Byoung-Tak Zhang
Seoul National University, KOREA

Hypernetworks: A Molecular Evolutionary Architecture for Cognitive Learning and Memory

Abstract: Recent interest in human-level intelligence suggests a rethink of the role of machine learning in computational intelligence. We argue that without cognitive learning the goal of achieving human-level synthetic intelligence is far from completion. Here we review the principles underlying human learning and memory, and identify three of them, i.e., continuity, glocality, and compositionality, as the most fundamental to human-level machine learning. We then propose the recently-developed hypernetwork model as a candidate architecture for cognitive learning and memory. Hypernetworks are a random hypergraph structure higher-order probabilistic relations of data by an evolutionary self-organizing process based on molecular self-assembly. The chemically-based massive interaction for information organization and processing in the molecular hypernetworks, referred to as hyperinteractionism, is contrasted with the symbolist, connectionist, and dynamicist approaches to mind and intelligence. We demonstrate the generative learning capability of the hypernetworks to simulate linguistic recall memory, visual imagery, and language-vision crossmodal translation based on a video corpus of movies and dramas in a multimodal memory game environment. We also offer prospects for the hyperinteractionistic molecular mind approach to a unified theory of cognitive learning.

I. Introduction

Imagine a system of a vast number of molecules floating and interacting in 3D liquid-state media. Imagine further that each molecule represents a memory fragment or a "molecular concept". Upon temperature control and enzymatic catalysis, the molecules replicate themselves, self-assemble into compounds, and disassemble into fragments with some "association reaction" error. In this molecular model of mind, a single memory fragment exists in multiple copies, where the strength of the memory trace is proportional to the number of molecular copies. The massive chemical interaction between the molecular concepts in the molecular mind, referred to as hyperinteraction (Box 1), can be formalized as a random hypergraph structure called hypernetworks.

Here, we investigate the use of hypernetworks for simulating human-like learning or cognitive learning. As suggested above, hypernetworks were originally proposed as a parallel associative memory model [77], [44], [33], [29] inspired by biomolecular networks and realized in chemical self-assembly using DNA computing [90], [91], [84]. A hypernetwork consists of a large number of random hyperedges (molecular concepts), each of which combines vertices (atomic concepts) of arbitrary size and thus is able to encode higher-order interactions among the concepts. In language modeling, this combinatorial property can be used for learning the higher-order associations of the words from a text corpus. In visual memory modeling, a hypernetwork represents an internal structure of an image as a mixture of higher-order combinations of the visual words in the image.

The hypernetworks have a fluidic, reconfigurable molecular structure and, thus, are learning-friendly. Both their structure (hyperedge compositions) and parameters (hyperedge weights) are learned by molecular evolutionary processes using the primitive operations of matching, selection, and amplification of hyperedges. Each primitive operation is performed by massive interaction and self-assembly of hyperedge molecules. The chemical nature of

1556-603X/08/$25.00©2008IEEE

information processing in the hypernetworks provides an interesting analogy to the mental chemistry model of mind suggested by John Stuart Mill more than a century ago [7] and, more recently, the chemical machine model of mind used in biological psychiatry [1]. The evolutionary thinking of population-based memory formation and learning in hypernetworks has also some similarity to the global brain theory of neural Darwinism [11].

The goal of this article is to introduce hypernetworks as a cognitive architecture for learning and memory and to suggest the use of molecular evolution as an effective method for learning the cognitive hypernetworks. Recently, there has been much discussion about human-level intelligence, i.e., creating high-performance machine intelligence based on the cognitive substrate of human intelligence [56], [22], [6], [17], [46], [8]. We believe human-level intelligence requires human-level learning capability, and learning-friendly cognitive architectures such as hypernetworks are an important step toward human-level machine intelligence.

This article is organized as follows. In Section II, we review the previous studies in artificial intelligence and cognitive brain science to derive the most important organizational principles for cognitive learning in achieving human-level intelligence. In Section III, we describe the hypernetwork model and examine its properties by simulating cognitive learning and memory phenomena. In Section IV, we demonstrate that the hypernetworks can learn to solve the image-text crossmodal translation problems, such as generating a text dialogue for a given movie scene image. We discuss why these kinds of recall-memory learning tasks are hard to simulate with existing machine learning models and how these kinds of cognitive tasks are facilitated in hypernetworks. We also investigate the convergence behaviors of evolutionary hypernetworks for continuous, lifelong learning situated in a noisy environment. Section V suggests future research directions toward human-level machine learning and intelligence.

II. Cognitive Learning

A. Toward Human-Level Intelligence

Humans are creative, compliant, attentive to change, resourceful, and able to take a variety of circumstances into account [57]. In comparison to machines, however, humans are imprecise, sloppy, distractable, emotional, and illogical. To achieve human-level intelligence these properties should be taken into account. It is also important to understand how human intelligence is developed and works. In this subsection we summarize some of the important findings from artificial intelligence and cognitive brain research.

Humans are *versatile* and can come up with many new ideas and solutions to a given problem [55]. Machines are good at solving a specific problem that they are designed to deal with. But, they are brittle outside the scope of the problem domain. Humans are not just reactive, they are proactive and imaginative. Human brains produce some illusions like confabulation [30] and synaesthesia [64], but these are also sources of imagination and metaphors. Complexity, such as creativity and intelligence, emerges from simple elements [31].

Imagination requires *recall memory* [12]. During recent decades, researchers have revealed how memory is organized and works at the cellular and molecular level [38], [73], [75]. Recently, neural-net researchers have also emphasized the importance of memory architecture in technical problem solving, such as visual pattern recognition and language processing [25], [28], [81].

Human intelligence develops *situated* in *multimodal* environments [21]. The brain is constantly active, perceiving and acting with each evolving situation. It is interesting to note that situated or embodied intellects may solve problems more easily than isolated or disembodied intellects. This seems counter-intuitive since, for machines, embodiment or more modalities may mean more noisy data and more requirements for computing power. However, intelligent behavior in a disembodied agent requires a tremendous amount of knowledge, lots of deep planning and decision making, and efficient memory storage and retrieval [57]. When the agent is tightly coupled to the world, decision making and action can take place within the context established by the sensory data from the environment, taking some of the memory and computational burden off the agent.

The human mind makes use of *multiple* representations and problem-solving strategies. The brain consists of functional modules which are *localized* in cortical or subcortical areas and specialized to perform some specific cognitive functions. These are known by different names such as cell assemblies [27], microcircuits [23], or schemas [78]. But these modules are widely *distributed* and work together on the whole-brain scale [43], [19], [58]. For example, words are represented and processed in the brain by strongly connected distributed neuron populations exhibiting specific topographies [63]. This is why human decision making is robust in a wide range of domains.

One important issue in achieving human-level intelligence is how to *integrate* the multiple tasks into a coherent solution [36], [76]. It is conjectured that language is organized in terms of constructions, each of which integrates many aspects, such as meaning, context, affect, phonological form, and so on [15]. In brain science, this is an important part of the *binding* problem [98].

B. Learning in Humans and Machines

There is no more wonderful and necessary function of the brain than its ability to learn and to retrieve what is learned in the memory process [10]. Learning is a fundamental faculty of the human brain, and without learning it seems impossible to achieve human-level intelligence. Here we review the mechanisms and properties of human learning as compared to the current status of machine learning.

Learning in humans can be characterized in several ways. One classical, but still effective theory says that humans use three different types of learning: accretion, tuning, and restructuring [65]. Accretion is the accumulation of facts as in learning

Digital Object Identifier 10.1109/MCI.2008.926615

Box 1: From Mind to Molecules and Back

Human intelligence is a mental faculty of the mind (Figure 1). The mind has a physical basis in the brain which consists of 10^{11} neurons. The neurons are connected at the synapses where there are thousands of molecular species massively interacting with each other to build complex, biochemical signaling networks.

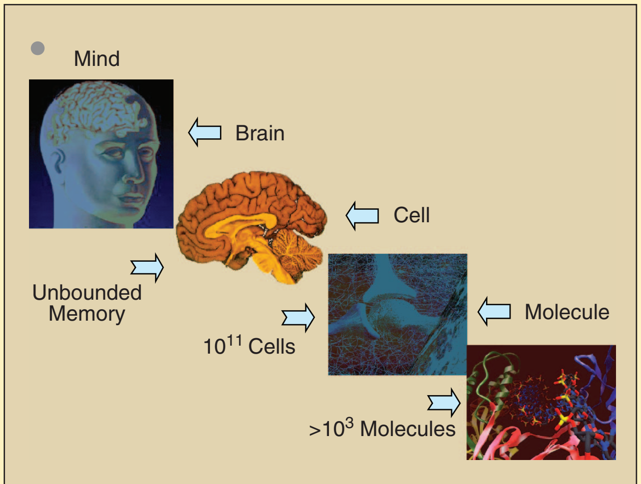


FIGURE 1 From mind to molecules and back. Human intelligence is a mental faculty of the mind. The mind can be viewed as an emergent phenomenon of complex networks of concepts generated from the memory of an "effectively unbounded" capacity on the neural basis of the brain. The brain consists of approximately 10^{11} neurons, each connected to 1,000–10,000 other neurons at synapses. The synapses are junction gaps between two neurons where thousands of molecular species, such as neurotransmitters, receptors, neuromodulators, and chemicals, build biomolecular networks of massive interaction.

There have been several paradigms proposed for understanding natural intelligence and constructing synthetic intelligence. Table 1 summarizes the characteristics and properties of four of these paradigms: symbolism, connectionism, dynamicism, and hyperactionism. (1) The symbolist paradigm is based on the metaphor of intelligence as symbol manipulation. It represents knowledge in modular, localized forms like predicates, rules, and procedures. Processing involves syntax-based sequential operations. (2) The connectionist paradigm models cognition as patterns of neural activity [14], [66]. Knowledge is represented in distributed connections of neural networks. Processing involves parallel propagation of electrical signals between neural units. (3) The dynamicist paradigm sees mind as motion [62], [18], [80]. Dynamical systems theory is used to describe the trajectory of the systems behavior in a state space. Conceptual knowledge is usually not represented explicitly. (4) The hyperinteractionist paradigm takes analogy from the biomolecular systems of massive and cooperative interaction [68], [39], [13] and the related unconventional computation paradigms [5], [59], [67], [86], [47]. Knowledge is represented as molecular micromodules assembled into a complex hyper-graph of combinatorial structure [85], [91]. Graph theory and probabilistic logic are used to describe and control the evolution of the systems behavior.

TABLE 1 Paradigms for computational intelligence.

