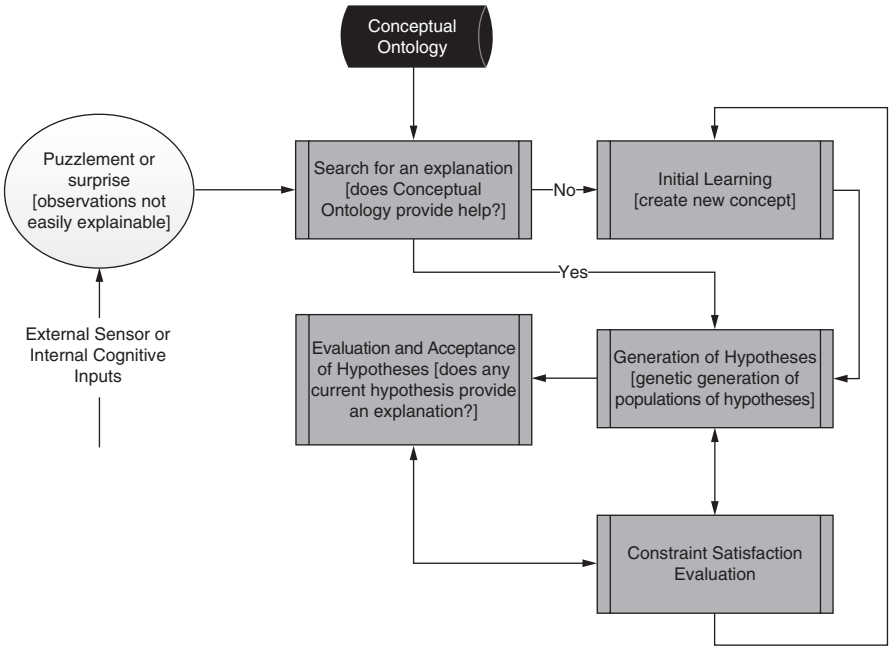


Fig. 11.4 Elementary continuous abductive learning

11.3 Elementary Continuous Abduction

Continuous machine learning requires continuous abduction, which drives us to constantly look for ways to explain either the external environment, or things within the AI system (self-reflection). This requires an architecture and process for continuous abduction. Figure 11.4 illustrates this process.

Here, the assumptions are:

- The Occam causes are mutually exclusive and constitute exhaustive coverage of the effects.
- Each of the Occam causes is conditionally independent.
- Each of the Occam causes is not mutually incompatible.
- None of the Occam causes cancel the abductive explanatory capability of any other Occam cause.

From Fig. 11.4, we see that when observations are present for which there are no explanations, the Occam abduction system creates a set of hypotheses (possible explanations). Each of these hypotheses is tested to create a plausible set of explanations. The system expands to generalized hypotheses if needed. Figure 11.5 illustrates a high-level architecture for a generalized life-long machine learning abduction model. This architecture generalizes the observations into categories. If no concepts exist to explain the observations, new concepts must be created to accommodate the observations.

Hypotheses are generated by looking at similarities and differences between the observations and categories. Conflict between hypotheses must be adjudicated. Eventually, a set of non-interfering, non-overlapping hypotheses that explain the observations is created, learned from, and decisions made. Attributes of these

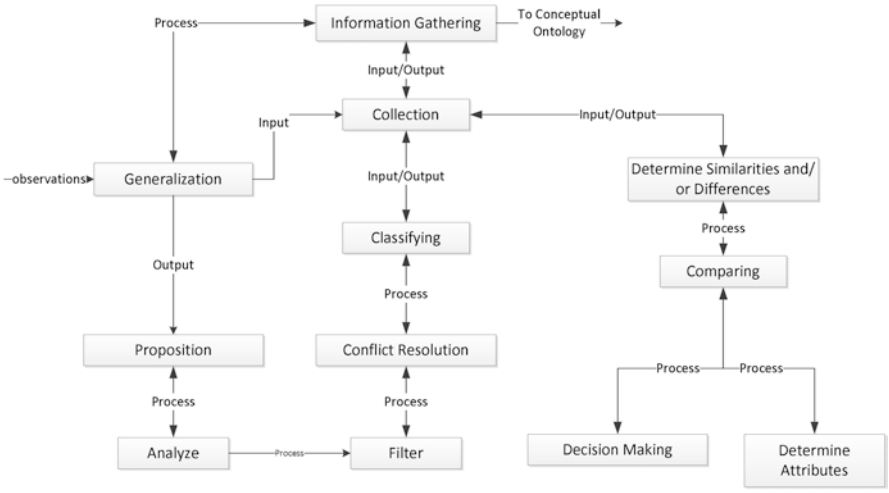


Fig. 11.5 Generalized life-long abductive machine learning

hypotheses are categorized and learned, including any memory triggers that are needed.

11.4 Conclusions and Discussion

This is very preliminary work and much more research is required. Here we have presented a high-level view and discussion of the possibility of an AI system with continuously adapting, life-long machine learning. The architectures, structures, methods, and algorithms require a complete change from current thinking and development. We believe this is the future of autonomous and semi-autonomous AI systems. Research must be continued on the Occam learning algorithms to determine what constitutes an acceptable Occam abduction energy level and to understand how to apply the weighting factors in the Occam energy equation (i.e., is it domain specific?).

References

1. Crowder, J. A. (2010). The continuously recombinant genetic, neural fiber network. *Proceedings of the AIAA Infotech@Aerospace-2010*, Atlanta, GA.
2. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. *Proceedings of the 12th International Conference on Artificial Intelligence*, Las Vegas, NV.
3. Stadler, M. (1997). Distinguishing implicit and explicit learning. *Psychonomic Bulletin & Review, 4*(1), 5–62.
4. Crowder, J. A. (2010). Flexible object architectures for hybrid neural processing systems. *Proceedings of the 11th International Conference on Artificial Intelligence*, Las Vegas, NV.
5. Crowder, J. (2016). AI inferences utilizing occam abduction. *Proceedings of the 2016 North American Fuzzy Information Processing Symposium*, University of Texas, El Paso.
6. Crowder, J., & Carbone, J. (2017). Abductive artificial intelligence learning models. *Proceedings of the 2017 International Conference on Artificial Intelligence*, Las Vegas, NV.
7. Carbone, J. N., & Crowder, J. (2011). Transdisciplinary synthesis and cognition frameworks. *Proceedings of the Society for Design and Process Science Conference 2011*, Jeju Island, South Korea.
8. Crowder, J. (2005). Cognitive systems for data fusion. *Proceedings of the 2005 PSTN Processing Technology Conference*, Ft. Wayne, IN.
9. Jahanshahi, W. (2007). The striatum and probabilistic implicit sequence learning. *Brain Research, 1137*, 117–130.
10. Crowder, J. A. (2002). *Machine learning: Intuition (concept learning) in hybrid neural systems*. NSA Technical Paper-CON-SP-0014-2002-06, Fort Meade, MD.
11. Franklin, S. (2005). Cognitive robots: Perceptual associative memory and learning. *Proceedings of the 2005 IEEE International Workshop on Robot and Human Interaction*.
12. Crowder, J. (2004). Multi-sensor fusion utilizing dynamic entropy and fuzzy systems. *Proceedings of the Processing Systems Technology Conference*, Tucson, AZ.

Chapter 12
Implicit Learning in Artificial Intelligence

12.1 Introduction

Current research asserts that implicit learning is a fundamental and continuous process in overall cognition of an entity [1]. This notion of "learning without awareness" has far reaching implications for autonomous artificial intelligent entities as we push toward systems with continuous, life-long machine learning, that can continually adapt as they experience their respective environments. We seek to understand the associative learning mechanisms within overall continual machine learning and look for statistical dependencies between the environments they experience and the implicit learning and the knowledge representations they create and store as implicit memories.

