Method and apparatus for artificial general intelligence

By Correy Kowall and Bion Howard
Of the Universal Molecular Corporation
16 August 2016

CROSS-REFERENCE TO RELATED APPLICATIONS

Patent # 9,418,343: Multistage learner for efficiently boosting large datasets
Patent # 9,418,241: Unified platform for big data processing
Patent # 9,418,334: Hybrid pre-training of deep belief networks

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

No federal funding was used to develop this patent.

BACKGROUND OF THE INVENTION

0001

Many areas of human endeavor suffer from a combinatorial explosion of dimensions or information overload in general. This often results from interactions between multiscale systems, systems made of subsystems composed of nested subsystems in turn. As systems are understood in ever finer detail, the prediction of interactions between said systems becomes intractable. Many methods for processing information use layered probabilistic networks to extract statistical features from data or predict the consequences of behaviors and behavior patterns. 'Deep' systems are not necessarily the ideal configuration for machine learning in applications where the number of available options is large. The most capable neural network system known is the neocortex of *Homo sapiens sapiens,* a network only six layers deep yet millions of network modules wide. Systems which explore multiscale, option rich environments by 'brute force' become bogged down because it is impossible to explore infinite spaces in finite time. Often, learning systems store learned information in a tangled nest of network edges, making it difficult or impossible to troubleshoot or label deep learning systems. Current state-of-the-art machine learning lacks methods and systems for efficient, distributed, modular unsupervised learning and exploration in complex multiscale domains. Novel methods and systems are desirable.

BRIEF SUMMARY OF THE INVENTION

0002

In a preferred embodiment of the disclosed material, a computer-implemented method may use a policy with a number of elements to create a system termed an agency, comprised of a variable number of network learning agents. This agency is used to explore a conceptual and/or real-world space. The agency uses several innovations in combination to explore said space more efficiently. First, the agents within the agency use the calculus of information to decompose the environment and range of possible actions into modules, which may be hierarchical, and such agents may allocate more attention to modules which offer the most utility for effective learning and performance on desired goals. Further, the agency modifies the policy, describing elements including but not limited to number, network topology, resource usage, inputs, receptive fields, outputs, and parameterization of agents in a dynamic manner to maximize performance, while learning systems known to the art only modify a minor number of these properties if any. Next, agents within the agency communicate information on their performance to other agents and to the agency to increase the overall performance of the agency within the environment. Further, the agency may use the results of the modular extraction and learning process to generate one or more searchable indices of the domain(s) of interest. The combination of trained agency, searchable index and/or dynamic ontology serve as a compressed form of the domain information, enabling rapid responses to queries.

0003

In one implementation of the disclosed material, a computer-implemented method may begin with a query to a user for one or more domain(s) to be explored. Given this input, an agency may be created with the target domain(s) and a default policy. This agency may then retrieve a sampled set of information from the target domain. This agency may then instantiate a number of neural network agents with properties defined by the default policy and connections to the environment of choice. Agents may be instantiated with properties which differ from the default policy in a probabilistic manner to increase the variety of the agents. Agents may then retrieve data from their respective receptive fields within the domain(s). Agents may then use the calculus of information to decompose this receptive field space into modules and sequences of modules defined by prediction error boundaries and therefore the domain of free energy. The agents may then directly command modular actions or action sequences, or the agency may receive and aggregate actions and action sequences from agents into a different number of modules and sequences. The agency may then estimate the utility of each module or sequence, and the agency may directly command external systems to perform these actions or action sequences in series and/or parallel based on their utility rank, and the agency may then update the policy based on the environmental response to these modular actions.

0004

An apparatus implementing the disclosed material may include a computer which may include various components known to the art, a computer which may use the method described above to instantiate a broad autodidactic (self-teaching) agency using the network of components within the computer. This agency may perform a parallelized agent-based unsupervised learning and exploration of receptive fields within desired domains, and thus optimize its ability to answer user-defined questions. The computer may be connected to a tester, which tests the actions or sequences proposed by the agents and/or the agency in series and/or parallel using actuators and/or biomicroelectromechanical systems. Chemical and biochemical flow synthesis enables simulated and real experimentation in parallel to study relevant models of biology. The tester may include sensors which measure the result of the actions or sequences performed, and the tester may transmit information gained from the sensors back to the computer to train the agency.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

0005

The accompanying drawings are included to enhance the understanding of the disclosed material, are incorporated in and constitute a part of this specification. The drawings illustrate embodiments of the disclosed material and together with the detailed description serve to explain the principles of embodiments of the disclosed subject matter. No attempt is made to show structural details in more detail than may be necessary for a fundamental understanding of the disclosed material and various ways in which it may be practiced.