	SYMBOLISM	CONNECTIONISM	DYNAMICISM	HYPERINTERACTIONISM
METAPHOR	SYMBOL SYSTEM	NEURAL SYSTEM	DYNAMICAL SYSTEM	BIOMOLECULAR SYSTEM
MECHANISM	LOGICAL	ELECTRICAL	MECHANICAL	CHEMICAL
DESCRIPTION	SYNTACTIC	FUNCTIONAL	BEHAVIORAL	RELATIONAL
REPRESENTATION	LOCALIST	DISTRIBUTED	CONTINUOUS	COLLECTIVE
ORGANIZATION	STRUCTURAL	CONNECTIONIST	DIFFERENTIAL	COMBINATORIAL
ADAPTATION	SUBSTITUTION	TUNING	RATE CHANGE	SELF-ASSEMBLY
PROCESSING	SEQUENTIAL	PARALLEL	DYNAMICAL	MASSIVELY PARALLEL
STRUCTURE	PROCEDURE	NETWORK	EQUATION	HYPERGRAPH
MATHEMATICS	LOGIC, FORMAL LANGUAGE	LINEAR ALGEBRA, STATISTICS	GEOMETRY, CALCULUS	GRAPH THEORY, PROBABILISTIC LOGIC
SPACE/TIME	FORMAL	SPATIAL	TEMPORAL	SPATIOTEMPORAL

new vocabulary or the spelling of already known words. Tuning is a slow process of practice requiring prolonged, laborious effort. It takes thousands of hours to become an expert pianist or soccer player. Restructuring is the process of forming the conceptual structures. It requires exploration, comparison, and integration of concepts.

However, current machine learning models do not fully exploit these types of learning. Neural networks and connectionist models focus on the tuning mode of learning. Most learning algorithms do not support the accretion mode of learning either. Learning is typically defined as a function approximation problem from a fixed set of training points [26], [9], [70]. The real accretion, however, in human learning is a long-lasting process of accumulation of knowledge. The weakest part of existing machine learning architectures is the lack of restructuring capability. Neither kernel machines [70] nor graphical models [4] offer any effective methods for incremental assimilation of new facts.

Aristotle identified three laws of association, in effect, three laws of learning: similarity, contrast, and contiguity [7]. Interestingly, most current machine learning algorithms are based on the principles of similarity and contrast. Unsupervised

The chemical nature of information processing in the hypernetworks provides an interesting analogy to the mental chemistry model of mind suggested by John Stuart Mill more than a century ago and, more recently, the chemical machine model of mind used in biological psychiatry.

learning methods are algorithms to search for similarities between objects. Supervised learning methods, especially classification algorithms, try to maximize the contrast between the classes. Contiguity, especially temporal contiguity, is not exploited in machine learning, except where some aspects are considered in reinforcement learning [74].

More than a century ago, James Mill (1773–1836) suggested three criteria for the strength of associations: permanence (resistance to forgetting), certainty (confidence), and spontaneity (reaction time). He was also concerned with how simple ideas get combined into complex ideas and proposed the notion of mental compounding. His son, John Stuart Mill (1806–1873), developed the theory of mental compounding into mental chemistry, a term for a complex idea that originally derived from constituent simple ideas but which was qualitatively different from the sum of the simple constituents. In the 1970s Tulving proposed the encoding specificity principle which is quite similar to John Stuart Mill's theory, in that the association of two ideas results in a unique combination that may render the constituents unrecognizable by themselves. More recently, Feldman suggests chemical scrambling or conceptual blending as a form of metaphor building in language processing [15]. In medicine, especially in neuropsychiatry, mind is often viewed as a chemical machine [1].

Ironically, the old idea of mental compounding or mental chemistry is not reflected in machine learning research to date. Connectionist models or neural networks are not suitable to model this kind of learning. Mental chemistry requires building blocks or modules that can be combined, which is not possible by weight adjustment alone. There has been some work on learning by symbolic composition in the 1980's but not many methods survived the 1990s [53], [54], [9]. The pure symbolic approach does not assure predictive accuracy from a training set. Some exceptions are inductive logic programming [54] and probabilistic relational models [20]. It should also be noted that the popular learning algorithms, such as support vector machines and Bayesian networks, are not appropriate to simulate the mental chemistry either. It will be a challenge to have a learning machine that shows high performance and can simulate cognitively plausible learning behavior, i.e., cognitive learning.

C. Three Principles of Cognitive Learning

Based on the reviews above, we propose the following three principles as the most fundamental to cognitive learning and thus guide towards human-level machine learning.

1. The Principle of Continuity

Learning in humans is a continuous, lifelong process. The consequence of acquiring new information is stored in memory. The experiences of each immediately past moment are memories that merge with current momentary experiences to create the impression of seamless continuity in our lives [52]. As noted before, Aristotle already identified this principle.

2. The Principle of Glocality

Cognition is developed situated in a multifaceted environment. Perception is dependent on context and it is important to maintain both global and local, i.e., glocal, representations [60]. The brain consists of specialized functional modules connected as a complex system in an intricate but integrated fashion [19], [11].

3. The Principle of Compositionality

The brain is highly versatile in its response. Versatility requires a means to compose various concepts from constituent concepts. The brain activates existing metaphorical structures to form a conceptual blend, consisting of all the metaphors linked together [15]. As already stated by James Mill, compositionality is an important property of human learning and memory.

Understanding the principles of natural cognitive learning is one thing; using them for constructing synthetic systems or "cognitive machine learning" is another. How do we develop learning architectures and algorithms based on these principles? In the following section, we present the hypernetwork model as a candidate architecture for cognitive learning and examine its properties.

III. Hypernetworks for Cognitive Learning

A. Hypernetwork Models of Memory

A hypernetwork is a weighted random hypergraph in which higher-order interactions of vertices are explicitly represented in hyperedges (Box 2). Each hyperedge is associated with a weight representing the strength of the association across the vertices forming the hyperedge. A hyperedge connecting k vertices is called a k-hyperedge. A hypernetwork is called a k-hypernetwork if all its hyperedges are k-hyperedges. As an associative memory model, the vertices represent primitive memory units (or concepts) and the hyperedges represent compound memory units (higher-order concepts). The weights of the hyperedges represent the associative strengths of the memory units.

Given a data set $D = \{x^{(n)}\}_{n=1}^{N}$ of N example patterns, the hypernetwork represents the probability of generating the data set D:

$$P(D|W) = \prod_{n=1}^{N} P\left(x^{(n)}|W\right), \tag{1}$$

where W denotes both the structure of the hyperedges and their weight values. It is important to note that hypernetwork memory makes use of a large number of random memory fragments $x_{i_1}^{(n)} x_{i_2}^{(n)} \ldots x_{i_k}^{(n)}$ of varying order k to estimate the probability. Thus, the probability of an example is expressed as

$$P\left(x^{(n)}|W\right) = \frac{1}{Z(W)} \exp\left[\sum_{k=1}^{K} \frac{1}{C(k)} \times \sum_{i_1, i_2, \ldots, i_k} w_{i_1 i_2 \ldots i_k}^{(k)} x_{i_1}^{(n)} x_{i_2}^{(n)} \ldots x_{i_k}^{(n)}\right], \tag{2}$$

where $Z(W)$ is the normalizing term and $C(k)$ is the number of possible hyperedges of order k. Note that the number of possible memory fragments grows exponentially.

Finding a compact, or minimal, ensemble of hyperedges is an important problem related to Occam's razor [61], [92], [49], [2]. Human cognitive memory obviously manages this problem. The next subsection describes a hypernetwork-based method that attempts to simulate human-like memory phenomena, such as recall memory and visual imagery.

B. Learning Hypernets by Molecular Evolution

We represent a hypernetwork as a collection of hyperedges. Each hyperedge is encoded as a molecule. For example, DNA sequences of four nucleotide types (A, T, G, and C) can be designed to encode the hyperedges [72], [41]. The weights of the hyperedges are encoded as the copy numbers of the molecules. The whole hypernetwork is represented as a library of DNA molecules. In this molecular representation, memories are learned by acquiring new molecules and adapting their compositions and copy numbers. Thus, learning involves evolving the molecular library, leading to a molecular evolutionary learning algorithm.

Learning starts with a library L of random molecules. It proceeds in an incremental way (Box 3). Given an example pattern, $x^{(n)} = (x_1^{(n)}, x_2^{(n)}, \ldots, x_I^{(n)})$ a large number of random hyperedges are sampled from this example, encoded as molecules, and mixed with the molecules in the library. Through molecular recognition, the matching molecules are selected and copied. This process increases the density of the molecular memory fragments proportional to the frequency of observations in the data set [89]. In a supervised learning regime where there is a target output associated with the input, the molecules that match with the input but mismatch with the target output are removed from the library. Essentially, we use three primitive operations of matching, selection, and amplification to adapt the composition of the molecular individuals (hyperedge structure) and their distribution (hyperedge weights) in the library (hypernetwork). Note this is an evolutionary process occurring at the molecular level.

Formally, the molecular evolutionary algorithm performs gradient search to find maximum-likelihood parameters for the training data set. To see this, we take the logarithm of the likelihood function of Eqn. (1)

$$\begin{aligned} \ln P(D|W) &= \ln \prod_{n=1}^{N} P\left(x^{(n)}|W\right) \\ &= \sum_{n=1}^{N} \left\{ \left[\sum_{k=1}^{K} \frac{1}{C(k)} \sum_{i_1, i_2, \ldots, i_k} \times w_{i_1 i_2 \ldots i_k}^{(k)} x_{i_1}^{(n)} x_{i_2}^{(n)} \ldots x_{i_k}^{(n)} \right] - \ln Z(W) \right\}, \end{aligned} \tag{3}$$

and its derivative to get (see Box 3)

$$\begin{aligned} &\frac{\nabla}{\nabla w_{i_1, i_2, \ldots, i_k}^{(k)}} \ln \prod_{n=1}^{N} P\left(x^{(n)}|W\right) \\ &= N \left\{ \left\langle x_{i_1} x_{i_2} \ldots x_{i_k} \right\rangle_{Data} - \left\langle x_{i_1} x_{i_2} \ldots x_{i_k} \right\rangle_{P(x|W)} \right\}, \end{aligned} \tag{4}$$

which is the difference between the average frequencies of the hyperedges in the data set and those in the hypernetwork model. Minimizing this difference, as performed by the molecular evolutionary procedure described above, is equivalent to maximizing the likelihood.

C. An Example: Learning Linguistic Memory

Human language and thought are strongly shaped by our experience with our physical and social environments. Language and thought are not best studied as formal mathematics and logic, but as adaptations that enable creatures like us to thrive in a wide range of situations [15]. To know how to use language correctly, we need to have integrated knowledge involving various aspects of language such as grammar (the patterns), lexicon (the words), semantics (identity of the subject), a cultural image, and associated knowledge. There must be precise linkages across all these modalities.

