RJ10522 (ALM1406-008) June 18 2014
Computer Science

IBM Research Report

Cognitive Systems

Jeffrey T. Kreulen
IBM Research Division
Almaden Research Center
650 Harry Road
San Jose, CA 95120-6099
USA

Cognitive Systems

Jeffrey T. Kreulen, PhD
kreulen@us.ibm.com

Introduction

The IBM Research SyNAPSE project [1] defines cognitive systems as "the quest for approximating the mind-like function, low power, small volume, and real-time performance of the human brain."

IBM [2] has described cognitive systems as a new generation of computing systems characterized as data-centric, designed for statistical analysis, scale-in, and automate system and workload management with exploration in core technologies, architectures and applications.

Dr. Kelly [3] has described a new era of cognitive systems characterized by learning from interactions and generating insight from data, becoming ubiquitous and pervasive while evolving from command driven to advice offering.

Rob High [4] asserts that cognitive systems are able to learn their behavior through education; they support forms of expression that are more natural for human interaction; their primary value is their expertise; and they continue to evolve as they experiences new information, new scenarios, and new responses; and does so at enormous scale.

Cognition has its roots in the meaning of "to know", it is said to reside in the mind and be capable of judgment. In this paper, we propose a theoretical framework that uses human cognition as a model to explain a continuum of computational systems, referred to as cognitive systems. Cognitive systems are adaptive and intelligent systems that enhance information organization, learning and understanding for the benefit and augmentation of people, business and the world.

First, the theoretical framework begins with eight elements of cognition, which make up the mind-like function mentioned by [1]. Second, a set of capability metrics are introduced to differentiate and evaluate cognitive capabilities in multiple important dimensions. Third, architectural alternatives for implementing the elements of cognition are surveyed. Lastly, we conclude with some observations.

Elements of Cognition

In this model, cognition consist of eight elements:

- ***Perception*** – Senses and sensors are the inputs, they provide the external observation space. Focus is used to characterize both the clarity and contextual value (interestingness) of collections of inputs.

- ***Actuation*** – Actions or behaviors are the outputs and point of interaction with the external. Behaviors have form, content and function. Behavioral physical forms include waves, electrons, light, sound, and physical objects. Behavioral content is exemplified in language. Behavioral function ranges from the intentional to the automatic.

- ***Memory*** – There exists both short-term and long-term memory, which is capable of encoding, storing and retrieving information in both raw and conceptualized encodings.

- ***Conceptual Learning*** – Concepts or entities are represented as a high dimensional feature-space with multiple encodings. Learning is achieved by continually comparing and calibrating concepts relative to internally and externally generated stimuli. Comparison is implemented with a variety of algorithms, heuristics and rules. The implementation is directly related to the range of behavioral functions in actuation. A special class of entity is time, and is integral to the cognitive feedback loop.

- ***Conceptual Logic*** – Concepts are related to each other in a variety of ways. Multiple logics can express the principals of the detected relationships. Entity relationships provide the semantics. A special class of conceptual logic is mathematics. Conceptual logic can be encoded in a variety of ways, the most abundant examples being text and data. Conceptual logic enables cognition to be meta-aware, which means that logic applies to both specific instances as well as generalized abstractions through a process referred to as reflection. Inductive logic uses knowledge of instances to probabilistically infer generalizations. Deductive logic uses knowledge of generalizations to precisely infer for instances.

- ***Problem Solving*** –Problem solving consists of goals, processes and strategies. Goals can be for a variety of purposes, for example personal, business, or a community. Those for which resources are applied are said to get attention. The problem solving process, referred to as reasoning, involves a loop with four steps: hypothesis, analysis of alternative strategies, execution of the strategy, and evaluation of the results. Typical problem solving strategies include:
    - Find a pattern using a list, table, graph, diagram or picture.
    - Use trial and error or experimentation.
    - Use a model. Model classes include optimization, prediction, and classification.
    - Use an object or tool.
    - Use language, mathematics and logic. Sometimes just the process of output and input is sufficient to make progress on the problem. The output can also be perceived by other cognitive automata, which can contribute to the problem solving as well. This is sometimes referred to as crowdsourcing.
    - Work backward or forward.
    - Simplify.
    - Question your assumptions.
    - Use scarcity and competition.
    - Use environmental adaptation.
    - Use time.
    - . . .