FIG. 1 shows an example process for implementation of the method described in the disclosed.

FIG. 2 shows an example agency object structure diagram for implementation of the method described in the disclosed.

FIG. 3 shows an example agent object structure diagram for implementation of an agent described in the disclosed.

FIG. 4 shows an example module diagram for implementation of an extracted module described in the disclosed.

FIG. 5 shows an example apparatus for an embodiment of the disclosed.

DETAILED DESCRIPTION OF THE INVENTION

0006

Learning systems act to extract useful information from raw data. Information is a measurable and tangible physical quantity serving as the inverse of uncertainty: as more novel information of relevance to a given domain is contained within a given system, that system becomes more capable of accurately representing the reality of a given domain. Intelligence is the capability of systems to acquire new knowledge and skills through the acquisition of information. General intelligence is intelligence which is applicable to many domains, as opposed to intelligences which are specialized to function in one particular domain. Broad Autodidactic Agencies (BAAs) are a novel embodiment of generally intelligent systems which contain a distribution of specialist components. These agencies use a group of independent agents termed an agency to extract useful information in a parallelized, hierarchical, modular, and model-free manner from any space of interest. These agencies use this information to optimize further learning and behavior, and store this information in an indexed database and/or dynamic ontology, enabling rapid responses to user queries. A domain, also known as a space of interest, an action and/or behavior space, or an environment, may be a field of human endeavor, for example: the study of ethics, the design of improved mechanical devices, techniques for mutually beneficial interaction with humans, or the development of novel molecular therapies for disease, amongst many potentially valuable domains. Embodiments of the disclosed are a useful implementation of artificial general intelligence.

0007

This disclosure is patentable because it improves upon the prior art in an inventive way, because the specific algorithm is beyond the level of a mental process, because the claims made are for a specific method and apparatus rather than the abstract concept of artificial general intelligence, and because this specific method and apparatus has not been anticipated. Embodiments of the present disclosure may be considered an improvement on the prior art for several reasons, including but not limited to: because information extracted in a modular manner is more easily interpreted and labeled by human users, enhancing the ability to troubleshoot malfunctions; because the agents within agencies learn from their performance and interactions within the domain as opposed to learning solely from the information content of the domain itself; because the learning process extracts useful performance data in the context of changes in network topology of the agents; because of the arbitrarily scalability and applicability of model-free broad autodidactic agencies to any domain; because the agency itself is a higher-level learning system which adjusts a policy of numerous elements to optimize performance; because the learning agency creates one or more searchable indices with artificial intelligence; and because the combination of the trained agency, searchable index, and/or dynamic ontology serves as a compressed form of the information content of the domain of interest; because the agents can forecast and prefer to discover and reuse invariant symbolic abstractions of useful learning techniques like hold-out experiments, or systematic perturbation of the environment; because agents may construct dynamic networks of modular concepts and behavior to represent complex knowledge and skills; because the agents maximize the diversity and magnitude of expected positive consequences of their capabilities while minimizing the diversity and magnitude of expected negative consequences, and minimizing the complexity of the models and processes which control these capabilities; and because this maximization of capability is used to accelerate the learning process itself by planning for learning and acting in ways that tend to maximize learning. These improvements, the mode by which they are achieved, and the specific processes and systems within the disclosed elevate the disclosed beyond the level of abstract ideas and into the realm of inventive elements. As a practical matter, the use of a computer is required to implement the mathematics and parallelized multiscale algorithms of the disclosed on big data at a useful rate, elevating the disclosed beyond the level of mental processes. While the broad concept of artificial intelligence has a long history, this specific combination of innovations and applications has not been anticipated in the prior art to the best of our knowledge. Further, a lack of widely available artificial general intelligence systems and a long history of research failures evidences the non-obviousness of this matter. Finally, no claim is made here to the abstract concept of artificial intelligence, rather, claim is made for one specific method to implement this concept, and claim is made for an apparatus which applies this specific method to the practice of molecular engineering, with great potential for medicine. For these reasons we believe the disclosed to be patentable matter.