We evaluate the use of hypernetworks for learning colloquial language in a pseudo-embodied, integrated, multimodal environment. We collected a corpus of 290K dialogue sentences from TV movies and dramas such as Friends, House, 24, and Prison Break [95]. We want the hypernetwork to learn the general language patterns from the dialogues of everyday situations, and to show linguistic recall capability, such as making a complete sentence or generating a new sentence given a partial list of words.

The hypernetwork is organized and learned by repeatedly observing the sentences. On each observation of a sentence $x^{(n)} = (x_1, x_2, \ldots, x_I)$ a large collection of hyperedges $E_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_j}, \ldots, x_{i_k})$, $i_j \in \{1, 2, \ldots, I\}$, $j = 1, \ldots, k$ are randomly sampled from $x^{(n)}$. The random hyperedges represent word association patterns. Then, they are matched

Box 2: Hypernetwork Architecture

A hypernetwork is a random hypergraph structure augmented with weights to the edges. Hypergraphs generalize simple graphs by allowing for edges of higher cardinality [3], [34], [50]. The edges in a hypergraph are called hyperedges. Formally, we define a hypernetwork as a triple $H = (X, E, W)$, where $X = \{x_1, x_2, \ldots, x_I\}$, $E = \{E_1, E_2, \ldots, E_{|E|}\}$, and $W = \{w_1, w_2, \ldots, w_{|E|}\}$ are the sets of vertices, edges, and weights, respectively. The cardinality of the hyperedges $E_i = \{x_{i1}, x_{i2}, \ldots, x_{ik}\}$ (size) is $k \geq 1$, i.e., the hyperedges can connect more than two vertices while in ordinary graphs the edges connect up to two vertices, i.e., $k \leq 2$. A hyperedge of cardinality k will be also referred to as an order k hyperedge or k-hyperedge. A hypernetwork consisting of k-hyperedges only is called a k-hypernetwork.

Figure 2 shows an example hypernetwork. It consists of seven vertices $X = \{x_1, x_2, \ldots, x_7\}$ and five hyperedges $E = \{E_1, E_2, E_3, E_4, E_5\}$, each having a different order $k_i = |E_i|$, and weights $W = \{w_1, w_2, w_3, w_4, w_5\}$. A hypernetwork can be represented as an incidence matrix $[a_j^i]$ with $|E|$ rows that represent the hyperedges of H and $I + 1$ columns that represent the vertices of H. The elements of the matrix are $a_j^i = 1$ if $x_i \in E_j$ and $a_j^i = 0$ if $x_i \notin E_j$ for columns $i = 1, \ldots, I$ and rows $j = 1, \ldots, |E|$, and $a_j^0 = w_j$ for the 0-th column and rows $j = 1, \ldots, |E|$.

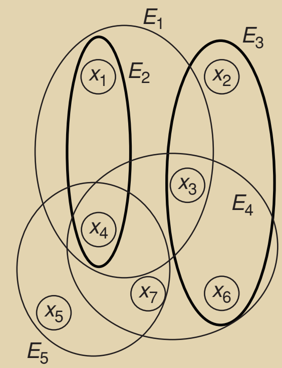
Representing a Hypernetwork

$H = (X, E, W)$
$X = \{x_1, x_2, \ldots, x_7\}$
$E = \{E_1, E_2, \ldots, E_5\}$
$W = \{w_1, w_2, \ldots, w_5\}$

$E_1 = \{x_1, x_3, x_4\}$, $w_1 = 2$
$E_2 = \{x_1, x_4\}$, $w_2 = 10$
$E_3 = \{x_2, x_3, x_6\}$, $w_3 = 7$
$E_4 = \{x_3, x_4, x_6, x_7\}$, $w_4 = 2$
$E_5 = \{x_4, x_5, x_7\}$, $w_5 = 4$

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7
E_1	2	1		1	1			
E_2	10	1			1			
E_3	7		1	1			1	
E_4	2			1	1		1	1
E_5	4				1	1		1

(a) Incidence Matrix



(b) Hypergraph

FIGURE 2 Hypernetwork architecture. A hypernetwork is a random hypergraph with weighted (hyper)edges. The cardinality k of the hyperedge, called the order, can range from 1 to $|E|$. In this example, the hypernetwork consists of seven vertices and five hyperedges of orders $k_i \in \{2, 3, 4\}$ and weights $w_i \in \{2, 4, 7, 10\}$. The hypernetwork can be represented as an incidence matrix (a) of vertices and edges augmented with weights (x_0) or graphically with ovals denoting the hyperedges encircling the vertices (b), where their thicknesses are proportional to the strengths of their weights.

The hypernetworks can be used as a probabilistic associative memory to store a data set $D = \{x^{(n)}\}_{n=1}^N$ so that they can be retrieved later by content, where $x^{(n)}$ denotes the n-th pattern to store. We define the energy of the hypernetwork as

$$\varepsilon\left(x^{(n)}; W\right) = -\sum_{i=1}^{|E|} w_{i_1 i_2 \ldots i_{|E_i|}} x_{i_1}^{(n)} x_{i_2}^{(n)} \ldots x_{i_{|E_i|}}^{(n)}, \tag{2.1}$$

where W represents the parameters (hyperedge weights) for the hypernetwork model. Note that $x_{i_1}^{(n)} x_{i_2}^{(n)} \ldots x_{i_{|E_i|}}^{(n)}$ is a combination of $k_i = |E_i|$ elements of the data item $x^{(n)}$, which is represented as a k_i-hyperedge in the network. Then, the probability of the data being generated from the hypernetwork is given as a Gibbs distribution

$$P\left(x^{(n)} | W\right) = \frac{1}{Z(W)} \exp\left\{-\varepsilon\left(x^{(n)}; W\right)\right\}, \tag{2.2}$$

where $\exp\{-\varepsilon(x^{(n)}; W)\}$ is called the Boltzmann factor and the normalizing term $Z(W)$ is expressed as

$$\begin{aligned} Z(W) &= \sum_{x^{(m)}} \exp\left\{-\varepsilon\left(x^{(m)}; W\right)\right\} \\ &= \sum_{x^{(m)}} \exp\left\{\sum_{i=1}^{|E|} w_{i_1 i_2 \ldots i_{|E_i|}} x_{i_1}^{(m)} x_{i_2}^{(m)} \ldots x_{i_{|E_i|}}^{(m)}\right\}. \end{aligned} \tag{2.3}$$

In effect, the hypernetwork represents a probabilistic model of the data set using a population of hyperedges and their weights.

against or "self-assembled" with the hyperedges E_j in the learned hypernetwork. Each matching hyperedge is then copied to increase its weight. When normalized, the distribution of the hyperedge weights represents the probability distribution of the word association patterns.

To test the recall memory, a query sentence is given with some words missing or corrupt. The hypernetwork should fill in the missing words. This can be done by retrieving all the hyperedges having the same context and estimating $P(x_i | x_{-i}, W)$ where x_i is the missing word

Box 3: Learning Hypernetworks

The hypernetwork is represented as a collection of hyperedges. Each hyperedge is encoded as a DNA molecule [72]. The whole hypernetwork is then represented as a library of DNA molecules (Figure 3) and learned by a molecular evolutionary algorithm, called molecular programming [90], that simulates the DNA computing process. Upon temperature control, two single-stranded DNA molecules of complementary sequences (nucleotides A-T and G-C) hybridize each other to form a single double-stranded DNA molecule [41], [48]. The evolutionary operators of matching, selection, and amplification are performed by DNA chemistry based on molecular recognition.

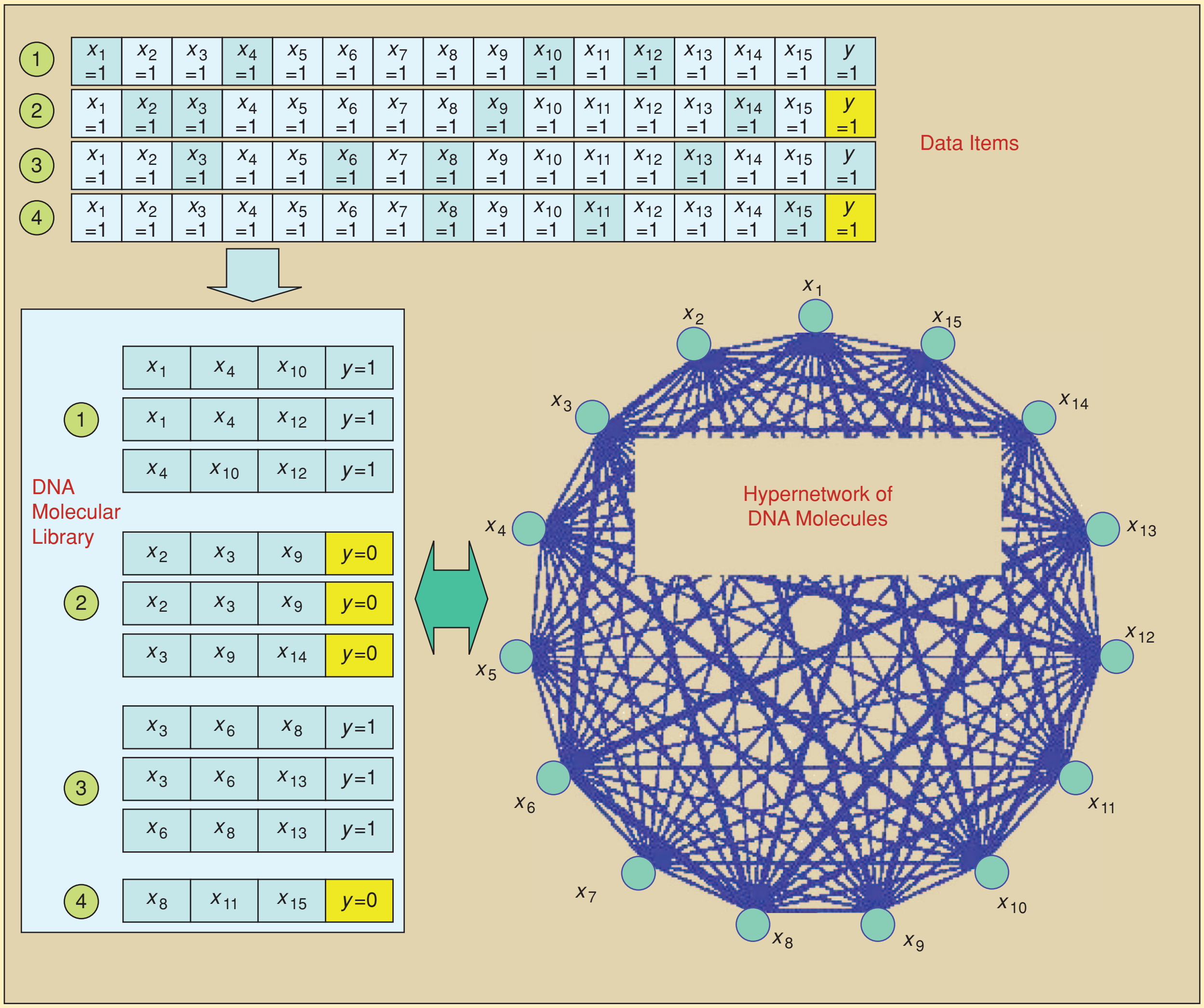


FIGURE 3 Learning a molecular hypernetwork from training data items. A hypernetwork is represented as a collection of hyperedges, encoded as a library of molecules where each molecule represents a hyperedge. Given a training data item, a collection of hyperedges are constructed and matched to the library representing the hypernetwork. The matching hyperedges are amplified and spuriously-matching hyperedges are removed (not shown in the figure). By repeating this process, the distribution of the molecular hyperedges is adapted to reproduce the distribution of the training data set.