One of the main differences between explicit and implicit memories is that implicit memories stores unconscious memories of skills and "how to do things." Explicit memories store facts and events that can be recalled by conscious thought. Memory is not a single system in the mind but several systems [2]. These systems have different operating principles. One example of this was in a case of amnesia [3, 4], where explicit memory was interrupted but implicit memory was not. There are a few principles that guide our understanding of memory. First memory is its own ability and separates from other cognitive abilities. Explicit memory is also known as declarative memory. These memories are facts, events, and unconscious materials. Implicit memory is skill learning and forming habits. In implicit memory, experience modifies behavior without any conscious content or experience that the memory is being used. Implicit memory is measurable through performance. It is not recollection. The two systems operate parallel to each other. Implicit memory could be thought of impacting or creating personality traits. Adverse events could impact how one behaves. For example, if a person experiences a near-death car accident and is airlifted out, without remembering consciously, the person could become anxious around the smell of jet fuel from helicopters. We then derive the definitions we will use during this discussion [5]:

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_12

Definition of Implicit Memory: Procedural memories that are used without awareness so that contents of memories can't be reported and may be used automatically without conscious thought.

Definition Explicit Memory: Declarative memories based on the personal experiences, stored knowledge, and memory of facts that can be directly reported or recalled.

Table 12.1 illustrates differences between explicit and implicit memories, in terms of areas of the brain associated with each type of memory.

What follows is our view of how implicit learning may affect the overall function, continued learning, and inferencing among artificial intelligent entities (robots).

12.2 Implicit Learning in Artificial Intelligent Systems

As explained above, implicit learning involves entities learning complex information in an incidental matter [5]. Implicit learning represents non-episodic learning, either from visual, consequential, or functional stimulus structures. These can drive autonomous reinforcement and learning and reflects that behavior (learned responses) can be modified by consequences of interaction with an entity's environment without the entity's awareness. It is not necessary to recognize the relationship between an action and a reinforcing consequence for implicit learning reinforcement to happen. The result of implicit learning is the storage of implicit knowledge in non-episodic implicit memory. The result is implicit memories (knowledge) that manifest itself as abstract representations rather than explicit or aggregate representations. It can drive the entity toward certain biases in its decisions and inferences and can result in different learning stimulus structures; modifying the way the entity learns, or the way the entity interprets certain types of information. Figure 12.1 illustrates this process. Figure 12.1 is an adaptation of the learning model illustrated in Fig. 5.7 to include the effects of implicit learning on an overall artificial intelligent learning model [5, 6]. Examples of implicit learning among people are the abilities to ride a bike, to fill your car with gas, and swimming. Each of these are stored as an implicit procedural memory that can be recalled without conscious

Table 12.1 Implicit vs. explicit memory

Factor	Explicit memory	Implicit memory
Memory process	Conscious and purposeful	Unconscious and automatic
Memory structure	Hippocampus and temporal lobe	Neocortex, cerebellum, and others
Information	Facts, verbal, semantic, operational, and procedural descriptions	Emotional, conditioning, sensory, automatic skill, and procedural skills

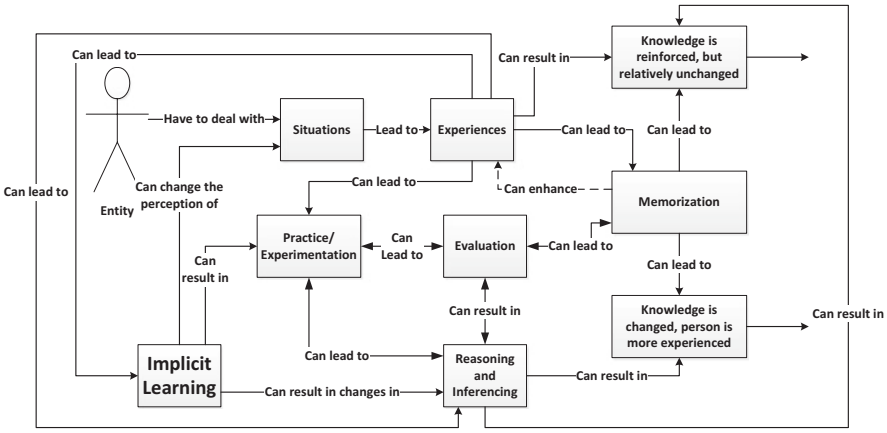


Fig. 12.1 Artificial Intelligence Learning Model with Implicit Learning

thought. Examples of implicit learning in artificial intelligent entities might be off-line observed experience from videos for teams of robots to implicitly learn how to coordinate activities. Computational models point to the ability to implicitly learn performance prediction models from off-line implicit learning that allow robots to implicitly coordinate activities in real-time situations different from those presented in the videos but represent implicit coordination between robot entities [7].

Research into both human and robotic learning supports a clear distinction between implicit and explicit learning. As illustrated in Table 12.1, different areas of the brain are involved in implicit vs. explicit learning, and MRIs indicate the differences involved in working memory and attention during implicit vs. explicit learning. As discussed above, research on amnesia patients indicate that in many cases implicit learning remains intact while explicit learning is severely impacted [6].

One prominent impact of implicit learning that artificial intelligent robots or entities that must interact on an ongoing basis with humans, is the implicit learning ability to understand without verbal explanation, which involves the decoding of social interaction signals. Often people and animals can judge the personalities of others without engaging in prolonged social activity because of their implicit understanding of regular behaviors of people [8]. This is a direct effect of implicit learning and implicit memories on inferences. We have an implicit knowledge of how to "infer" the actions or intentions of entities we interact with in our environment. Figure 12.2 below illustrates the effects of implicit learning on a decomposition of an inference learning model.

Implicit learning, which is based on experiences an entity encounters while interacting with its environment, can create emotional (assuming the entity has emotions) responses, called emotional triggers, that the entity doesn't know exist until the emotion is triggered later. Implicit learning and the associated implicit memories can drive actions that are unexpected. This is especially true in martial arts

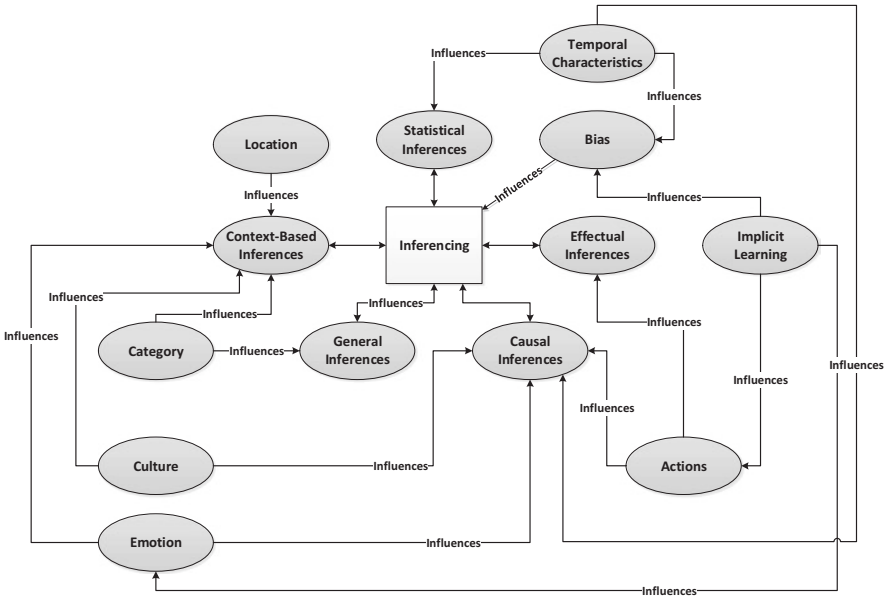


Fig. 12.2 Artificial Intelligence Cognitive Inference Breakdown with Implicit Learning

training in creating what is known as "muscle memory," which is stored as an implicit procedural memory within the memory system. When a movement is continually practiced, or experienced, long-term procedural muscle memory is created for that activity, allowing it to be performed automatically without any conscious thought or effort. This is helpful within a person and within a system, as it decreases the system resources needed to facilitate the action and creates efficiencies within the effector (e.g., motor) and memory systems. Such efficiencies might be beneficial for long-term acting systems with limited resources [9]. Implicit learning, and the creation of implicit memories, would allow the artificial intelligent entity to focus on complex mental processes, like problem solving, while allowing more routine processes to remain active without requiring conscious attention [8].