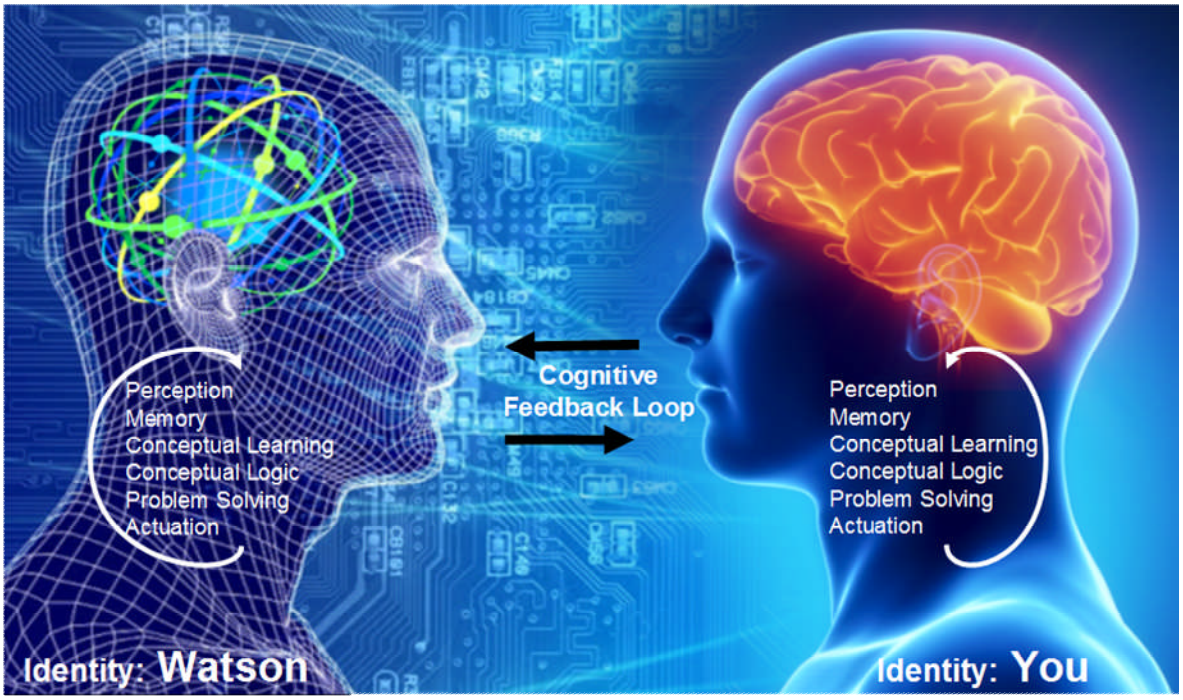

Figure 1: Cognitive Feedback Loops

- ***Cognitive Feedback Loop*** – In this model of cognition, there are two feedback loops. The first is the external feedback loop (Figure 1 in black), which begins with perception, followed by memory, conceptual learning, conceptual logic, problem solving and action. The external loop is introduces new information that is process into concepts, comparison algorithms, logics and problem solving techniques. In addition to new information, it is also used for calibration and error correction. The second is the internal feedback loop (Figure 1 in blue). The internal loop is uses memory, conceptual learning, conceptual logic and problem solving independent of action and perception.

- ***Identity*** – The identity is connected to an embodiment, which could be implemented in carbon, silicon or other physical structures. The identity is where choice and responsibility reside. Choice or judgment can be driven by a variety of factors including values, preservation, reward, creativity, emotion, and personality. Some of the factors exhibit non-determinism, some of which may just be a lack of current understanding. Responsibility is both the cognitive accumulation and the economic consequence.

A computer scientist's perspective would be to think of the elements of cognition in an object model (Appendix A: Cognitive Object Model). In this model, Identity is a class, which can be instantiated. Each Identity object has variables (memory and concepts) and can implement methods (logic for problem solving) to operate on them. Each identity object has a main function that implements the outer and inner cognitive feedback loop.