Given a data set $D = \{x^{(n)}\}_{n=1}^{N}$, the goal of molecular evolution is to find a hypernetwork that maximizes the likelihood function:

$$P(D|W) = \prod_{n=1}^{N} P\left(x^{(n)}|W\right), \tag{3.1}$$

where W represents the hypernetwork parameters (structures and weights of the hyperedges). The algorithm starts with a random hypernetwork which is evolved incrementally on observing a new data item.

The procedure is illustrated in Figure 3. The procedure is summarized as follows. Here we describe the supervised learning regime where the training example $x^{(n)} = (x, y)$ consists of two parts, the input x and the target output y. In the unsupervised regime, the

algorithm proceeds similarly, except that the entire training pattern is considered as the input and, also, the target output.

1. Let L be the molecular library of hyperedges representing the current distribution $P(X, Y|W)$ of the hypernetwork.
2. Get the n-th training example $x^{(n)} = (x, y)$.
3. Classify x using L as follows:
   3.1 Extract all molecules matching x into M.
   3.2 From M separate the molecules into classes:
   Extract the molecules with label $y = 0$ into M^0
   Extract the molecules with label $y = 1$ into M^1
   3.3 Compute $y^* = \text{argmax}_{Y\in\{0,1\}} |M^Y|/|M|$
4. Update L
   4.1 If $y^* = y$, then $L_n \leftarrow L_{n-1} + \{\Delta c(u, v)\}$ for $u = x$ and $v = y$ for $(u, v) \in L_{n-1}$,
   4.2 If $y^* \neq y$, then $L_n \leftarrow L_{n-1} - \{\Delta c(u, v)\}$ for $u = x$ and $v \neq y$ for $(u, v) \in L_{n-1}$.
5. Go to step 2 if not terminated.

As a new data item is observed (Step 2), a large number of random hyperedges are sampled and encoded as molecules, and added to the library to compute the predicted output (Step 3). Then the library is updated (Step 4) by amplifying the perfectly matching molecules (Step 4.1) and by removing the partially mismatching molecules (Step 4.2). The algorithm makes use of the concentrations of molecules to represent their probabilities. How to control the number of copies, i.e., the learning rate, is described in [Zhang and Jang, DNA10-2005].

It can be shown that this evolutionary learning process performs "chemical" gradient search to find the maximum-likelihood parameters for the training data set. To see this, take the logarithm of the likelihood:

$$\begin{aligned}\ln P(D|W) &= \ln \prod_{n=1}^{N} P\left(x^{(n)}|W\right) \\ &= \sum_{n=1}^{N}\left\{\left[\sum_{k=1}^{K}\frac{1}{C(k)}\sum_{i_1,i_2,\ldots,i_k} w^{(k)}_{i_1 i_2\ldots i_k}\, x^{(n)}_{i_1}\, x^{(n)}_{i_2}\ldots x^{(n)}_{i_k}\right] - \ln Z\left(W\right)\right\},\end{aligned} \tag{3.2}$$

where Eqn. (2) is used for $P(x^{(n)}|W)$. We take the derivative of the log-likelihood

$$\begin{aligned}&\frac{\nabla}{\nabla w^{(k)}_{i_1,i_2,\ldots,i_k}} \ln \prod_{n=1}^{N} P\left(x^{(n)}|W\right) \\ &= \frac{\nabla}{\nabla w^{(k)}_{i_1,i_2,\ldots,i_k}} \sum_{n=1}^{N}\left\{\left[\sum_{k=1}^{K}\frac{1}{C(k)}\sum_{i_1,i_2,\ldots,i_k} w^{(k)}_{i_1 i_2\ldots i_k}\, x^{(n)}_{i_1} x^{(n)}_{i_2}\ldots x^{(n)}_{i_k}\right] - \ln Z(W)\right\} \\ &= \sum_{n=1}^{N}\left\{\frac{\nabla}{\nabla w^{(k)}_{i_1,i_2,\ldots,i_k}}\left[\sum_{k=1}^{K}\frac{1}{C(k)}\sum_{i_1,i_2,\ldots,i_k} w^{(k)}_{i_1 i_2\ldots i_k}\, x^{(n)}_{i_1}\, x^{(n)}_{i_2}\ldots x^{(n)}_{i_k}\right] - \frac{\nabla}{\nabla w^{(k)}_{i_1,i_2,\ldots,i_k}}\ln Z(W)\right\} \\ &= \sum_{n=1}^{N}\left\{x^{(n)}_{i_1}\, x^{(n)}_{i_2}\ldots x^{(n)}_{i_k} - \left\langle x_{i_1} x_{i_2}\ldots x_{i_k}\right\rangle_{P(x|W)}\right\} \\ &= N\left\{\left\langle x_{i_1}\, x_{i_2}\ldots x_{i_k}\right\rangle_{Data} - \left\langle x_{i_1} x_{i_2}\ldots x_{i_k}\right\rangle_{P(x|W)}\right\},\end{aligned} \tag{3.3}$$

where the two terms in the last line are defined as

$$\left\langle x_{i_1} x_{i_2}\ldots x_{i_k}\right\rangle_{Data} = \frac{1}{N}\sum_{n=1}^{N}\left[x^{(n)}_{i_1}\, x^{(n)}_{i_2}\ldots x^{(n)}_{i_k}\right] \tag{3.4}$$

$$\left\langle x_{i_1} x_{i_2}\ldots x_{i_k}\right\rangle_{P(x|W)} = \sum_{x}\left[x_{i_1}\, x_{i_2}\ldots x_{i_k}\, P(x|W)\right]. \tag{3.5}$$

In effect, the learning process maximizes the likelihood by minimizing the difference between the average frequencies of the hyperedges in the data set and in the hypernetwork model.

and x_{-i} are the rest of the words in the query sentence. For example, given the corrupt sentence query $x^{(q)}$ = ("who", ?, "you"), the missing word ? can be generated by retrieving all the hyperedges E_i = ("who", *, "you") and choosing the word in the position * that appears most frequently.

Table 2 shows the results for two different uses of the hypernetwork memory, i.e., recall (completion) and recognition (classification). In this particular experiment, the sentences were labeled with the title of the video, i.e., the source of the sentence. For example, a training sentence consists of ("who", "are", "you", Friends), where Friends denotes the sentence "Who are you" came from the drama Friends. In the recall task, the hypernetwork is given, say, (?, "are", "you") and should complete the missing word to produce "What are you". In the recognition task, the hypernetwork is to output "Friends" as the source of the sentence.

For the sample sentence "You need to wear it" which appeared in the movie 24 as well as in House, the learned hypernetwork could generate, for example, the sentences like "I need to wear it", "You want to wear it", and "You need to do it" with the right recognition of the sources.

Table 3 shows further results for the sentence completion task. For this experiment, we used order $k = 3$ hyperedges. It can be observed that the hypernetwork faithfully reconstructs the original sentences from the partial sentences with 50% of the words removed. It should be noted that we did not provide any explicit hints on syntax or semantics in the training set or during the learning process. The molecular evolutionary learning process discovered the memory fragments that build the language model, i.e., important patterns for assembling the sentences.

TABLE 2 Hypernetwork memory used for sentence completion and classification.

QUERY (PARTIAL INPUT)	COMPLETION (RECALL)	CLASSIFICATION (RECOGNITION)
WHO ARE YOU		
? ARE YOU	WHAT ARE YOU	FRIENDS
WHO ? YOU	WHO ARE YOU	FRIENDS
WHO ARE ?	WHO ARE YOU	FRIENDS
YOU NEED TO WEAR IT		
? NEED TO WEAR IT	I NEED TO WEAR IT	24
YOU ? TO WEAR IT	YOU WANT TO WEAR IT	24
YOU NEED ? WEAR IT	YOU NEED TO WEAR IT	24
YOU NEED TO ? IT	YOU NEED TO DO IT	HOUSE
YOU NEED TO WEAR ?	YOU NEED TO WEAR A	24

TABLE 3 Results for sentence completion. The hypernetwork was trained on 290K dialogue sentences from a video corpus of movies and dramas. The task is to recall the complete sentence from a partial list of words.

? ? A DREAM ABOUT ? IN COPENHAGEN
⇒ **I HAD A DREAM ABOUT YOU IN COPENHAGEN**
? ? DON'T NEED YOUR ?
⇒ **IF I DON'T NEED YOUR HELP**
? STILL ? BELIEVE ? DID THIS
⇒ **I STILL CAN'T BELIEVE YOU DID THIS**
? GONNA ? UPSTAIRS ? ? A SHOWER
⇒ **I'M GONNA GO UPSTAIRS AND TAKE A SHOWER**
? HAVE ? VISIT THE ? ROOM
⇒ **I HAVE TO VISIT THE LADIES' ROOM**
WE ? ? A LOT ? GIFTS
⇒ **WE DON'T HAVE A LOT OF GIFTS**
? APPRECIATE IT IF ? CALL HER BY ? ?
⇒ **I APPRECIATE IT IF YOU CALL HER BY THE *WAY***
? YOU ? FIRST ? OF MEDICAL SCHOOL
⇒ **ARE YOU GO FIRST DAY OF MEDICAL SCHOOL**
I'M STANDING ? THE ? ? ? CAFETERIA
⇒ **I'M STANDING IN *THE ONE* OF THE CAFETERIA**
? THINK ? I ? MET ? SOMEWHERE BEFORE
⇒ **I THINK BUT I AM MET HIM SOMEWHERE BEFORE**

D. Cognitive Learning with Hypernetworks

We are now in a better position to examine the properties of the hypernetwork model. We are interested to see whether or how much the principles for cognitive learning are reflected in the hypernetwork.