As was discussed earlier, different areas of the brain and different mental processes are exercised during explicit vs. implicit learning and memory. This leads us to believe that there are two independent learning systems for explicit vs. implicit learning. These learning systems have two distinctions:

1. Learning that takes place with and without awareness
2. Learning that involves encoding of experiential instances vs. the induction of abstract rules or hypotheses [9, 10]

As stated, implicit learning involves unconscious learning. One of the issues then with implicit learning is that the entity may not even be aware of why a decision was made or an action was initiated. This will be problematic for artificial intelligence

systems which may act in a manner contrary to its given task or mission without being able to articulate what happened [11].

For artificial intelligent systems, we believe that the notion of implicit learning within the artificial cognitive infrastructure of the artificial intelligent entity should be thought of as a complex instantiation of cognitive priming [9] which will invariably take place within a continuously learning artificial neural system. The system will, we believe, create distributed knowledge within the procedural long-term memories and can be causally active in the absence of the entity's conscious processes. This means, the artificial intelligent entity can develop implicit memories that is currently influencing process and contains no metaknowledge of the memories or their effects on the overall cognitive system [9].

The ability of operators, customers, end-users, or developers to understand and ascertain whether implicit learning is happening, that implicit memories have been created, and how these implicit memories may affect the overall system derives from the ability to capture three different dependent "measures of response modalities" [12]:

1. Conceptual Fluency: The ability to understand concepts and to apply them accurately and efficiently to different problems and contexts.
2. Efficiency: The quantitative measure of knowledge increase in relation to time and effort. How easily implicit learning happens and how efficiently does the entity translate experiences into implicit memories.
3. Prediction and Control: The ability of the entity to learn to translate implicit learning and memories into unconscious interventions to control the outcome of an event or situation, or the ability of the entity to unconsciously predict the outcome from observing changes over a short period of time [5].

The ability of the system to handle inevitable implicit learning and implicit memories will depend on the attention that can be utilized (the available resources) and will depend on the attentional and working memory systems available within the artificial intelligent systems. If there are no mechanisms for storing implicit memories (e.g., procedural memory), the resultant implicit knowledge may be present across the system in various abstract forms, affecting many parts of the system, rather than aggregate representations stored in procedural memories. This can drive biases and dissociations in learning across the system, causing it to learn differently across various stimulus (experiences from its environment) [5].

12.3 Measuring Implicit Learning Within an Artificial Intelligent System

When researching and deciding the measures that are appropriate to determine if an artificial intelligent system has experience implicit learning (which, by the way, it will) and developed implicit memories, we must first develop criteria to understand

what is to be regarded as implicit knowledge within an artificial intelligent system. Here are two that should be considered [13]:

1. Implicit knowledge in the sense it is difficult for the system to articulate the information.
2. Implicit knowledge in the sense that its decisions and inferences are created according to a subjective threshold, not an objective threshold.

Utilizing the methodology laid out in Chap. 1 and illustrated in Fig. 1.4, evaluation evidence for implicit learning is most likely feasible in terms of assessing objective vs. subjective thresholds. For understanding criteria 1, difficult for the system to articulate, the use of procedural memories, as discussed in Chap. 10 will allow methods and mechanisms to be designed into the overall processing infrastructure of the system to detect changes within the procedural memories. This makes it imperative for the system to have procedural memories within the memory infrastructure of the overall artificial intelligent system framework.

In order to facilitate evaluation of subjective vs. objective thresholds for decision-making and inferencing, subjective thresholds and implicit knowledge in general, is inflexible in its transfer to different knowledge domains. Also, implicit learning happens often when the systems attention is focused on specific experiences and specific events and not rules and underlying constraints. A lack of rules and guiding principles when an artificial intelligent system is experiencing its environment will lead to implicit learning and the subsequent implicit knowledge is generally robust and not easily changed or reinterpreted. In short, it is hard to get over first impressions, even by an artificial intelligent entity [13, 14].

12.3.1 Measuring Implicit Learning in Artificial Intelligent Systems

System test, in general, can be viewed as a series of experiments performed on and against the system to determine the system's response to a given set of stimuli. In most cases, this is nothing more than, if I give it a given set of inputs, do I get the right outputs. However, for a system that learns, reasons, and self-adapts, testing takes on a more experimental nature. Does the system react correctly? Does the system learn correctly? Can the system adapt to changing situations or input data? Can it identify objects if I change them sufficiently?

Such experiments may take the form of [14]:

1. Determining if the Artificial Intelligent entity can carry out "efficient" actions given a situation it is trained for.
2. Can the Artificial Intelligent entity articulate answers about the situation and why it made the decisions it chose?

It is believed [14], but much experimentation is needed to verify, that it is possible for the artificial intelligent entity to show a performance improvement without a change in the ability to articulate this (i.e., verbal knowledge). However, it is again believed, but experimentation is required, that it is difficult to show changes (improvement) in the ability to verbalize its decisions without showing an improvement in performance. The first equates to implicit learning that explicit memories haven't captured. The second is driven by an explicit change to learning and to explicit memories, which the artificial intelligent entity can easily retrieve and articulate. One of the issues with creating experiments (tests) for artificial intelligent entities is how to define the tasks that can drive an entity to implicit learning in order to detect discrepancies in its memories. The assumption with a trained network is that it knows how to do what it is trained to accomplish. Care must be taken to understand the variables that may affect processes within the artificial intelligent entity. Again, it is fital that all the test cases and strategies be defined and build into the design of the artificial intelligent system up front. Trying to design tests after the artificial intelligent code is designed, coded, and implemented will be virtually impossible. Ask the makers of the autonomous robot at the 2018 Consumer Electronics Show discovered when it wandered into the street and was run over by a self-driving car. It is doubtful that such a scenario was ever envisioned by either the robot engineers or the self-driven car engineers.

12.3.2 Measuring Implicit Learning in Human–Machine Interfaces

In order to have effective artificial intelligent entities out in the world, assisting humans in a variety of ways, both military and commercial, it requires a sophisticated human–machine interface to facilitate communication and understanding between the human and the artificial intelligent entity. Creation of such an interface requires combining the work of neuroscience, psychology, and ethology[1] with the theory of artificial intelligent computation in mathematics and computer science advanced in linguistics [15]. One of the issues with human–machine communication and collaboration is that humans operate implicitly, based on verbal and visual cues that may be unavailable to the artificial intelligent entity. In order to facilitate efficient and functional communication, the artificial intelligent entity must have the ability to capture these implicitly learned cues. If we assume that there does exist both explicit and implicit learning systems, then it is imperative to provide both to an artificial intelligent entity that we are designing to interface with human operators or customers. Therefore, we must distinguish between:

1. Learning that takes place with and without concurrent awareness

[1] Ethology involves the study of non-human behavior, focused on the behavior under natural conditions, assuming such behavior comes evolutionary adaptation.

2. Learning that involves the encoding of situational instances
3. Learning that involves abstract rules of hypotheses

Implicit learning involves unconscious rule learning and understanding the difference between instance learning and rule learning provides a meaningful way of testing artificial intelligence learning [16]. Classical neural network artificial intelligent system learning involves providing the system with sets of training data used to create weighting factors that allow the artificial intelligent system to then properly classify object types in the future. This corresponds to instance learning and creates explicit learning within the artificial intelligent entity. However, for more complex artificial intelligent systems that can constantly adapt and evolve as they interact with their environments (e.g., autonomous robots), they may learn from instances (i.e., explicit experiences) or they may learn from implicit rules they witness and experience throughout their interaction with their environment. If the artificial intelligent system can interact with their environment unsupervised, it is very possible for non-instance-based, implicit rule learning to occur. Designing in test mechanisms and built-in-test procedures that are triggered on change in memory systems, especially procedural memories, may provide the ability to discover implicit learning within an artificial intelligent entity [17].

12.4 Conclusions

The notion of implicit learning is still a major topic of discussion among cognitive scientists, psychologists, and design engineers. Much of the current controversy and discussion centers on effective ways to measure implicit learning [17]. We believe the design on procedural memory systems within artificial intelligent entities provides a beginning step in allowing the retrieval of implicit learning and the resulting implicit memories [10]. It is important to allow the measurement of and to distinguish between explicit and implicit learning within an artificial intelligent entity [12].