0008

To achieve artificial general intelligence is to implement scalable and tractable unsupervised meta-learning in any domain. Unsupervised learning and semi-supervised learning are related subfields of artificial intelligence and machine learning which seek to develop systems to extract useful information and performance optimization from unlabeled raw data, as opposed to supervised learning, which extracts information and optimization from human-labeled data. A fundamental example of unsupervised learning is given by the technique known as non-negative matrix factorization (NMF). This technique decomposes a visible domain matrix **V** into a combination of weight matrix **W** and hidden element matrix **H**, subject to the constraint that all elements of **W** and **H** be positive. The NMF algorithm and approximate versions thereof are commonly used as recommender systems. Broad Autodidactic agency learning is an example of unsupervised learning because it is capable of processing raw data either with or without a prior in the form of a value function. In a preferred embodiment of the disclosed material, a computer-implemented method may explore user-defined domains by using a policy with a number of elements to create a system termed an agency, comprised of a variable number of networked learning agents. This agency is used to explore one or more conceptual and/or real world space(s). The agency uses several innovations in combination to explore said space more efficiently. First, the agents within the agency uses calculations and approximations of information density to decompose the environment and range of possible actions into modules which may be hierarchical, and such agents may allocate more resources and/or attention to modules which offer the most utility for effective learning and performance on desired goals. Further, the agency modifies its policy, describing elements including but not limited to number of agents, network topology, resource usage, inputs, receptive fields (for both , outputs, information measures and approximations and parameterization of agents in a dynamic manner to maximize performance, while learning systems known to the art only modify a minor number of these properties if any. Next, agents within the agency communicate learned information and performance data to other agents and to the agency to increase the overall performance of the agency within the environment. Further, the agency may use the results of the modular extraction and learning process to generate one or more searchable indices of the domain(s) of interest. The trained agency and searchable indices categorize and compress the domain information, enabling rapid responses to queries.

0009

To explore unbounded domains, such as the space of all possible polymers, actions, or action sequences, a policy which minimizes the cost of exploration is desired. The disclosed combines these spaces to employ a heuristic called Lattice Induction to explore available combinations of concepts, actions, and sequences of action. Lattice Induction creates a navigable model of option space $(m := \Psi,\ \textit{where induced R are selected to maximize the utility of } \Delta m)$ by predicting the route of exploration which should yield the highest combination of information value from both $\Psi \wedge A$. The network of available options discovered while selecting behavioral options A and thereby completing a model m consisting of a minimal spanning high dimensional relational network suitable for prediction of traversal patterns across Ψ. The disclosed method maximizes the explored domain of combined spaces $R \wedge A \wedge \Psi$ to progressively explore large and/or unbounded domains. This procedure maximizes the effectiveness of the learning process by systematically increasing the complexity of that which is learned, by learning that which holds the most immediately applicable value (when learned), and by forgetting that which holds little value. Lattice Induction and selection of goal states using this partly hypothetical network of Options leads to progressive exploration of an environment. Selection of goals on the Induced Lattice of Options involves choosing options on state space that increase epistemic information therefore unvisited states have the highest predicted yield. This “simple first” “bottom up” approach explores in a manner similar to biological selective attention, not by pruning options which are too difficult or too easy as some models of interest based attention suggest, but instead by only enacting induced behaviors and behavioral sequences which maximize gains in utility of a model m_x by solving the path planning problem on the combined spaces of the self and the environment. The disclosed method progressively considers more complex model information m while distributing subsets of the history h to the modules inverse to their predicted and measured information density. Using a minimum-complexity-first ordering of induced neural network subcomponents, modules systematically progress toward increased behavioral complexity. The growth in model complexity is throttled by feedback from learned nonlinearities. Thus, the system constructs and attempt to traverse minimum spanning paths on the combination of environmental space(s) and/or internal states m of interest using both a derived map of option transitions and an induced lattice of inferred options.

0010

The ‘curse of dimensionality’ or ‘combinatorial explosion’ is a common problem in highly associative learning algorithms and their underlying structures. Network data structures used as processing motifs solve this issue using ‘depth,’ in the form of many layers only interrelated with adjacent layers. Separation of network learners into independent modules, or ‘breadth,’ can achieve a similar effect by partitioning nodes in a layer into associated groups and pruning or discarding the numerous less useful edges present in a fully connected network with the same number of nodes. The disclosed agencies utilize various heuristics to create a collection of modules. These modules may be clustered for generalization; their density may be minimized to discover invariant models; and they may be used for expectation maximization of utility, value, or novelty. This process relies on the progressive mapping of relationships between expressions of constituent networks and state phenomena which result from experiments. The disclosed mechanism conducts the mapping process to predict of behavioral outcomes via interpolation of or extrapolation from existing examples.