First, the continuity principle says that the learning process should be able to build a consistent memory based on a continuous stream of observed data. To meet this condition, it is important to incorporate the newly observed data while maintaining the old, persistent memory. In hypernetworks, the memory is maintained as a population of a large number of small random fragments (hyperedges). The old memory is updated by partially adding new hyperedges from a new observation or by removing those conflicting with the new example pattern. Since the whole population is "moving" as an ensemble, the hypernetwork can maintain long-term consistency while adapting to short-term change.

Second, the glocality principle asserts that learning should allow for both local and global representations and for changes in their balance. In hypernetworks, the hyperedges of large order k represent the specialized, thus local, information while those of small k represent the general information used more globally. By adopting a range of k in a single hypernetwork, we can support both local and global representations in a single model. It is interesting to see that the hypernetwork employs characteristics of both kernel machines and graphical models, which are respectively representatives of discriminative and generative learning models. On one hand, a large number of hyperedges expand the data space into a higher-dimensional hyperspace to make the decision boundaries easier to separate. This is similar in spirit to the kernel trick [69], [70], but the hypernetwork uses a large number of random kernels each of which is based on subdimensions of the data space. On the other hand, the hypernetwork builds a probability model of data as in graphical models [37], [4]. However, hypernetworks represent higher-order interactions explicitly, making it more efficient to discover the complex patterns in the data. In effect, the hypernetwork model makes a unified model of the two major architectures of today in a principled way. We mention that there are alternative methods to combine the discriminative and generative models, for instance [35], but these are completeley different from the hypernetwork approach.

Box 4: Multimodal Memory Game (MMG)

The game consists of a machine learner and two or more human learners in a digital cinema (Figure 4). All the participants including the machine watch the movies. After watching, the humans play the game by question-and-answering about the movie scenes and dialogues. In a simple version [83], there are two human players, called I2T and T2I. The task of player I2T (for image-to-text) is to generate a text given a movie cut (image). Player T2I (for text-to-image) is to generate an image given a text from the movie captions. While one player is asking, the other is answering. The two players alternate their roles. When the player is in the questioning mode, he receives all the multimodal inputs.

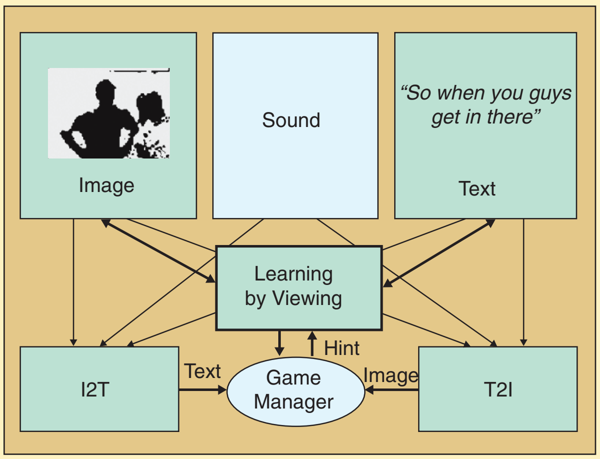


FIGURE 4 Multimodal memory game (MMG). There are two human learners (I2T and T2I) and one machine learner (learning by viewing) watching video movies in a digital cinema. The missions of the human learners are specialized to translate between modalities, e.g., I2T is to learn to recall text from image and T2I has to generate image from text. The goal of the machine learner is to perform crossmodal translation, e.g., generating a sentence given an image out of the movie or vice versa. The machine learner gets hints from the human learners playing the game by asking questions and answering them in different modalities.

When the player is in an answering mode, he receives the multimodal inputs except the modality in which he has to give an answer. The goal of the machine learner in this multimodal memory game is to imitate the human players by watching the movies, reading the captions, listening to the sounds, and observing the players enjoying the games over their shoulders.

The multimodal memory game is a real challenge for state-of-the-art machine learning technology. But the difficulty of the task can be controlled in a real-life environment. MMG also employs the aspects of the three fundamental principles for cognitive learning. First, the movies can be played as long as time allows (the continuity principle). In addition, the lifespan or difficulty of the continuity can be controlled by increasing or decreasing the movie length and scenarios. Second, the game involves the vision and language problems, the hardest core of human-level intelligence. Any solution to these problems will involve the representation problems, especially the global and local representations and their combinations (the glocality principle). Third, any machine learners solving the MMG will require a capacity to compose linguistic concepts and visual compounds (the compositionality principle).

The learned models can be used for recognition and recall memory tasks involved with multimedia data. Thus, for recognition, the model can be used for text retrieval by image or, also, for image retrieval by text. For cued recall tasks, the model can be used to generate natural language from images and to generate images from linguistic descriptions. The system can be used to study linguistic memory and visual memory.

The basic modes of the game can be modified to make the task harder or easier and to scale the problem size up or down. We may add new modalities, more players, and new media. We can also study gaming strategies. By modifying the parameters above, the platform can be tuned for long-term human-level performance studies as well as for short-term applications of industrial interest.

Third, the compositionality principle suggests that cognitive learning requires the possibility of building compound structures and generating versatile concepts. In hypernetworks, new concepts are built by cognitive chemistry, i.e., by randomly sampling and selectively amplifying the molecular hyperedges of varying order. Note that there is no explicit mutation or crossover, but there does exist variation in this evolutionary optimization process [16]. The variation comes from the random sampling of the hyperedges from the training examples. We found this is a very efficient way to introduce diversity while not wasting time creating infeasible hyperedges. We also found that structural learning by combinatorial compostions of hyperedges combined with the parametric learning by selective amplification, is a powerful source of high accuracy [40], [42], [24]. This emphasis on structural learning in hypernetwork models is distinguished from conventional machine learning architectures that focus on numerical parameter learning.

It is remarkable that both the structure and parameters of the hypernetwork architecture are modified by the same molecular evolutionary operators and that there is no need for repair. In this sense, the hypernetwork architecture is "evolution-friendly". This is a significant innovation over previous evolutionary approaches for machine learning, such as classifier systems [32], genetic programming [45], evolutionary neural trees [93], [94] where a tree-structured architecture is used so as to be crossover-friendly, and evolutionary neural networks where evolutionary algorithms are used for efficient optimization of standard architectures [82].

IV. Evolutionary Learning Experiments with Cognitive Hypernetworks

As discussed in Section II, it is important for an agent to be situated in an environment, and to get multimodal sensory inputs, if it is to evolve to human-level intelligence. The multimodal

memory game (MMG) is a research platform designed to study cognitive learning architectures in a multimedia environment [83].

The game is played in a digital cinema consisting of a machine learner and two or more human learners (Box 4). The machine and humans watch the movies through three modalities of vision, audio (speech), and text (caption). The goal of the machine learner is to learn crossmodal translation, i.e., given an input in one modality, it has to produce an answer in a different modality. The machine learner learns by viewing the movies and getting hints from the human learners who play the game by question-answering about the movie scenes and dialogues. Here we demonstrate a successful proof-of-concept by using hypernetworks to solve two crossmodal translation problems: image-to-text and text-to-image generation.

A. Language-Vision Crossmodal Translation

We experimented on the corpus of the TV dramas and movies described in Section III C. In this experiment, we collected approximately 3000 pairs of caption sentences and the corresponding scene images. The image data are preprocessed to generate the "visual words" or a visual vocabulary V_I. The text data are converted to a linguistic vocabulary V_T. The words in V_I and V_T can be considered as primitive concepts or features for building higher-level concepts or cognitive models of the multimodal sensory information. The vocabulary sizes for cognitive memory were $|V_I| = 80 \times 60 = 4800$ $|V_T| \cong 2500$ and The preprocessed images and text data are used to train the learner. The hypernetwork was used to learn the combined language-vision model by sampling the hyperedges consisting of features both from the image and the text parts of the input. Mathematically, the hypernetwork represents the joint probability of image I and text T:

$$P(I, T|W) = P(x_I, x_T|W) = P(x|W) \tag{5}$$

$$P(x|W) = \frac{1}{Z(W)} \exp\left[\sum_{k=1}^{K} \frac{1}{C(k)} \times \sum_{i_1, i_2, \ldots, i_k} w^{(k)}_{i_1 i_2 \ldots i_k} x_{i_1} x_{i_2} \ldots x_{i_k}\right], \tag{6}$$

where W are the parameters for the hypenetwork and $x = (x_I, x_T)$ is the training pattern consisting of image features x_I and text features x_T. The hypernetwork estimates the probability distribution by making use of a large number of conjunctive compositions of the visual and linguistic features. Smaller compositions of features represent general hyperfeatures while larger compositions represent more specialized hyperfeatures, conforming to the principle of glocality.

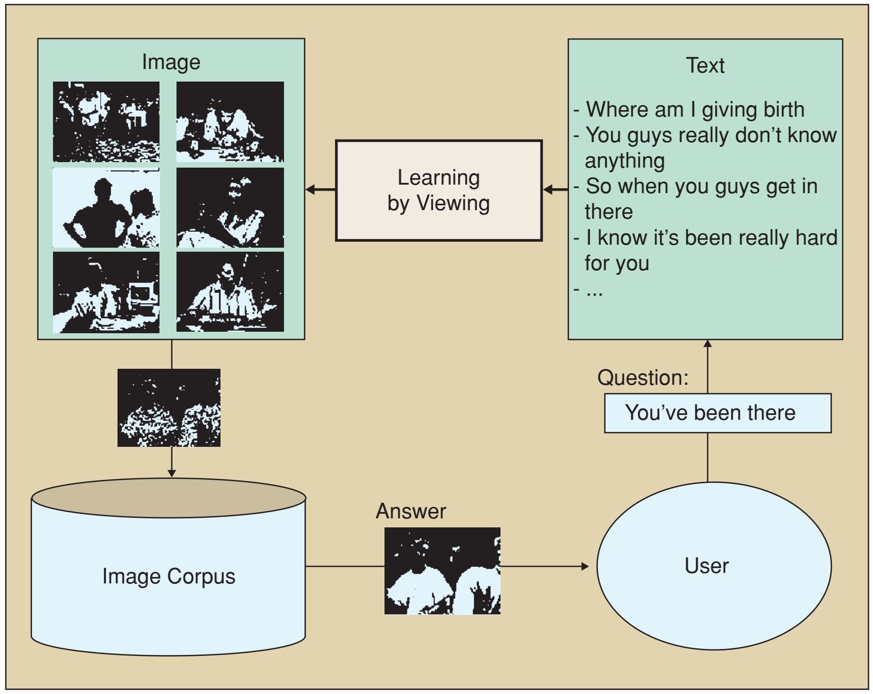


FIGURE 5 Text-to-image (T2I) crossmodal translation using the hypernetwork. A hypernetwork is learned to construct an image-text multimodal memory by viewing the video. Given a query in text, the hypernetwork memory is used to generate a pseudo-image from the text. The pseudo-image is then used as a visual query to retrieve from the video corpus the scene corresponding to the caption sentence given as the query.