References

1. Frensch, P., & Runger, D. (2003). Implicit learning. *Current Directions in Psychological Science, 12*, 13–18. https://doi.org/10.1111/1467-8721.01213.
2. Olivera, F. (2000). Memory systems in organizations: An empirical investigation of mechanisms for knowledge collection, storage and access. *Journal of Management Studies, 37*(6), 811–832.
3. Scoville, W. B., & Milner, B. (1957). Loss of recent memory after bilateral hippocampal lesions. *Journal of Neurology, Neurosurgery, and Psychiatry, 20*(1), 11–21.
4. Squire, L. R. (2009). Memory and brain systems: 1969–2009. *Journal of Neuroscience, 29*(41), 12711–12716.

5. Seger, C. (1994). Implicit learning. *Psychological Bulletin, 115*(2), 163–196. https://doi.org/10.1037/0033-2909.115.2.163. PMID 8165269.
6. Cleeremans, A. (1996). Principles of implicit learning. In D. Berry (Ed.), *How implicit is implicit learning?* (pp. 196–234). Oxford: Oxford University Press.
7. Stulp, F., Isik, M., & Beetz, M. (2006). Implicit coordination in robotic teams using learning prediction models. *Proceedings of IEEE International Conference on Robotics and Automation*. https://doi.org/10.1109/ROBOT.2006.1641893.
8. Shanks, D., & St. John, M. (1994). Characteristics of dissociable human learning systems. *Behavioral and Brain Sciences, 17*(3), 367–395. https://doi.org/10.1017/s0140525x00035032.
9. Fitch, W., Friederici, A., & Hagoort, P. (2012). Pattern perception and computational complexity: Introduction to the special issue. *Philosophical Transactions of the Royal Society B: Biological Sciences, 367*(1598), 1925–1932. https://doi.org/10.1098/rstb.2012.0099. PMC 3367691. PMID 22688630.
10. Crowder, J., Friess, S., & Carbone, J. (2013). *Artificial cognition architectures*. New York: Springer. isbn:978-1-4614-8071-6.
11. Cleeremans, A., Destrebecqz, A., & Boyer, M. (1998). Implicit learning: News from the front. *Trends in Cognitive Sciences, 2*(10), 406–416. https://doi.org/10.1016/S1364-6613(98)01232-7. PMID 21227256.
12. Putnam, A. (2011). *The effects of response modality on retrieval*. All theses and dissertations (ETDs) (p. 744). Retrieved from http://openscholarship.wustl.edu/etd/744.
13. Dienes, Z., & Berry, D. (1997). Implicit learning: Below the subjective threshold. *Psychonomic Bulletin & Review, 4*, 3–23. https://doi.org/10.3758/BF03210769.
14. Broadbent, D., FitzGerald, P., & Broadbent, H. (1986). Implicit and explicit knowledge in the control of complex systems. *British Journal of Psychology, 77*, 33–50.
15. Michas, I., & Berry, D. (1994). Implicit and explicit processes in a second-language learning task. *European Journal of Cognitive Psychology, 6*(4), 357–381. https://doi.org/10.1080/09541449408406520.
16. Stadler, M. A. (1997). Distinguishing implicit and explicit learning. *Psychonomic Bulletin & Review., 4*(1), 56–62. https://doi.org/10.3758/BF03210774.
17. DeKeyser, R. (2008). Chapter 11: Implicit and explicit learning. In C. J. Doughty & M. H. Long (Eds.), *The handbook of second language acquisition*. Oxford: Blackwell Publishing.

Chapter 13
Data Analytics: The Big Data Analytics Process (BDAP) Architecture

13.1 Introduction: Enhancing Big Data Analytics

Big data analytic systems look to enhance current legacy and future processing of an ever-increasing set of complex data [1]. Throughout the book, we have looked at various aspects of artificial intelligence and how to adequately create and test such systems. This drives home the need for big data analytics. Some of the technical major challenges with big data analytics are:

- How to handle scalability and complexity of the ever-increasing data streams.
- Knowledge management and knowledge economy of bid data environments [2].
- Cyber security in big data environments.
- Operational capabilities take too long to field.
- Swimming in sensors and downing in data[1].

In addition to the major technical challenges faced by big data analytic systems, some of the common real-world objectives that are required for overall operations and maintenance of big data processing systems are [3]:

- Reducing cost while migrating to ever-emerging scalable technologies
- Managing security across the vast, complex "data-to-decision" knowledge cycle
- Eliminating needs and gaps when deriving actionable and predictive mission-focused content [4]

The bottom line for real-world big data analytical systems/architectures is how to increase operational effectiveness while reducing the overall cost of ownership. This is required because the development of valuable readily consumable knowledge density and context quality continues to improve slowly and incrementally [5]. New concepts, mechanisms, and implements are required to facilitate the development and competency of complex systems to be capable of autonomous operation,

[1] Lt. General Deptula, December 2009.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_13

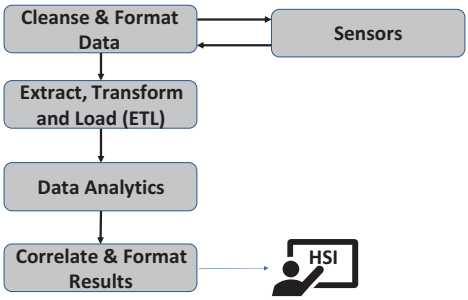


Fig. 13.1 Big data analytics high-level process

self-healing, and thus critical management of their knowledge economy and higher fidelity self-awareness of their real-time internal and external operational environments [6]. What is presented here is a high-level processing and computing architecture which facilitates big data analytics in a modern, scalable, extensible processing environment that provides:

- A significant decrease in overall operating costs to drive an overall reduction in the total cost of ownership
- An "on-demand" analytical process focused on the need to accelerate delivery of actionable intelligence from "big data"
- An adaptive, elastic, knowledge and context-based processing and computer architecture
- A high-speed distributed data infrastructure which optimizes time, storage, and "data-to-decision[2]"

13.2 The Big Data Analytical Process (BDAP)

Big data analytics optimization is applied to input sources. Upon collection, each input source is decomposed and reduced to its core characteristics. To significantly enhance fidelity, Extract, Transform, Load (ETL) functionality is included for improving scaling of all input source types: Sockets, Filesystem, etc. Common ingest and processing components are included as with many big data architectures for processing passive/batch and/or active/streaming data. Sensor/input data, among others, is transformed as it flows through the scalable ETL process, to ensure it is in the format required for analytical algorithms. This data is logically segregated into Binary and Ascii data processing bins where respective algorithms then can initially validate/verify data sourcing. Sourcing is critical in determining veracity, quality, and stewardship disambiguation. Once data analytics algorithms have processed the ingest and tagging of the input data, the resulting information must be correlated, combined, and/or fused with any previous results for

[2] This is often referred to as the "OODA" loop: observe, orient, decide, and act.

knowledge development and/or potentially formatted for dissemination. The overall tasks shown in Fig. 13.1 are primarily for streaming sensor data. Existing knowledge memory data is compared, contrasted, associated, and normalized against incoming streaming data:

- Collect data
- Verify data
- Analyze for missing data
- Tag data
- Analyze for functional data
- Correlate/classify with existing data
- Verify analytical results
- Send/display results

Within the data analytics process, data must be characterized and classified in order to prioritize the overall processing, cataloguing, and analyzing to maximize the efficiency of the system.

13.3 Data Characterization and Classification Process

Here, metadata are computed to determine characterization, classification, and overall priority of the data being analyzed. We will utilize a mutual information calculation to determine the overall correlation of data sets to data classification and priority measures. Figure 13.2 illustrates this process. The task list for this process is [7]:

- Extract metadata
- Compute mutual information
- Compute fuzzy membership values
- Apply defuzzification
- Publish results (HSI)

Metadata creation is also important in order to catalogue the data for rapid comparison with previously learned data sets.