0011

Hierarchical learning seeks to extract features from data in a multiscale manner to model multiscale systems in a more realistic way. As mentioned in the background of the present disclosure, most natural systems are multiscale, particularly systems which are poorly understood at the time of this disclosure. This motivates a focus on multiscale model-free learning to more effectively process information from these natural domains. Broad autodidactic agencies perform a multiscale modular extraction using a multiscale system, and thus BAAs achieve the goal of hierarchical learning.

0012

Learning systems are often grouped into model-based and model-free categories. Models are representations of real or conceptual spaces. A model-based learning system uses pre-programmed models as a basis for the learning procedure, while model-free learning systems can create their new models in the course of the learning procedure. The number of possible spaces and representations of spaces is quite large, and thus a model-based learning system is at a disadvantage when applied to data representing spaces which do not match their particular pre-programmed model(s). Thus, model-free systems, or systems with a broader range of self-generated models, are more broadly applicable in the domain of unsupervised learning, and are critical to systems intended to find novel invariant properties and definitive models of complex systems. BAAs use a hierarchical network architecture in which agents and modules independently generate independent models simultaneously, and then combine these models to optimize their learning and performance.

0013

Broad Learning is the strategy of separating modular elements using expressions of information uptake. Broad Autodidactic Agents use signals from a modular extraction process to select behaviors or sets of behaviors. Specifically, agents choose routes of exploration which maximize the acquisition of information present in the environment. BAAs learn to plan to utilize independent operational modes in combinations which maximize learning uptake in terms of any type of parameter improvement and particularly those which increase temporal abstraction, conceptual abstraction, or coverage of option space by independent modules. BAAs also minimize the complexity of the learning process. Broad Learning Autodidactic Agents learn to plan to learn more modular elements and therefore distinct behaviors, which are automatically separated by prediction accuracy boundaries and/or consistency with beliefs about available trajectories. In contrast to typical Deep Learning approaches, Broad Learning finds explicit partitions between subsets of the data (in histories and contemporaneously) and novel modalities in the form of independent modular elements specialized to those data subsets. What Deep Learning is to “And” and cognitive abstraction, Broad Learning is to “Or” and the separation of learned elements. Broad Learning is the policy of separating modular elements using expressions of information uptake. One such expression is in terms of the Manhattan Length of the matrix which describes the weight space which is operative part of the neural encoding. Another reason to assess value for the behavior of an Agent or Module is the utility of a discovered or induced domain and salience or confidence Module over that domain. Selective Attention and the policy of selecting modules on the basis of their potential contribution to an optimization process is Autodidactic Broad Learning and it too can produce a signal when undergoing improvement.

0014

It is often desirable to reduce the total amount of data necessary to represent a given set of information; such information minimization is called compression. Compression is a natural function of symbolic logic systems such as the human neocortex, because a symbol is a simple link to a larger corpus of information relevant to a linked domain represented by said symbol. For example, a human uses words to represent chunks of information, but a word does not typically encode all of the information relevant to the concept or system symbolized by said word. A human may say the word, ‘house’ far more quickly and simply than a human may be able to explain the full information content of the concept of a house, and so the word house is a compressed symbolic link to the domain information about houses. Broad autodidactic agencies may be described as performing compression because the modules extracted from raw data by BAAs serve as symbolic links to larger amounts of information. Thus, the modular extraction process is functionally equivalent to a compression of domain data. BAA systems can be implemented using a subset of the output space to explicitly control external attention. Elements of the BAA depicted in the Agent Structure Diagram are used to focus internal attention explicitly in the form of evocation through the Selection to Name channel in the case of Selectors, through the gating of reality vs. constructed predictions in the Imagination Planning loop, and implicitly through signals on that same loop and via associated value predictions.