Once learned the joint probability of image and text, the hypernetwork can be used to perform several tasks, including the generation of a text T given an image I and the generation of an image I given a text T. The image-to-text (I2T) problem can be solved by computing the conditional probability

$$P(T|I, W) = P(x_T|x_I, W) = \frac{P(x_T, x_I|W)}{P(x_I|W)}, \tag{7}$$

where $P(x_I|W)$ can be estimated by counting the number of hyperedges of the hypernetwork matching with the image I. Likewise, the text-to-image (T2I) generation problem can be solved by computing

$$P(I|T, W) = P(x_I|x_T, W) = \frac{P(x_I, x_T|W)}{P(x_T|W)}, \tag{8}$$

where $P(x_T|W)$ can be estimated by counting the number of hyperedges matching with the text T.

Figure 5 illustrates the procedure for the text-to-image (T2I) crossmodal translation using the hypernetwork. The hypernetwork is constructed to build an image-text multimodal memory by viewing the video. Given a text query, the

hypernetwork memory generates a pseudo-image from the text. The pseudo-image is then used as a visual query to retrieve the corresponding scene.

Figure 6 shows the results of text-to-image (T2I) experiments. The left column shows the query sentences and the middle column the images reconstructed by the hypernetwork. The right column shows the images retrieved using the reconstructed image as a visual query to the image database. It can be observed that the reconstructed images are blurred but very similar to the original video cuts. The hypernetwork achieved 97% of correct retrieval for the observed images.

Figure 7 shows the results of the image-to-text (I2T) experiments. The three columns from left to right show the query images, the sentence fragments recalled by the hypernetwork, and the sentences reconstructed by the fragment sentences. 98% of the original sentences were correctly reproduced.

Though on a small scale, these experiments are non-trivial. In fact, no major machine learning algorithms, such as support vector machines or Bayesian networks, can solve these problems directly.

B. Lifelong Learning in Noisy Environments

The random hypernetwork architecture was originally inspired by the complex, heterogeneous organization of molecular networks in biological systems. It would be interesting to simulate the process by which an unorganized memory system organizes itself to an organized system to perform a task persistently while adapting to be robust against temporary perturbations from the environment, for example, by noisy or incorrect training examples. We investigate this behavior on a suite of data sets as a surrogate for the noisy environment. These include 3760 digit images (of 64 bits each), 165 face images (480 bits each), and 120 gene expression samples (12,600 bits each) [84], [97]. The characteristics of these data sets is that they represent highly noisy and corrupted

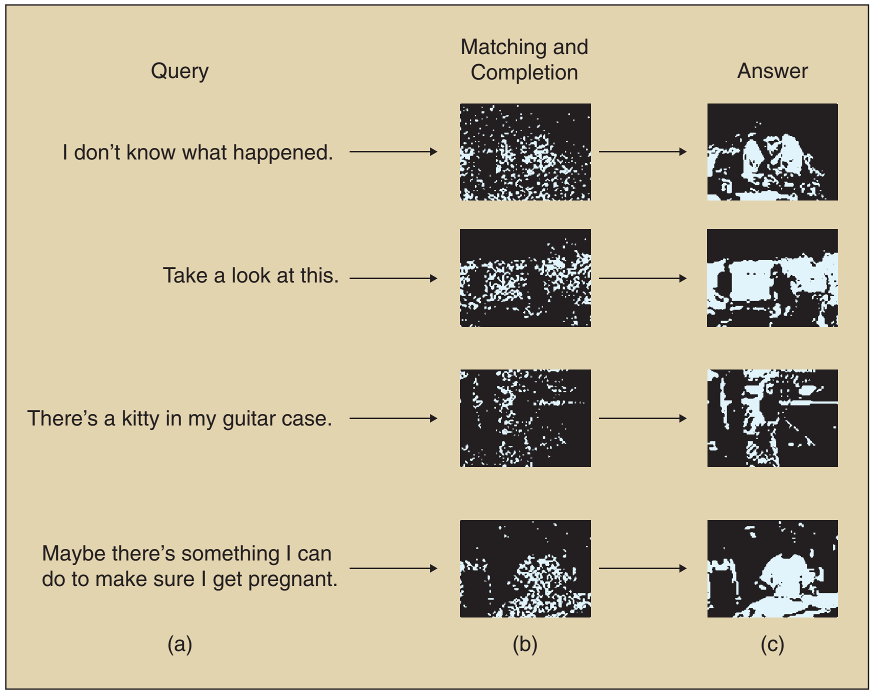


FIGURE 6 Text-to-image (T2I) generation using the hypernetwork trained on the image-text data from the video corpus of TV movies and dramas. The hyperedges of the learned hypernetwork encode the associations between the words of the text and the visual words of the image in the training set. From the query sentence (a), words are sampled randomly to build a large number of hyperedges that are matched with the text part of the hyperedges of the hypernetwork (not shown). From the matching hyperedges, the image fragments are selected and assembled to generate pseudo-images (b). The most probable pseudo-image is used to as a visual query to retrieve the original image in the video database (c).

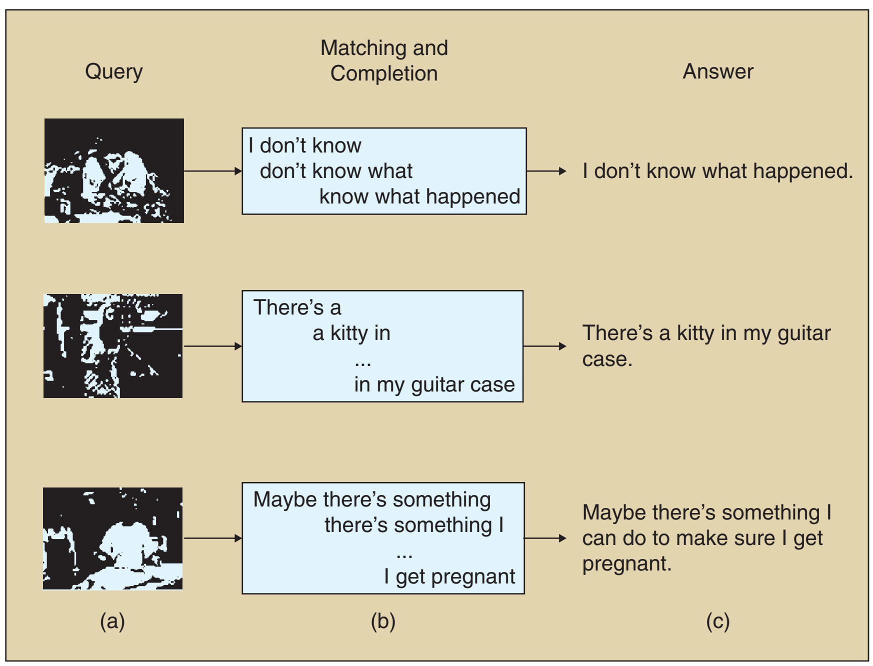


FIGURE 7 Image-to-text (I2T) generation using the hypernetwork trained on the image-text data from the video corpus of TV movies and dramas. The hyperedges of the learned hypernetwork encode the associations between the words of the text and the visual words of the image. From the query image (a), a large number of visual words are sampled to build hyperedges that are matched with the image part of the hyperedges of the hypernetwork (not shown). (b) From the matching hyperedges, the sentence fragments are generated and assembled to complete sentences. (c) The most probable sentence is chosen as the answer to the query image.

environments. Based on the data we build hypernetworks, and the structural and functional properties of the networks are examined to see what factors are crucial. In particular, we analyze the relationship between the diversity of the hyperedges and the performance of hypernetworks for a wide range of tasks.

The use of a large number of relatively simple yet heterogeneous hyperedges may offer insights into the role of diversity and flexibility of representation in adaptability of the whole system. This can test, for example, the hypothesis that the organizational diversity and flexibility of cognitive memory is a source of its resourcefulness that has an evolutionary advantage in survival and prosperity, as an analogy to biological networks [68], [51].

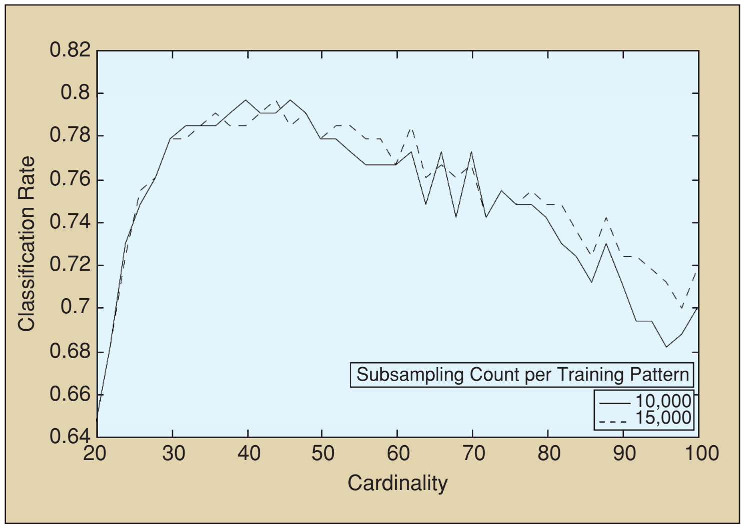


FIGURE 8 Hypergram for the face data set. The curve shows the profile of classification accuracy over the hyperedge size k (cardinality or order) ofuniform k-hypernetworks (not trained). For each face image consisting of 480 pixels, 10, 000 (solid curve) and 15,000 (dotted curve) hyperedges of varying order k were randomly sampled, respectively. Best performances were obtained for k values of 30 to 50. This result suggests that a hypernetwork with a large number of random samples of small k hyperedges (image fragments), e.g., $k = 30 \ll 480$, can make a good pattern recognizer on this data set.

Figure 8 shows the hypergram, i.e., the performance profile for the k-hypernetworks of various k values, on the face data. The best performance is obtained for hyperedges of order $k = 30$ to 50. Note that $k = 30$ is a 1/16 fraction of the total number of 480 variables in the image. The entire search space is of size 2^{480}. For k-hypernetworks, the effective ratio of search for $k = 30$ is $r = |E(H^{(k)})|/|E| = 1{,}500{,}000/2^{480} \ll 1$, where the number 1,500,000 comes from 10,000 hyperedges $\times$ 150 images (of 15 persons). The hypernetwork achieved very competitive performance compared to the state-of-the-art machine learning algorithms, including multilayer perceptrons, decision trees, support vector machines, and Bayesian networks. This and other comparative analyses [40], [42], [24] support the view that a combination of a large number of small fragments can make a good pattern classifier. This implies that small k-hypernetworks are effective to solve this problem if we use a large number of hyperedges. We also note that the general shape of the hypergrams for the digit data and the bio data are similar to those for the face data, except that the detailed regions and slope of low k values slightly differ [84].