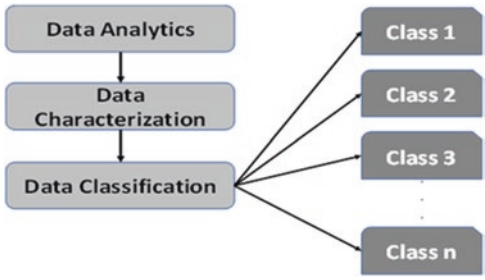


Fig. 13.2 The BDAP data characterization and classification process

Once the metadata are created, a pair-wise classification process is utilized to understand how the new data sets are compared to previously processed data, and their similarities and differences are computed and questions are generated for a hypothesis-driven, abductive analysis.

13.4 Feedback-Driven Analysis/Classification

A feedback-driven process will be provided that incorporates changes in prioritization and operator-feedback to the classification algorithms to provide a human-mentored software process that learns and adapts as data environments change, as prioritization changes, and as more information is gathered through continued processing and analysis of continuous streams of data. Figure 13.3 illustrates this proposed process.

Fuzzy membership functions are utilized to classify the data sets and allow state transition prediction to understand how likely the data are to change classifications. Much of the data analyzed for big data analytics are complex, stochastic data that can change rapidly over time (e.g., weather data, crowd sourcing data). A state transition analysis is performed with an abductive, hypothesis-driven analytical process that will be explained later.

13.5 State Change Prediction Process

Once the fuzzy membership designation has been determined and the metadata verified, a prediction process is utilized to understand the potential for the data environments to change state. A hypothesis-driven process will be utilized to analyze the results to predict whether a state change is likely. Examples of state changes would be a change from a tropical storm to a hurricane, or a likely state change of a

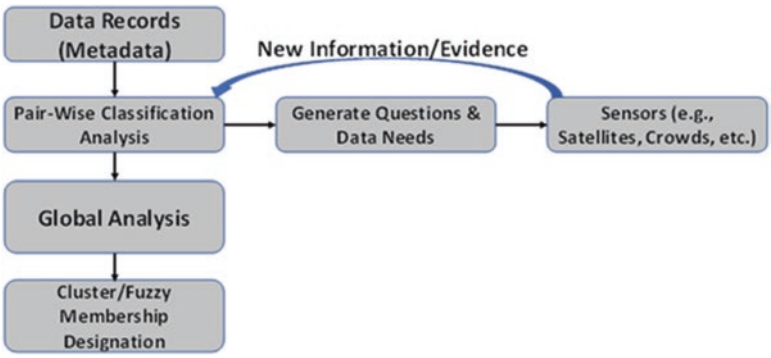


Fig. 13.3 BDAP feedback-driven data analysis/classification

crowd from agitated to mob-mentality state, or crowdsourcing state change between possible choices being posed to the "crowd." Figure 13.4 illustrates this proposed state transition process.

Within the state transition prediction process is a hypothesis-driven Martingale state transition prediction engine that analyzes the data sets using stochastic derivatives to understand the volatility of the data sets [8].

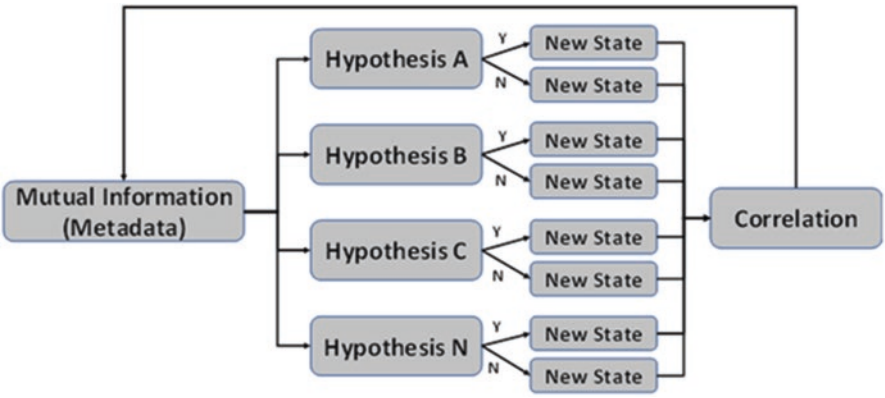
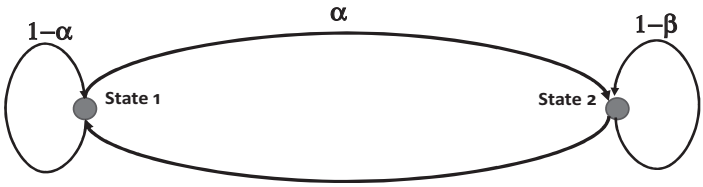


Fig. 13.4 BDAP data classification and state transition prediction process



$$P_{TR} = \begin{bmatrix} 1 - \frac{\sum_{i=1}^{n} \alpha_i}{n} & \frac{\sum_{i=1}^{n} \alpha_i}{n} \\ \frac{\sum_{i=1}^{n} \beta_i}{n} & 1 - \frac{\sum_{i=1}^{n} \beta_i}{n} \end{bmatrix}$$

$$a_i = State\ 1\ fuzzy\ memberships$$
$$\beta_i = State\ 2\ fuzzy\ memberships$$

$$\mu_{i1} = {}^{\beta_i}/_{\alpha_i + \beta_i}, \qquad \mu_{i2} = {}^{\alpha_i}/_{\alpha_i + \beta_i}$$
$$\mu_{i1} + \mu_{i2} = 1$$
$$\frac{\sum_{i=1}^{n} \mu_{i1}}{n} + \frac{\sum_{i=1}^{n} \mu_{i2}}{n} = 1$$

- When the summed average of all the State 1 individuals fuzzy memberships are greater than ½, then a change of state to State 2 is imminent.
- When the summed average of all the State 2 individuals fuzzy memberships are less than ½, then a change of state back to State 1 is imminent.

Fig. 13.5 BDAP stochastic process state change detection

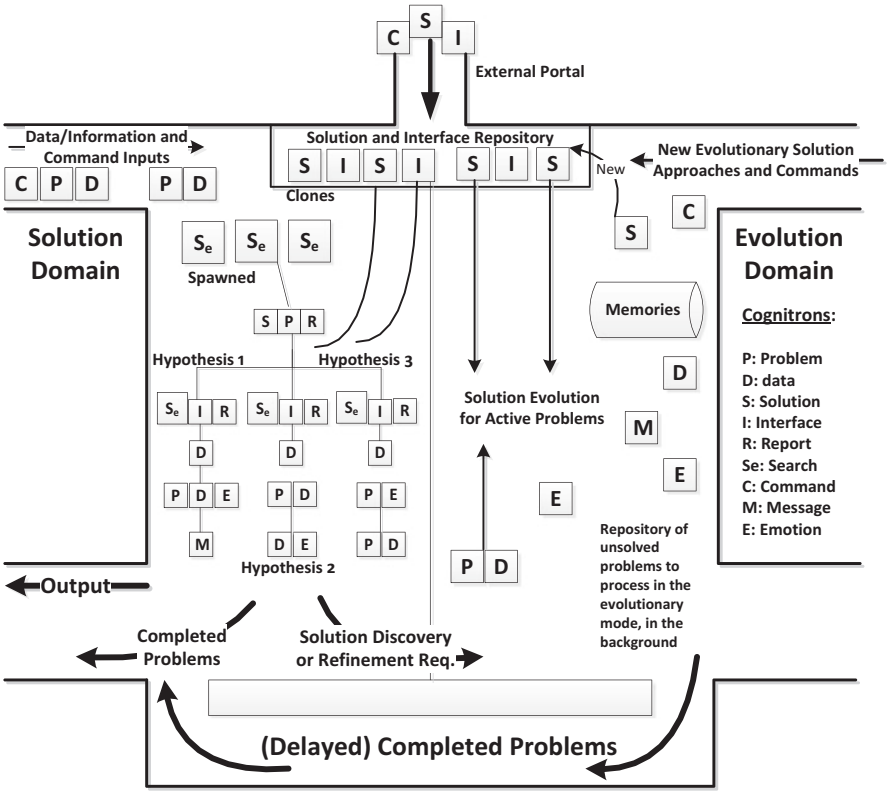


Fig. 13.6 BDAP analytical solution building process

13.6 Hypothesis-Driven Prediction/Classification Process

As discussed above, we are researching a hypothesis-driven analytical process that allows adaptation. The proposed *State Transition Prediction* (STP) process is driven by a Markov transition calculation illustrated in Fig. 13.5. Here the probability of state transition is computed from the average of fuzzy membership functions. Each can be weighted by a priority (from 0 to 1) that will affect the overall scoring.