0015

It is often desirable to increase the speed and responsiveness of engineered systems for various reasons. An index is a data structure designed for the purpose of accelerated search in exchange for an upfront cost in processing cycles. Indexing of one or more domain(s) of interest may be performed by systems which may embody the disclosed during the learning process, and such indices may enable more rapid responses to queries submitted by users, agents, or agencies. Said indexing may be performed according to a policy element optimized by one or more agencies, or the nature of said indexing may be controlled by one or more manually input policy elements. Said indices may be stored in short-term random access memory to utilize hardware for further speed increases. A dynamic ontology may be used as a structural prior for knowledge map design.

0016

Overfitting is a problem commonly encountered by machine learning algorithms in which the learning system adapts excessively to the data used for training purposes to a degree which reduces the performance of the trained learning system in regards to new data. This is typically referred to as a failure to generalize. Numerous techniques exist to reduce overfitting, such as dropout, in which neural networks learn repeatedly from training data but exclude random network nodes. This prevents overspecialization of the network and forces redundancy. Broad autodidactic agents abstract Module models into Module Creators and the non-linearity of the Module Creator comes to embrace structure in the domain. The default proposition is that the domain of a Module Creator is unbounded so the basis for a Module is generalized automatically, until discredited on some part of the domain. Further the Broad Learning Rule as applied to Modules in the disclosed method employs a merge function which attempts to anneal Modules or Module Creators with low state or weight covariance. These methods and the finite number of Module Creators effect the minimization or reduction m not overlearn the training data.

0017

Closely related to the concept of overfitting is the concept of generalization. Generalization is said to occur successfully when a trained learning algorithm demonstrates improved performance on data which falls outside of the training set. This implies that the algorithm has learned invariant features from the data of the training set, features which are applicable to test set data which the trained system has yet to encounter. Broad Autodidactic Agencies achieve generalization by inferring the existence of modules suitable to create other modules and module networks from one or more domain(s) of interest. The learning mechanisms which govern the selection and/or creation of novel modules may combine novel behaviors with familiar domains or familiar behaviors with novel domains to enable trained agents or agencies to more successfully predict unknown characteristics of novel stimuli given known characteristics, or to plan combinations of modular behaviors so as to achieve desired goals in novel contexts. Thus, the methods and embodiments of the disclosed enable learning which maximizes generalization by weighting the reinforcement signal by the expected domain and potentially discarding modules or module creating modules which do not generalize well. As one of a variety of examples of the utility of generalization in the disclosed, a broad autodidactic agency may be initialized for the purpose of therapeutic design in medicine. Such an agency may, in the course of the learning process, learn a modular network corresponding to the therapeutic index, and optimize its parameters by studying relevant domains as well as the therapeutic indices of known molecules. Such an agency may develop optimized hierarchical network models for the expected magnitude of efficacy and side effects, and given such training, a broad autodidactic agency may then predict the therapeutic index of novel molecules. Such a trained model may in the course of its activity generate novel hypothetical molecules in a probabilistic manner based on module networks extracted from domains such as pathology, pharmacology, and systems biology. A trained model may evaluate the potential therapeutic index of hypothetical novel molecules as one component of a composite free energy value function, may accept molecules with potential benefit exceeding a learned threshold and potential risk below a learned threshold, and may test such molecules with the assistance of human users or networked tester systems.

0018

Local optima are areas within a value space which are more optimal than nearby surrounding value space yet less optimal than the global optima, the most optimal area in the entire value space. Optimization algorithms which traverse gradients often become trapped in locally optimal areas of value space because they lack the ability to explore less optimal spaces to find the most optimal spaces. Numerous techniques exist to avoid or reduce the problems with local optima, specifically techniques which allow algorithms to explore less optimal spaces in the hopes of finding global optima. One example technique capable of escape from local optima of value space is simulated annealing, which probabilistically accepts lower-valued spaces according to a temperature variable which decreases over time. Broad autodidactic agencies are not stuck in local optima of value space because such agents repeatedly test their belief space, and because agencies use available resources to explore other areas of value space, even if a local optima is found.