The effect of noise in hyperedge sampling and matching was investigated. Biological cells seem robust against environmental noise and perturbations [51], [71]. Human cognitive memory also has similar robustness. We test this effect in our surrogate data by randomly corrupting the digit images.

Figure 9 shows the classification error (y-axis) vs. the corruption intensity (x-axis) for various k values (large k meaning large memory fragments). As expected, the error rate increases as the corruption intensity increases. The effect of corruption is more dramatic for large k-hypernetworks than for small k-hypernetworks. This implies that high k-hypernetworks are more susceptible to data corruption. Conversely, the low k-hyperedges are likely to be more robust against corruption and more useful for building a reliable system in a dynamic environment.

V. Outlook: Toward Human-Level Machine Learning

We introduced the hypernetwork model of cognitive memory together with a molecular evolutionary algorithm for learning the hypernetworks. We empirically evaluated the potential of using this architecture for cognitive modeling of human learning in language and vision domains. The hypernetwork model turns out to be faithful to the principles of continuity (forming lifetime memory continously), glocality (organizing a plastic structure of localized micromodules connected globally), and compositionality (generating compound structures from primitive ones), which are fundamental to achieving human-level machine learning and intelligence. In comparison to other paradigms for computational intelligence, the hyperinteractionistic molecular mind approach is unique in that the hypernetwork structure contains both the localist and globalist representations in a single architecture and continually adapts to environmental change through cognitive chemistry of matching, selection, and amplification based on molecular self-assembly. The self-assembling fluidic hyperinteraction network of micromodular structures facilitates to balance short-term adaptation and long-term continuity in a lifelong learning situation. We also find that the hypernetwork architecture is a unified model of learning that combines the major advantages of kernel machines and probabilistic graphical models.

Though we demonstrate the potential of the hypernetwork architecture for cognitive learning and memory, much work remains to achieve human-level learning and, eventually, to arrive at the goal of human-level intelligence. Future work can be pursued in three directions.

One involves extending the learning architecture. Though we find the current "flat" hypernetwork structure plausible on

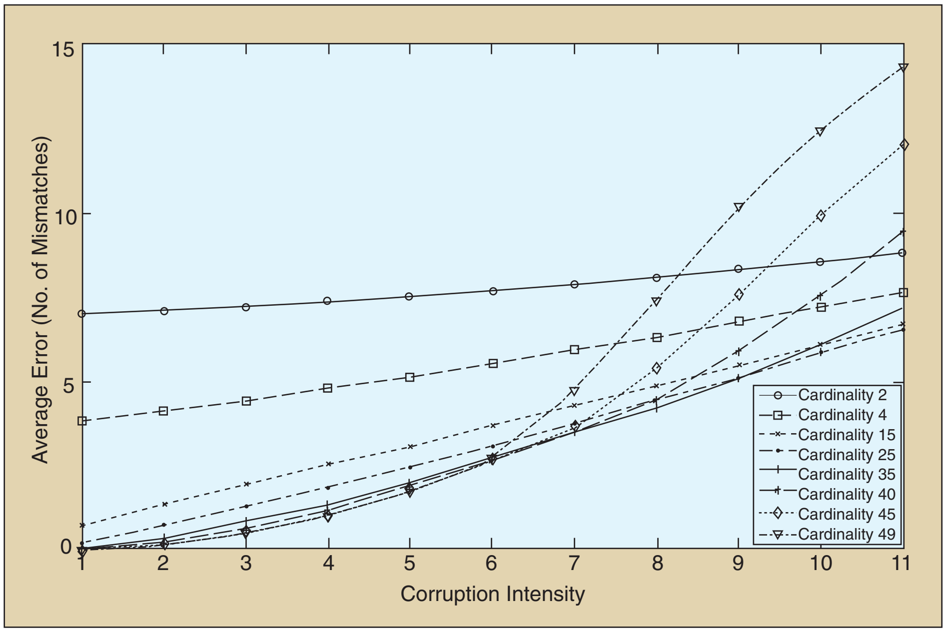


FIGURE 9 Tolerance against data corruption on the digit data set. The curves show the average error as a function of corruption intensity for k-hypernetworks of various order k, i.e., of hyperedges of cardinality k. Small k-hypernetworks, e.g., $k = 2, 4$, are more robust against the corruption. Large k-hypernetworks, e.g., $k = 40, 45$, show some critical points above which the error increases rapidly (e.g., corruption intensity of 7 for $k = 45$). The overall results suggest that small memory fragments (small k-hyperedges) are useful for maintaining long-term stability of the learning system.

a certain cognitive scale, and very competitive in solving many machine learning problems of practical interest, the size of computer memory for implementing larger-scale hypernetwork models increase rapidly since the number of hyperedges grows exponentially in the number of variables and in the order k. One solution to this problem is to build "deep" hypernetworks by introducing additional layers of hypergraphs, like a hypernetwork of hypernetworks. The human memory also has such a hierarchical organization on the whole-brain scale [73], [75].

A second direction involves learning strategies. As our review points out learning in humans is a continuous, lifelong process in an open, social environment. Machines that learn in this situated environment require selective attention and meta-level learning, otherwise there is too much noise, as well as too many signals to learn from. The notions of exploratory learning, such as query learning [79], selective learning and self-teaching [87], [96], and meta-learning [76], combined with structural learning, such as self-development [88], would be a necessity, rather than an option, to deal with the complexity of the real-world learning problem.

Another direction of future research includes further evaluation of the hyperinteractionistic molecular mind paradigm to develop theoretical and technological tools for constructing and testing large-scale computational theories of natural learning and intelligence. For cognitive modeling, the molecules may represent mental codes or concepts. For neuronal modeling, the molecules may represent neural codes or structures. The molecular assemblies and their interactions in the hypernetwork architecture can be used to explore the organizational principles of, for example, cell assemblies and their interactions.

Acknowledgments

This work was supported by the Ministries of MOST (NRL Biointelligence Project, 2002–2007), MICE (Molecular Evolutionary Computing Project, 2000–2009), and MEHR (BK-IT Program, 2006–2009). The author thanks Jung-Woo Ha, Ha-Young Jang, Joo-Kyoung Kim, Sun Kim, and Chan-Hoon Park for simulations and In-Hee Lee, Ji-Hoon Lee, and Kyung-Ae Yang for DNA computing experiments. Thanks are also due to Bob McKay and Xuan Hoai Nguyen for insightful discussions.

References

[1] S.H. Barondes, *Molecules and Mental Illness*. Scientific American Library, 1993.

[2] E. Baum, *What Is Thought?* MIT Press, 2004.

[3] C. Berge, *Graphs and Hypergraphs*. North-Holland Publishing, 1973.

[4] C.M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[5] C.S. Calude, J. Casti, and M.J. Dinneen (Eds.), *Unconventional Models of Computation*. Springer-Verlag, 1998.

[6] N.S. Cassimatis, "A cognitive substrate for achieving human-level intelligence," *AI Magazine*, pp. 45–56, Summer 2006.

[7] R.G. Crowder, *Principles of Learning and Memory*. Lawrence Erlbaum, 1976.

[8] W. Duch, R.J. Oentaryo, and M. Pasquier, "Cognitive architectures: Where do we go from here?," *Proc. First Conf. on Artificial General Intelligence*, University of Memphis, 2008.

[9] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Wiley, 2000.

[10] J. Eccles and D.N. Robinson, *The Wonder of Being Human: Our Brain and Our Mind*. Free Press, 1984.

[11] G.M. Edelman, *Wider Than the Sky*. Yale University Press, 2004.

[12] H. Eichenbaum, *The Cognitive Neuroscience of Memory*. Oxford University Press, 2002.

[13] M. Eigen and R. Winkler, *Laws of the Game: How the Principles of Nature Govern Chance*. Princeton University Press, 1993.

[14] J. Feldman and D. Ballard, "Connectionist models and their properties," *Cognitive Science*, vol. 6, no. 3, pp. 205–254, 1982.

[15] J. Feldman, *From Molecule to Metaphor: A Neural Theory of Language*. MIT Press, 2006.

[16] D. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press, 1995.

[17] K.D. Forbus and T.R. Hinrichs, "Companion cognitive systems: A step toward human-level AI," *AI Magazine*, pp. 83–95, Summer 2006.

[18] W.J. Freeman, *Neurodynamics: An Exploration in Mesoscopic Brain Dynamics*. Springer-Verlag, 2000.

[19] J. Fuster, *Cortex and Mind: Unifying Cognition*. Oxford University Press, 2003.

[20] L. Getoor, N. Friedman, D. Koller, A. Pfeffer, and B. Taskar, "Probabilistic relational models," In L. Getoor and B. Taskar (Eds.), *Introduction to Statistical Relational Learning*, pp. 129–174, 2007.

[21] R.W. Gibbs Jr. , *Embodiment and Cognitive Science*. Cambridge University Press, 2005.

[22] B. Goertzel and C. Pennachin (Eds.), *Artificial General Intelligence*. Springer-Verlag, 2005.

[23] S. Grillner and A.M. Graybiel (Eds.), *Microcircuits: The Interface between Neurons and Global Brain Function*. MIT Press, 2006.

[24] J.-W. Ha, J.-H. Eom, S.-C. Kim, and B.-T. Zhang, "Evolutionary hypernetwork models for aptamer-based cardiovascular disease diagnosis," *The Genetic and Evolutionary Computation Conference (GECCO 2007)*, pp. 2709–2716, 2007.

[25] J. Hawkins and S. Blakeslee, *On Intelligence*. Times Books, 2004.

[26] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1994.

[27] D.O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. Wiley, 1949.

[28] R. Hecht-Nielsen, "Cogent confabulation," *Neural Networks*, vol. 18, no. 2, pp. 111–115, 2005.

[29] G.E. Hinton and J.A. Anderson (Eds.), *Parallel Models of Associative Memory*. Erlbaum, 1989.

[30] W. Hirstein, *Brain Fiction: Self-Deception and the Riddle of Confabulation*. MIT Press, 2004.

[31] J. Holland, *Emergence: From Chaos to Order*. Perseus Books, 1998.

[32] J. Holland, "Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems," in Michalski, R., Carbonell, J., & Mitchell, T (Eds.), *Machine Learning: An Artificial Intelligence Approach*, vol. 2, Morgan Kaufmann, 1986.