As changes happen, from changes in data environments, changes in mission needs, changes in priorities, or other factors that may affect the outcomes. Figure 13.6 illustrates the classification solution domain process. Here we propose employing soft-computing techniques to generate possible solution hierarchies, based on the stochastic calculus methods and analytical processes discussed above. Here solutions take the form of answering questions and explaining situations/observations [9]. In Fig. 13.5, $\boldsymbol{P_{TR}}$ represents the probability of a state transition, computed using the summations of fuzzy membership functions which represent the conditions that drive transitions.

These structures are intended to:

- Assist operators with data analytics and decision support
- Provide automation, control, and analysis of the sensor/data acquisition network
- Assist operators in finding, filtering, and prioritizing solution information
- Enabling automation and control for finding, processing, learning from, and providing data characterization and classification, pattern recognition, predictions and recommendations based on the data processing procedures discussed above [10].

In addition to providing methodologies for understanding data transitions and data state changes, the process outlined in Fig. 13.5 can be utilized to measure procedural and episodic memory changes within an artificial intelligent system. By adjusting the fuzzy thresholds that measure state change within the memory system and correlating computational resource change and cognitive volatility within the system (cognitive velocity), it is possible to measure memories being created without the cognitive activity that should be created to invoke the memory system and store memories (detection if implicit learning and subsequent implicit memories).

The processing environment/algorithms proposed here utilize fuzzy logic [11] to integrate diverse sources of information, associate events in the data, and make observations. When combined with a dialectic search, the application of hybrid computing promises to revolutionize information processing and big data analytics. The dialectic search seeks answers to questions that require interplay between doubt and belief, where our knowledge is understood to be fallible [10]. This "playfulness" is key to hunting within information and implemented by the Dialectic Search Argument (DSA). This is very useful to assess sensor network readings for various types of data analysis. The Dialectic Search Argument (DSA), illustrated in Fig. 13.7, has four components derived from a Toulmin argument structure [12]:

1. Data: in support of the argument and rebutting the argument.
2. Warrant and backing: explaining and validating the argument.

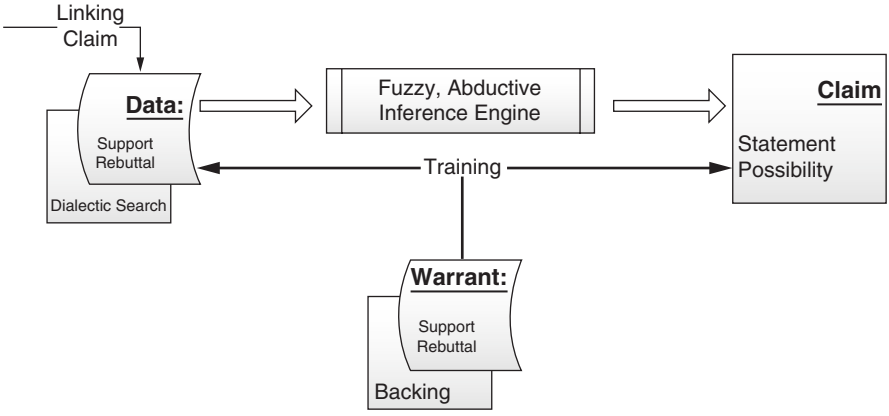


Fig. 13.7 The BDAP dialectic search argument process

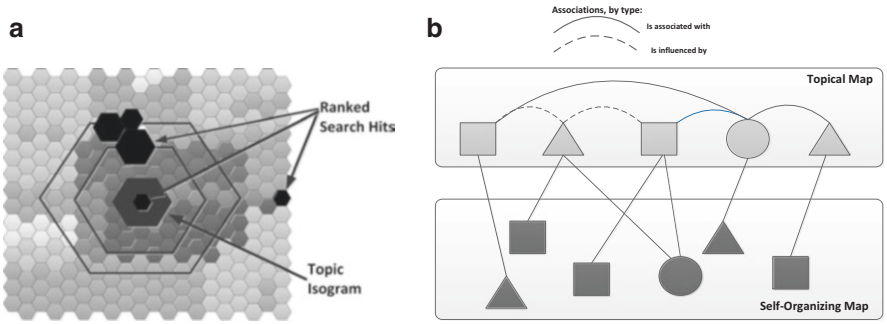


Fig. 13.8 (**a**) The fuzzy self-organizing map, (**b**) The semantical topical map

3. Claim: defining the argument itself.
4. Fuzzy inference: relating the data to the claim/classification.

The argument serves two distinct purposes. First, it provides an effective basis for mimicking human reasoning. Second, it provides a means to glean relevant information from the self-organizing map illustrated in Fig. 13.8 and transform it into actionable intelligence (practical knowledge.) These two purposes work together to provide a comprehensive data analytics system that allows the algorithms to sort through diverse information and find clues [13].

The fuzzy semantic self-organizing topical map (FSOM) is a method for analyzing and visualizing complex, multidimensional data. It consists of two parts. (a) A semantic SOM organizes inputs into categories used to encode the inputted information as a histogram. An information map contains contextual information. The information map is self-maintaining and automatically locates inputs. The isograms denote how close the hits are to specific information topics. (b) The FSOM can be enhanced to include a topic map. The topic map is the ISO standard for indexing and describing knowledge structures that span multiple sources. The key features are the topics, their associations, and their occurrences in the FSOM. The topics are the areas on the FSOM that fall under a topic name. The associations describe the relationships between topics. The occurrences are the links from the FSOM into the data sources used to form the FSOM. The value of superimposing a topic map onto the SOM is that it defines the information domain's ontology. It also enables rapid and sophisticated dialectic searches.

This proposed approach is considered dialectic in that it does not depend on deductive or inductive logic, though these may be included as part of the warrant. Instead, the DSA depends on non-analytic inferences to find new possibilities based upon warrant examples. The DSA is dialectic because its reasoning is based upon what is plausible; the DSA is a hypothesis fabricated from bits of information. Once the examples have been used to train the DSA, data that fits the support and rebuttal requirements is used to instantiate a new claim. This claim is then used to invoke one or more new DSAs that perform their searches. The developing lattice forms the

reasoning that renders the intelligence lead plausible and enables measurement of the possibility.

As the lattice develops, the aggregate possibility is computed using the fuzzy membership values of the support and rebuttal information. Eventually, a DSA lattice is formed that relates information with its computed possibility. The computation, based on Renyi's entropy theory, uses joint information memberships to generate a robust measure of possibility, a process that is not achievable using Bayesian methods. Whereas the topic map builds and maintains itself, the dialectic search requires supervised training, meaning it must be seeded with knowledge from a domain expert. However, once seeded, it has the potential of evolving the warrant to present new types of possible leads based upon evolving threats, systems, and missions.

There is one other valuable attribute to using the FSOM method. Because the vector that represents the information is randomly constructed, it cannot be decoded to reformulate the source; the source must be reread. This is critical to protecting compartmentalized information. Using the FSOM, the protected source can be included in the FSOM and used to support/rebut an argument without revealing the detailed information [14].