0019

Parallelization is a way to increase the performance of most types of algorithms because it allows algorithms to execute the same function on multiple receptive fields within one or more domain(s) of interest. An algorithm which processes receptive fields sequentially will take longer to cover the same volume of data. Broad autodidactic agencies apply parallelization in multiple ways: first, BAAs may explore multiple domains of interest simultaneously; second, BAAs may use multiple agents to explore each domain simultaneously; third, each agent may use broad neural networks to process information from their entire receptive field simultaneously; fourth, the agents within BAAs may predict the free energy value for multiple network topologies of extracted modules simultaneously, depending on available resources; fifth, BAAs and the resulting module indices may simultaneously explore domains of interest and respond to user queries. The disclosed is a parallelizable algorithm, and therefore it may be desireable to instantiate the disclosed methods or embodiments thereof on networks of multiple computers and/or mobile devices in order to apply more resources to the learning process simultaneously, or to distribute independent branches of a centralized learning process for application by multiple simultaneous users and/or multiple simultaneous use cases. It may also be desirable to implement an embodiment of the disclosed methods on one or more central server(s), and allow external computers or devices to communicate with the server(s) for information to achieve various goals.

0020

A computer or mobile device may implement a user interface according to an embodiment of the disclosed. Such an interface may include a number of interface elements to enhance the quality of the user experience as well as the effectiveness of the various embodiments. Interface elements populate the user interface to display information, and may include but are not limited to: an area for text-based communication with the agency or its independent agents; an area to drag-and-drop data files to be processed; a free-energy, profit, utility or surprise surface visualiser; a word cloud visualizer for weighted display of labels for modules extracted; one or more timer(s) for the run time of the learning process and/or the time allowance remaining for a particular unpaid or unsubscribed user; a multiscale network visualization of the connections between modules and/or the structure of the modules; one or more visualization(s) of the potential dynamics of various systems; one or more tools for manual manipulation of policy elements or parameters; one or more monitors for visualization of resource utilization; a web browser window to enable access to links suggested to the user by the learning system; one or more log windows providing verbose descriptions of the actions performed by the system; and/or one or more domain-specific interface element(s) as directed by developers or the agency in question.

0021

Salience and free energy minimization by prediction optimization have summable and otherwise relatable quantifiers, so the disclosed mechanism can exactly calculate the target values for predictors that can estimate information gain of a new Model m or the salience of that model. The disclosed employs a context free method to regularize information for comparison by encoding it as weight parameters in extensible and recurrent neural networks. By comparing both learning rate $\frac{\partial\sigma}{\partial t} = l'$ (derivative) and total learned quantity $\int_H^{\forall t \in h} \frac{\partial\sigma}{\partial t} = l$ (integrand) of prospective or imagined paths the disclosed apparatus can engage in path planning on the space of learning. The differential of information gain resulting from a given policy modification is independent of encoding or domain and can be evaluated using an analysis of free energy. This means that the probability of learning can be added to phenomenological categories like causes or effects; and can be evaluated in a manner similar to path planning by chaining a series of successive predictions (of both environmental states internal activation states or weights) where the resulting predictions (of selected modules) are used as a starting state in hypothetical next steps. Through the iterative projection of predicted states which attempt to reconstruct $m(t) = s(u)$ *where* $u >= t$ it is possible for an agent to perform analysis of untried behavioral processes with the same cognitive processes it has trained while operating on the environment. This projective hypothetical feedback loop enables a differential calculus of predicted information gain which is used by the agent to plan paths of maximal predicted learned quantity l, independent of the environment and dependent on current or local learning rate l'.

0022

The Free Energy Principle (FEP) can be interpreted as a basis to select behaviors using the solution to the formalized Minimization of Variational Free Energy problem defined below.

0023

Minimization of Variational Free Energy Definition (continuous formulation): Active inference rests on the tuple $\{\Omega, \Psi, S, A, R, p, q\}$

0024

A sample space $\{ \Omega \}$ – from which random fluctuations $\{ \omega \in \Omega \}$ are drawn

0025

Hidden or external states $\{ \Psi : \Psi\ x\ A\ x\ \Omega \rightarrow \mathbb{R} \}$– that cause sensory states and depend on action. In an agent these are sometime denoted with $\psi = f_{\psi}(\psi, a) + \omega$

0026
Sensory states $\{ S : \Psi \, x \, A \, x \, \Omega \rightarrow \mathbb{R} \}$ – a probabilistic mapping from action and hidden states. These are sometime denoted as $s = f_s(\psi, a) + \omega$.