Capability Metrics

The single most common qualitative description for human cognitive ability is intelligence. However, the ability to measure intelligence precisely has been elusive. Undeterred, we propose the following metrics. These metrics provide for a rich set of tradeoffs in implementation and ability. The metrics are:

- ***Adaptability*** – Adaptability is a measure of breadth and the ability to add new memory, conceptual learning, conceptual logic and problem solving to adapt to new or different circumstances. Adaptability can be measured by the number of application domains for which a cognitive identity is proficient. It has also been characterized by the continuum of static, dynamic and autonomic systems.

- ***Problem Complexity*** – Problem complexity is a measure of depth. This can be thought of in two ways: information theoretical and/or psychometric intelligence. Information theoretical complexity would be an aggregate complexity of the cognitive elements required to solve a given problem.

  Psychometric intelligence is most commonly measured with an Intelligence Quotient (IQ), for which there are several existing tests. These tests correlate with positive outcomes, this positive correlation is referred to as general intelligence (g). There are also theories of multiple intelligences, such as musical, linguistic, mathematical, and kinesthetic.

- ***Behavioral*** – Behavioral metrics can be categorized in four progressively more sophisticated capability levels: reactive, anticipatory, preventive, and proactive.

  The simplest reactive behaviors can be modeled with simple condition-action rules. Stigmergic behaviors incorporate memory with condition-action rules which enable specialization and coordination. Incorporating feedback enables reinforced learning which can be implemented with weighted condition-action rules with memory. Behaviors that have a built in expectation are useful for preparedness, typically initially to address imminent threats, are called anticipatory.

  Anticipatory behaviors can be implemented with what is called state-determined mechanisms, which require the addition of weighted condition-condition rules to weighted condition-action rules with memory. Higher order anticipatory behaviors are implemented with a network of state-determined mechanisms, which provides the ability to represent more complex concepts and used for comparison and similarity. The ability to determine similarity gives the ability to draw analogies.

  Preventive behaviors build on the reactive and anticipatory capabilities increasing resources to reduce risk through planning and beginning to incorporate social capabilities. Lastly, proactive behaviors build on preventive capabilities increasing resources, while incorporating goals and complex social behaviors for reflection and problem solving.

- ***Machine Complexity*** – Machine complexity is a measure of the aggregate complexity of the components in a machine that is used to solve a problem of given complexity. This could be measured in units such as transistors, megabytes, and/or algorithmic complexity.

- ***Power, Volume, and Response-Time*** – As outlined in [1] the ability to achieve the power, volume and response-time of the human brain are important measures to consider as we build cognitive systems.

- ***Productivity*** – Productivity is a measure of economic consequence per cost and time. In equation form: Productivity = ((Output * Value) / (Cost * Time)).

Architectures

There are multiple architectures capable of implementing a cognitive system, here we will survey the five most common.

- **Biological**
  There are biological architectures exemplified by the human brain. The human brain is not well understood and there are multiple efforts underway to study it through various reverse engineering methods. Examples of this include the Brain Activity Map in the US and the Human Brain Project in the EU.

- **Biology-inspired**
  A second class of architectures are biology-inspired or neuromorphic. IBM's SyNAPSE project is an example of this and is making significant strides towards an implementation with their TrueNorth chipset.

- **von Neumann**
  von Neumann architectures are ubiquitous in today's computers and are capable of implementing all of the elements of cognition.

- **Legal**
  There are legal architectures that are applicable to cognitive systems. In *Cognitive automata and the law* [4], Giovanni Sartor discusses the issue that people are given certain rights given their cognitive ability (or lack thereof) and that as machines become more intelligent, they can, and already are, entering into binding contracts on our behalf. So how intelligent does a machine have to be in order for this to be legally binding?

- **Distributed**
  We can think of cognitive systems as people, cognitive technology, or cognitive enterprise consisting of a combination thereof. At this point we enter the field of distributed cognitive systems. Topics such as the wisdom of crowds, memetic theory, and crowdsourcing all pertain to distributed cognitive systems.