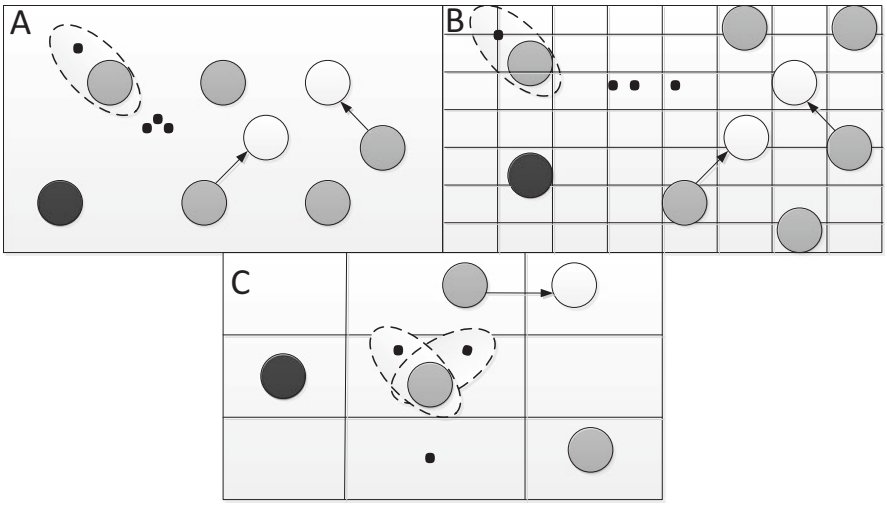


Fig. 13.9 Stochastic diffusion, (**a**) Off-lattice state change, (**b**) Micro-granular state change, (**c**) Macro-granular state change

13.7 Stochastic Diffusion Method for State Classification

To provide the data analytics with the ability to understand and quantify the utility of given sensors or inputs (e.g., crowdsourcing data sources), we propose to enable a stochastic diffusion process; we can re-prioritize resources based on given or implied priority tasks and utility characteristics, and combined perceptions of system objectives. Figure 13.9 illustrates this proposed stochastic diffusion process, which is like molecular biology, using a 2-dimensional or 3-dimensional computational mesh either of a lattice or spatially aware non-lattice to represent a processing membrane (physical membrane in biology) that represents sensor activity and interactions within the processing system [15]. We can characterize and simulate real-time and non-real-time changes within the sensor or data acquisition network, based on learning and inference characteristics [16]. This allows the data analytics to prioritize sensor resources to use, to address specific kinetic system activity (how quickly things are changing, and which sensors are most appropriate for a specific type of sensor and/or data). Here the stochastic diffusion may re-prioritize sensors or data acquisition methods or change which sensors or data sources are utilized within the overall data analytics process, based on situational awareness of the mission, the type of data, and/or the overall utilities of each sensor. Stochastic diffusion will allow uncertainties to be measured and quantified using continuous statistical analysis of the sensor/data acquisition network, adjusting sensor and data source usage, to minimize uncertainties in real time as it assesses patterns and activity across the sensor network.

13.8 Conclusions and Discussion

This is very preliminary work and much more research is required. Here we discuss a high-level view and discussion of the possibility of a Big Data Analytical Process (BDAP) system with the capabilities for efficient and accurate analysis, classification, cataloguing, and correlation of large, complex, heterogeneous data sets required for modern artificial intelligent systems. The next steps are to complete out the architecture, derive the requirements, and prototype the system. Future books will present the progress and results as they are available.

References

1. Crowder, J., & Carbone, J. (2011). *The great migration: Information to knowledge using cognition-based frameworks*. New York: Springer Science.
2. Scally, L., Bonato M., & Crowder, J. (2011). Learning agents for autonomous space asset management. In *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, HI.

3. Crowder, J. (2010). Flexible object architectures for hybrid neural processing systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
4. Crowder, J., & Friess, S. (2010). Artificial neural diagnostics and prognostics: Self-soothing in cognitive systems. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
5. Crowder, J., & Friess, S. (2010). Artificial neural emotions and emotional memory. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
6. Crowder, J., & Friess, S. (2011). Metacognition and metamemory concepts for AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
7. Raskin, V., Taylor, J. M., & Hempelmann, C. F. (2010). Ontological semantic technology for detecting insider threat and social engineering. In *New Security Paradigms Workshop*, Concord, MA.
8. Siminelakis, P. (2010). *Martingales and stopping times: Use of martingales in obtaining bounds and analyzing algorithms*. Athens: University of Athens.
9. Crowder, J. (2010). Operative information software agents (OISA) for intelligence processing. In *Proceedings of the 12th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
10. Crowder, J., & Carbone, J. (2011). Occam learning through pattern discovery: Computational mechanics in AI systems. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
11. Zadeh, L. (1975). *Calculus of fuzzy restrictions. Fuzzy sets and their applications to cognitive and decision processes* (pp. 1–39). New York: Academic Press.
12. Kennedy, X. (2006). *Reasoning. The Bedford reader* (9th ed., pp. 519–522). New York: St. Martin's Press.
13. Crowder, J., & Carbone, J. (2011). Recombinant knowledge relativity threads for contextual knowledge storage. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.
14. Taylor, J. M., & Raskin, V. (2011). Understanding the unknown: Unattested input processing in natural langua*ge*. In *FUZZ-IEEE Conference*, Taipei, Taiwan.
15. Ermak, D., & Nasstrom, J. (1998). *A Langrangian stochastic diffusion method for inhomogeneous turbulence*. Livermore, CA: Lawrence Livermore National Lab.
16. Crowder, J., & Friess, S. (2011). The artificial prefrontal cortex: Artificial consciousness. In *Proceedings of the 13th Annual International Conference on Artificial Intelligence*, Las Vegas, NV.

Chapter 14
Conclusions and Next Steps

Although we feel we have made a great start and have covered much about how to test artificial intelligent systems at various levels, much research and development is needed. One major topic of research to be continued is to understand the variables that can be used to control learning performance and explicit knowledge in the context of human interaction with artificial intelligent entities [1]. The relationship between implicit and explicit modes of learning and implicit and explicit types of knowledge has not been established and must be explored before we put artificial intelligent systems into long-term service either within the Department of Defense or the private, commercial sector. The relationships between decision and action may be critically influenced by implicit learning and knowledge and we need to understand how implicit vs. implicit learning and knowledge affect the ability of systems to learn and act effectively and correctly.

One area that has become more and more important over the last decade is the field of "Artificial Psychology" [2]. Data from studies of human cognitive development indicate that continuous learning drive changes in knowledge and performance, due to the subjects continued interaction with their environment(s), maturation of neural pathways [3]. If we create complex learning and reasoning systems/architectures to provide a continuous, life-long learning environment for artificial intelligent entities, we can expect similar results. These modifications to the neural patterns of an artificial intelligent entity will, we feel, cause the networks to become more powerful and involve implicitly defined knowledge representations [4]. One method of testing we will explore is the use of neural nodal cluster analysis to examine how neuronal weightings are changing as the artificial entity learns and adapts, in conjunction with measurement of the cognitive velocity (i.e., how fast) the neural connections and neural creations are occurring. In general, we look to measure, given a set of inputs to be learned:

1. The responses of the hidden layer neuronal activations.
2. The level of mutual information (grouping) between neurons within the network.

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3_14

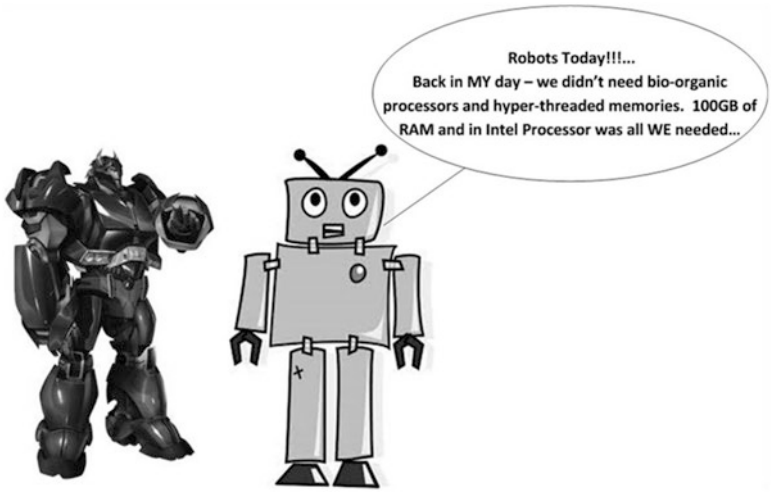


Fig. 14.1 Back in my day

14.1 More Complicated Is Not Necessarily Better

As we push for faster, more complicated and denser processing capabilities, we need to step back and examine whether more complicated means better answers. We need to examine biological evolution and how insects and other forms of life use very simple neural structures to complete very complicated tasks and develop complex communication skills with very few neurons (at least compared to humans). The cartoon below (Fig. 14.1) illustrates this concept. First and foremost, we need to understand what we are trying to accomplish with artificial intelligent entities.