0027
Action $\{A : S \, x \, R \rightarrow \mathbb{R}\}$ – that depends on sensory and internal states where $a = -\partial_{\mu} F(s, u)$

0028
Internal states $\{R : R \, x \, S \rightarrow \mathbb{R}\}$ – that cause action and depend on sensory states where $\mu = -\partial_{\mu} F(s, \mu)$

0029
Generative density $\{p(s, \psi|m)\}$ – over sensory and hidden states under a generative model $\{m\}$

0030
Variational density $\{q(\psi|m)\}$– over hidden states $\{\psi \in \Psi\}$that is parameterised by internal states $\{\mu \in R\}$.

0031

The objective of variational free energy based learning is to maximise model evidence $p(s|m)$ or minimise surprise $-log(p(s|m))$. These goals may be expressed as action $a(t) = arg\ min_a (F(s(t), \mu(t)))$, internal states $\mu(t) = arg\ min_{\mu}(F(s(t), \mu))$, where $F(s, \mu)$ is free energy, $E_q[-log(p(s, \psi|m))]$ is energy, $H[q(\psi|a)]$ is entropy, and $D_{KL}[q(\psi|\mu)||p(\psi|s, m)]$ is the Kullback-Leibler divergence. Optimization is done with respect to the relation:

$$F(s, \mu) = E_q[-log(p(s, \psi|m))] - H[q(\psi|a)] = -log(p(s|m)) - D_{KL}[q(\psi|\mu)||p(\psi|s, m)] \geq -log(p(s|m))$$

0032

Some instantiations of FEP lead to intractable marginalisation over hidden states, so surprise is sometimes approximated by an upper variational free energy bound. The disclosed method uses prediction error to approximate the upper variational free energy bound and directly plan traversals such that expected surprise can be forecast as the output of a predictor $v(r|m) = e_{ij} \in r|m$.

0033

The disclosed mechanism employs deep, broad, spatiotemporal pattern learning according to principles laid out in the FEP with respect to R. The disclosed method uses modules with homeomorphic components, so the self similarity in all modular interfaces allows easy hypothetical forecasting toward a goal in m via successive predictions on $s \in \Omega$ shared via a shared bus. An extreme example of the prior technique is entering a mode that forces the path of maximal integration of ψ by m regardless of cost or prior valuation.

0034

The disclosed mechanism employs Proctors which are Selectors on Pools with efficacy measured (salience * domains) either created, fused, or destroyed (or heuristics which utilize information densities to allocate, induce, or deprecate Modules, Creators, or Sequences. Proctors predict on the space of the differentials of the information calculus.

0035

The disclosed modules contain predictors v for e which can forecast pure and composed futures which can represent states which lead to progress on information value gradients based on the standard usage of FEP for agent motivations:

0036

- 1) Maximizing expected utility
- 2) maximizing expected reward
- 3) minimization of surprise (reduction of prediction error)
- 4) minimization of expected cost (including model complexity)

0037

Decision criteria used by Broad Autodidactic Agents to select either a part of an environment or problem domain to visit, plan a transit through several extant traversals, or novel traversals using a behavior option $r \in R$ based on m, which has a domain defined two states on Ω. A projected lattice of Options $O(s_i, s_j)$ and associated models $m \in M$ are converted into a pool of Module Creators. An explicit encoding of a network of Options is retained as a list of paired states and desired solutions. Options and the Module Creators that embody them are discovered in a semi-supervised manner in which additional models m or minimization of expected free energy (maximizing prediction accuracy) is used to partition the data set of $s \in S$ into a load minimizing, modular, self directed (i.e. unsupervised), progressive pedagogy (a teaching strategy which teaches the easiest items first). Further, on domains for which only partial information is available to an agent, at any given time (Partially Observable Markov Decision Processes (POMDPs)), a policy based on the FEP, which is constructive, progressively more complex, and starting with NOP (no operation), is equivalent to a theoretical universal optimal learning agent of the class which currently includes Marcus Hutter's (AIXI), Ray Solomonoff's Theory of Inductive Inference, and Levin Search.