Observations

In summary, we have presented a theoretical framework for cognitive systems consisting of eight elements of cognition. We have presented seven capability metrics, which can be used to evaluate the individual elements of cognition as well as the cognitive system as a whole. Lastly, we presented a range of architectures, which are being used to implement cognitive systems.

We conclude with four strategic observations:

- **Scaling Cognitive Ability**
  In order to scale cognitive ability for both problem complexity and adaptability, systems will need to scale the creation and maintenance of conceptual learning (entities) and conceptual logic (relationships, content and data), as well as problem solving (algorithms).

- **Cognitive Feedback Loop**
  Better understanding the patterns and modalities of this feedback loop will help make interactions more natural and is also important to scalability. This feedback loop is fundamental to interactions and dialog. Exemplary dialog patterns include: transfer (teacher-student), utilization (service), observation (knowledge), influence (marketing), calibration (social), and symbiotic (co-creation). Multiple asynchronous internal and external feedback loops enable the richness of reasoning and problem solving.

- **Cognitive Accumulation**
  Scaling also implies the ability to accumulate capabilities in each of the elements. The implication being that the cognitive enterprise should be actively accumulating perceptual, memory, conceptual learning, conceptual logic, problem solving and behavioral abilities in the domains that are important to them.

- **Cognition-as-a-Service**
  Cognitive systems can and will be deployable in a myriad of economic business models with as-a-service being the most interesting because of its economic efficiency and deployment practicality.

Acknowledgements

IBM, Cognitive Computing Study Team, Scott Spangler, Ying Chen, Jeff Welser, Laura Haas, Michael Karasick, John Kelly III, Paul Maglio, Melissa Cefkin, Jeanette Blomberg, Grady Booch, Pat Selinger, Susan Stucky, Jane Snowden, David Nahamoo, Rob High, Dharmendra Modha, Brent Hailpern, Anshu Jain, Bob Schloss, and more.

References

[1] "Compass: A Scalable Simulator for an Architecture for Cognitive Computing," D.Modha et al., SC12, 2012, http://www.modha.org/blog/SC12/SC2012_Compass.pdf

[2] "Cognitive Systems: A new era of computing", IBM Research, http://www.research.ibm.com/new-era-of-computing.shtml

[3] "Welcome to the Era of Cognitive Systems", John E. Kelly III, Smarter Planet Blog, 2012, http://asmarterplanet.com/blog/2012/05/welcome-to-the-era-of-cognitive-systems.html

[4] Presentation, IBM, Rob High, 2013

[5] "Cognitive automata and the law", G.Sartor, 2003, http://www.cs.uu.nl/jurix03/sartorpaper.pdf

[6] Cognitive Systems, Francis Heylighen, 2010, http://pespmc1.vub.ac.be/Papers/CognitiveSystems.pdf

Appendix A: Cognitive Object Model

This pseudo-code here is to exemplify an implementation of cognition in software. This implementation contains two synchronous threads, internal and external cognitive loops. In comparison to human cognition which appears to be multi-threaded and multi-synchronous.

```
public class Identity {
  String id;
  Sense senses[5];
  Memory memory[],concepts[],logics[],problems[],behaviors[];
  Objective goals[];
  Boolean alive = true;
  Time time = 0;

  Private static class InnerCognitiveLoop implements Runnable {
    Public void run() {
        //  internal cognitive loop
        concepts = ConceptualLearning(memory,concepts);
        logics = ConceptualLogic(memory,concepts,logics);
        problems = ProblemSolve(memory,concepts,logics,problems);
    } // run
  }  // class CognitiveLoop

  public static void main (String args[]) {
     while (alive) {
        Thread inner = new Thread(new InnerCognitiveLoop());
        inner.start();
        // external cognitive loop
        memory = Perception(senses);
        concepts = ConceptualLearning(senses,memory,concepts);
        logics = ConceptualLogic(senses,memory,concepts,logics);
        problems = ProblemSolve(senses,memory,concepts,logics,problems);
        behaviors = Actuation(senses,memory,concepts,logics,problems,behaviors);
        inner.join();
        time++;
    }  // while alive
  }  // main
} // class Identity
```

Appendix B: Cognitive Systems MindMap