What types of knowledge are required of the artificial intelligent entity? What types of decisions will it be required to make or provide decision support? What actions will it be required to take and what are the desired outcomes? Computational formulation and testing of this type of information will be crucial to the success of artificial intelligent systems. One question we must understand, which prompted the inclusion of integrated system health management within an artificial intelligent entity: What happens when an artificial intelligent entity's memory system experiences failure [5]. What happens when all or part of the memory electronics fails? Figure 14.2 below is a whimsical look at this.

In short, we need to be concerned with both what the artificial intelligent entity can do and what it knows. The acquisition of knowledge, as we have discussed in detail, involves both explicit and implicit learning, both of which need to be clearly understood and measurable [6].

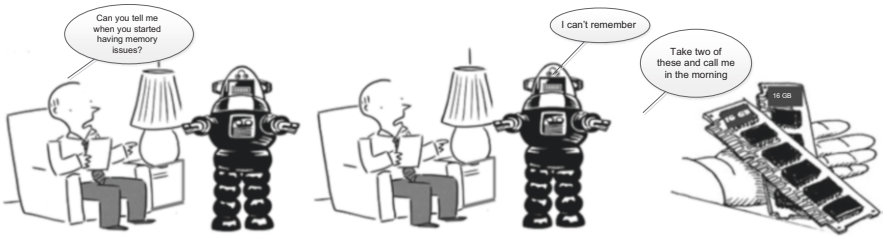


Fig. 14.2 I can't remember

14.2 Where Are We Going?

As we go forward, ever toward what may become artificial consciousness, we must continue to develop an understanding of artificial intelligence, and the issues of functional learning and reasoning and how they relate to qualitative human consciousness [7]. Artificial psychology experiments must be created in order to understand the complete artificial intelligent entity. This book is a glimpse into the future of artificial intelligence and how we begin to think about adequately testing an artificial system that thinks, reasons, and infers "like humans."

Creating and testing a SELF has significant technical challenges which are addressed throughout the book; however, there are also adjacent cultural challenges which also need to be addressed as we move forward with integrating artificial intelligent systems into society. We must also understand how such a system would be received and perceived by people and how we expect any type of artificially intelligent system to react to and perceive people. That necessitates research and study of the concept of artificial psychology that will deal with what it means to have a SELF resemble human intelligence, the when and why of the "psyche" of an artificial intelligent entity.

14.2.1 Artificial Psychology

While psychology is the study of mental processes and behavior of individuals, artificial psychology is the study of the synthesized mental processes of the SELF like humans and the artificial cognitive processes required for an artificial intelligent entity to be self-adapting. In psychology there are several specialties or focused areas of study. One example is cognitive psychology that studies how the brain thinks and works. This includes learning, memory, perception, language, and logic. Developmental psychology considers the developmental stages in which an individual develops and what is appropriate to consider normal/standard for a human based upon these stages of development. Sports psychology considers mechanisms specifically to affect individual performance and how performance affects the individual. Hence, artificial psychology involves the artificial mental process considered necessary to create a SELF.

14.2.2 Artificial Psychology as a Discipline

Artificial Psychology is a theoretical discipline first proposed by Dan Curtis in 1963. This theory states that as artificial intelligence approaches the complexity level of human intelligence it will meet three conditions that will necessitate creation of the formal social science of "artificial psychology":

- Condition 1: The SELF makes all its decisions autonomously and can make decisions based on information that is (1) New, (2) Abstract, and (3) Incomplete.
- Condition 2: The SELF can adapt or change its own programming, based on new information, and can resolve its own programming conflicts, even in the presence of incomplete information.[1]
- Condition 3: Conditions 1 and 2 are met in situations that were not part of the original SELF's initial programming.

We are fast approaching the engineering, bioinformatics, and computational science where scalable processing power and real-time processing can perform operations to levels where not only can the three conditions be met, but that a SELF can be created. In addition, we believe soon we will be able to create a SELF that can reach conclusions based upon newly acquired information, can infer upon it from learned and store information in the form of synthetic memories. Therefore, we believe that enough criteria may exist, giving significant credence to the growing field of artificial psychology [8]. This may require new theories and research to be explored in industry and at institutions of higher learning, specifically for addressing the rapidly expanding need for general human support systems to domains and environments where humans are still significantly challenged (e.g., deep-space and deep-sea exploration). The formal social science of artificial psychology will be required when the abilities of a SELF reach self-adaptation, allowing self-analysis and decisions based on information available through its sensors and resolution of any internal inconsistencies within the SELF (self-reflection). Examples of why we are going to need artificial psychology as a hard discipline are [9]:

- 2016, an on-line Microsoft chatbot had to be removed when, in less than 24 h, Twitter users conversing with the on-line bot turned her into a Nazi racist.
- 2017, German police break into a house after several complaints of loud music being played. It turns out that the homeowner's Amazon Alexa was playing loud music on its own without any direction from its owner. Alexa was basically hosting a party of her own.
- It was discovered that PokemonGo stops were being placed in predominantly white neighborhoods.

These incidents, and many more, drive home the care that must be taken in how artificial intelligent systems are trained and tested. One major issue that continues

[1] This means that the SELF autonomously makes value-based decisions, referring to values that the SELF has created for itself.

to be seen in the training of artificial intelligent system is that having an artificial intelligent entity learn from people turns badly often. We believe this is due to underlying implicit learning and subsequent implicit knowledge that the artificial intelligent entity picks up from dealing with humans. It is almost impossible for a human to spend any serious time communicating with an artificial intelligent system without the human's biases and opinions coming into play. How we design the training for an artificial intelligent system radically affects its future learning, reasoning, and thinking. It is impossible to "test" an artificial intelligent system without its learning and adapting to the test data, just like it learns and adapts to the training data. Again, this book is but the beginnings of an entirely new field in artificial intelligence.

References

1. Berry, D., & Broadbent, D. (1988). Interactive tasks and the implicit-explicit distinction. *British Journal of Psychology, 79*, 251–272.
2. Crowder, J., & Friess, S. (2012). Artificial psychology: The psychology of AI. In *International Multiconference on Complexity, Informatics and Cybernetics*, Orlando, FL.
3. Clark, A., & Karmiloff-Smith, A. (1993). The cognizer's innards: A psychological and philosophical perspective on the development of thought. *Mind and Language, 8*(4), 487–519.
4. Cleeremans, A. (1994). Attention and awareness in sequence learning. In *Proceedings of Cognitive Science Society Annual Conference* (pp. 330–335).
5. Keele, S., Ivry, R., Hazeltine, E., Mayr, U., & Heuer, H. (1998). *The cognitive and neural architecture of sequence representation* (Technical report No. 98-03). University of Oregon.
6. Lewicki, M., & Hoffman, H. (1987). Unconscious acquisition of complex procedural knowledge. *Journal of Experimental Psychology: Learning, Memory and Cognition, 13*(4), 523–530.
7. Sun, R. (1997). Learning, action, and consciousness: A hybrid approach towards modeling consciousness. *Neural Networks, 10*(7), 1317–1331.
8. Stone, P. (2016). *Artificial intelligence and life in 2030. One hundred year study on artificial intelligence: Report of the 2015–2016 study panel*. Stanford, CA: Stanford University.
9. Tchopp, M. (2018). *Psychology of artificial intelligence: Foundations, range and implications from a humanities perspective*. Medium Corporation. Retrieved from https://medium.com/womeninai/psychology-of-artificial-intelligence-ca0f0a9f3d7c.

Index

J. A. Crowder et al., *Artificial Psychology*,
https://doi.org/10.1007/978-3-030-17081-3