[33] J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci.*, USA, vol. 79, no. 8, 1982, pp. 2554–2558, 1982.

[34] S. Janson, T. Luczak, and A. Rucinski, *Random Graphs*. Wiley, 2000.

[35] T. Jebara, *Machine Learning: Discriminative and Generative*. Kluwer, 2004.

[36] G.F. Jones, "Integrated intelligent knowledge management," In *Achieving Human-Level Intelligence through Integrated Systems and Research: Papers from the AAAI 2004 Fall Symposium*, 2004.

[37] M.I. Jordan (Ed.), *Learning in Graphical Models*. MIT Press, 1998.

[38] E. Kandel, *In Search of Memory: The Emergence of a New Science of Mind*. Norton, 2006.

[39] S. Kauffman, *Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, 1993.

[40] J.-K. Kim and B.-T. Zhang, "Evolving hypernetworks for pattern classification," *IEEE Congress on Evolutionary Computation (CEC 2007)*, pp. 1856–1862, 2007.

[41] J.-S. Kim, J.-W. Lee, Y.-K. Noh, J.-Y. Park, D.-Y. Lee, K.-A. Yang, Y.G. Chai, J.-C. Kim, and B.-T. Zhang, "An evolutionary Monte Carlo algorithm for predicting DNA hybridization," *BioSystems*, vol. 91, no. 1, pp. 69–75, 2008.

[42] S. Kim, S.-J. Kim, and B.-T. Zhang, "Evolving hypernetwork classifiers for microRNA expression profile analysis," *IEEE Congress on Evolutionary Computation (CEC 2007)*, pp. 313–319, 2007.

[43] C. Koch and J.L. Davis (Eds.), *Large-Scale Neuronal Theories of the Brain*. MIT Press, 1994.

[44] T. Kohonen, *Content-Addressable Memories*. Springer-Verlag, Berlin, 1980.

[45] J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[46] P. Langley, "Cognitive architectures and general intelligent systems," *AI Magazine*, pp. 33–44, Summer 2006.

[47] I.-H. Lee, K.-A. Yang, J.-H. Lee, J.-Y. Park, Y.-K. Chae, J.-H. Lee, B.-T. Zhang, "The use of gold nanoparticle aggregation for DNA computation and logic-based biomolecular detection," *Nanotechnology*, 2008 (submitted).

[48] H.-W. Lim, J.-E. Yun, H.-M. Jang, Y.-G. Chai, S.-I. Yoo, and B.-T. Zhang, "Version space learning with DNA molecules," *Proc. 2002 Int. Annual Meeting on DNA Comp. (DNA8)*, LNCS 2568: 2003, pp. 143–155, 2003.

[49] D.J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[50] D.J. Marchette, *Random Graphs for Statistical Pattern Recognition*. Wiley, 2004.

[51] S. Maslov and K. Sneppen, "Specificity and stability in topology of protein networks," *Science*, vol. 296, pp. 910–913, 2002.

[52] J.L. McGaugh, *Memory & Emotion: The Making of Lasting Memories*. Columbia University Press, 2003.

[53] R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach*. Tioga Publishing, 1983.

[54] T. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[55] M. Minsky, *The Emotion Machine: Commonsense Thinking, Artificial Intelligence, and the Future of the Human Mind*. Simon & Schuster, 2006.

[56] N.J. Nilsson, "Human-level artificial intelligence? Be serious!," *AI Magazine*, pp. 68–75, 2005.

[57] D. Norman, *Things That Make Us Smart: Defending Human Attributes in the Age of the Machine*. Addison-Wesley, 1993.

[58] R.C. O'Reilly, "Biologically based computational models of high-level cognition," *Science*, vol. 314, pp. 91–94, October 2006.

[59] G. Paun (Ed.), *Computing with Bio-Molecules: Theory and Experiments*. Springer-Verlag, 1998.

[60] M.A. Peterson and G. Rhodes (Eds.), *Perception of Faces, Objects, and Scenes: Analytic and Holistic Processes*. Oxford University Press, 2003.

[61] T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature*, vol. 317, pp. 314–319, 1985.

[62] R.F. Port and T. Van Gelder, (Eds.), *Mind as Motion: Explorations in the Dynamics of Cognition*. MIT Press, 1995.

[63] F. Pulvermüller, *The Neuroscience of Language: On Brain Circuits of Words and Serial Order*. Cambridge University Press, 2003.

[64] V.S. Ramachandran, *The Emerging Mind*. BBC/Profile Books, 2003.

[65] D.E. Rumelhart and D.A. Norman, "Accretion, tuning and restructuring: Three modes of learning," In J. W. Cotton and R. Klatzky (Eds.), *Semantic Factors in Cognition*, Lawrence Erlbaum, 1976.

[66] D.E. Rumelhart, J. MaClelland (Eds.), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*. 2 vols., MIT Press, 1986.

[67] M. Scheutz (Ed.), *Computationalism: New Directions*. MIT Press, 2002.

[68] F.O. Schmitt, D.M. Schneider, and D.M. Crothers (Eds.), *Functional Linkage in Biomolecular Systems*. Raven Press, 1975.

[69] B. Schoelkopf, C. Burges, A. Smola (Eds.), *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

[70] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[71] A.M. Sengupta, M. Djordjevic, and B.I. Shraiman, "Specificity and robustness in transcription control networks," *Proc. Natl. Acad. Sci.* USA, vol. 99, no. 4, 2002, pp. 2072–2077, 2002.

[72] S.-Y. Shin, I.-H. Lee, D. Kim, and B.-T. Zhang, "Multi-objective evolutionary optimization of DNA sequences for reliable DNA computing," *IEEE Trans. Evol. Comp.*, vol. 9, no. 2, pp. 143–158, 2005.

[73] L.R. Squire and E.R. Kandel, *Memory: From Mind to Molecules*. Scientific American Library, 1999.

[74] R.S. Sutton and A.G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[75] J.D. Sweatt, *Mechanisms of Memory*. Elsevier, 2003.

[76] T. Thrun and L.Y. Pratt (Eds.), *Learning To Learn*. Kluwer, 1998.

[77] K.J. Thurber and L.D. Wald, "Associative and parallel processors," *ACM Computing Surveys*, vol. 7, no. 4, pp. 215–225, 1975.

[78] D. Tse et al., "Schemas and memory consolidation," *Science*, vol. 316, pp. 76–82, Apr. 2007.

[79] L.G. Valiant, "A theory of the learnable," *Communications of the ACM*, vol. 27, no. 11, pp. 1134–1142, 1984.

[80] L.M. Ward, *Dynamical Cognitive Science*. MIT Press, 2002.

[81] B. Widrow, "Cognitive memory and its applications," *Invited talk at The IEEE World Congress on Computational Intelligence (WCCI-2006)*, Vancouver, 2006.

[82] X. Yao and M.M. Islam, "Evolving artificial neural network ensembles," *IEEE Computational Intelligence Magazine*, vol. 3, no. 1, pp. 31–42, February 2008.

[83] B.-T. Zhang, "Cognitive learning and the multimodal memory game: Toward human-level machine learning," *IEEE World Congress on Computational Intelligence: Special Session on Cognitive Architectures for Human-Level Intelligence (WCCI-2008)*, 2008 (in press).

[84] B.-T. Zhang, "Random hypergraph models of learning and memory in biomolecular networks: Shorter-term adaptability vs. longer-term persistency," *The First IEEE Symposium on Foundations of Computational Intelligence (FOCI'07)*, pp. 344–349, 2007.

[85] B.-T. Zhang, "Hyperinteractionism: Computational investigations into artificial minds beyond symbolism, connectionism, and dynamicism," *Natural Science*, pp. 144–165, Summer 2005.

[86] B.-T. Zhang, "Molecular nanobiointelligence computers: Computer science meets biotechnology, nanotechnology, and cognitive brain science," *Communications Kor. Info. Sci. Soc.*, vol. 23, no. 5, pp. 41–56, 2005.

[87] B.-T. Zhang, "Accelerated learning by active example selection," *International Journal of Neural Systems*, vol. 5, no. 1, pp. 67–75, 1994.

[88] B.-T. Zhang, "Self-development learning: Constructing optimal size neural networks via incremental data selection," *Arbeitspapiere der GMD*, no. 768, German National Research Center for Computer Science, St. Augustin/Bonn, 1993.

[89] B.-T. Zhang and H.-Y. Jang, "A Bayesian algorithm for in vitro molecular evolution of pattern classifiers," *Proc. 2004 Int. Annual Meeting on DNA Computing (DNA10)*, LNCS 3384: pp. 458–467, 2005.

[90] B.-T. Zhang and H.-Y. Jang, "Molecular programming: Evolving genetic programs in a test tube," *The Genetic and Evolutionary Computation Conf. (GECCO 2005)*, pp. 1761–1768, 2005.

[91] B.-T. Zhang and J.-K. Kim, "DNA hypernetworks for information storage and retrieval," *Proc. 2006 Int. Annual Meeting on DNA Computing (DNA12)*, LNCS 4287: pp. 298–307, 2006.

[92] B.-T. Zhang and H. Mühlenbein, "Balancing accuracy and parsimony in genetic programming," *Evolutionary Computation*, vol. 3, no. 1, pp. 17–38, 1995.

[93] B.-T. Zhang and H. Mühlenbein, "Evolving optimal neural networks using genetic algorithms with Occam's razor," *Complex Systems*, vol. 7, no. 3, pp. 199–220, 1993.

[94] B.-T. Zhang, P. Ohm, and H. Mühlenbein, "Evolutionary induction of sparse neural trees," *Evolutionary Computation*, vol. 5, no. 2, pp. 213–236, 1997.

[95] B.-T. Zhang and C.-H. Park, "Self-assembling hypernetworks for cognitive learning of linguistic memory," *Proc. World Acad. Sci. Eng. Tech. (WASET)*, pp. 134–138, 2008.

[96] B.-T. Zhang and G. Veenker, "Neural networks that teach themselves through genetic discovery of novel examples," *Proceedings of the 1991 IEEE Int. Joint Conf. Neural Networks (IJCNN'91)*, pp. 690–695, 1991.

[97] B.-T. Zhang, J. Yang, and S.-W. Chi, "Self-organizing latent lattice models for temporal gene expression profiling," *Machine Learning*, vol. 52, no. 1/2, pp. 67–89, 2003.

[98] H.Z. Zimmer, A. Mecklinger, and U. Lindenberger, "Levels of binding: Types, mechanisms, and functions of binding in memory," In Zimmer, H. Z., Mecklinger, A., and Lindenberger, U. (Eds.), *The Handbook of Binding and Memory*, Oxford University Press, 2006.