0038

An apparatus embodying the disclosed estimates beliefs in the form of expectations or predictions of future valuations in terms of the FEP, sensory conditions, and learning progress. Such beliefs are approximated using predictors selected on the basis of target values which are sometimes provided by FEP is a prediction of the values of $\boldsymbol{v}$ (t1 + t2) based on any number of internal states and $\boldsymbol{v}$ (t1) where $\boldsymbol{v}$ is a vector representing a high order state of the environment at any given time. Belief formation is equivalent to inductive inference and essential to the process of planning actions on unknown spaces. Using a series of expressions from predictors $\boldsymbol{p}_{1..n}$ the valuation of a proposed path or pedagogy can be evaluated in terms of a matrix $E = \{e_{11}, e_{21}, \ldots, e_{x1}\}, \{e_{12}, e_{22}, \ldots, e_{x2}\}, \ldots, \{e_{1y}, e_{2y}, \ldots, e_{xy}\}$ where columns represent priors not limited to the independent parts of FEP and rows are used to represent the derived or inferentially induced (predicted) values of Goals or Options (and their representation as Modules or Module Creators) which form analogs to epistemic beliefs. E acts as an accumulation of specific values corresponding to specific states or state transitions. E embodies target values which are used to train value estimating predictors. This technique differs from simple propagation of probabilities in a value function in that it is extensible in terms of categorical necessity of particular parts in a plan, as context specific valuations, by adding additional columns. or additional modalities or goals by adding additional rows. In this manner a value table is extended to assess and store learning progress and appropriate valuation of learning progress in both E and the predictors assigned to the subparts of E. In order to evaluate a module using E as a value function each column is normalized independently and the matrix is summed on both basis.

$$estimated\ value\ of\ a\ plan\ r\ \in R\ given\ v\ =\ \sum_{x}^{i}\sum_{y}^{j}\left(\frac{e_{ij}}{|(arg\ max(e_{xj}|m) - arg\ min(e_{xj}|m))|}\right)$$

Where i is the index in a valuation matrix column for a specific type of valuation. This offers an advantage over the mathematics of Markov Blankets because it is based on sums rather than multiplied probabilities of Markov Chains which tend toward 0, in cases when several factors are multiplied or are simply nullified by unknowns.

0039

Fig 1. Shows an example process of the disclosed. The user designates one or more goal domain(s) by entering queries into a UI in process element **1.** The apparatus then decides **2** what resources will be dedicated to an induced set of modules or agents, which may or may not be adjusted at a later point in the process. The modules created in **3** are initialized from different starting points in the neural network encoding space so as to maximize initial and eventual encoding space coverage and the distribution of data domains in the naive example simply distributes data subsample domains. In **4** agents begin to independently seek and begin exploring their designated data subsets. The process of reallocating modules and agents in **6** leads to information being percolated back up **5**, to re engage **4.** In step **7** the agencies or agents decrease the total free-energy in a series exploratory and epistemic discovery operations which map the space of encodings-to-outcomes, while exploring state spaces in both the environment and itself. The agent or agencies that interact with the environment continue to gather feedback and incorporate it into agency policy **8** to better explore **9** utility space. **9.** The system may respond **10** to queries with images and text based on the findings of the exploratory steps.

0040

Fig 2. Shows the Object Structures associated with the top level usage **11** of an Agency **12**. Agencies have controllers **13** that use information density and error boundaries to allocate Agents **17,** according to policies resident in **13** which may be adjusted to redistribute unused resources **16**. Policy elements **14** are identical to Creators except they create Agents rather than Modules. The induction **15** of Agents **17** is followed by the instantiation of a local data reader **18b** which may or may not have an independent trainer **18a** for the purposes of improving domain context dependent parsing of data into **18c** which is a data stream for modules **20b** or their trainers **20a.** A command parser **19b** which may or may not differ from the response engine **22b** both of which may have independent trainers **19a** and **22a**. The command parser **19b** produces data of the prefered format for Agents **17**. The response engine **22b** interprets data from the agent(s) **17** into a format of the user's preference. A data visualizer **21b** for value and network topologies can depict chemosynthesis pathways as a stacked set of independent value topologies as they relate to cost, efficiency, and interest as it is reflected in different value approximations produced by **17.** The data visualizer **21b** may be coupled to a trainer **21a**, for the purpose of maximizing expression utility.

0041

Fig 3. Shows the Agent Object Structure or more specifically the data and scope relationships of objects in the definition of an implementation of the disclosed Agent. An agent is a model for a quasi-independent actor situated on a space which may include a database, the internet, a computer controlled experimental apparatus in a lab, or a map of the real-world location of a robot. The agent is tasked at first with abductively and selectively exploring the environment maximizing yield of data and/or implications thereof. The situated nature of the agent model is reused from the domain of robotics in the field of data science as a reapplication of attentional selection and epistemic and/or spatial exploration to intractable or very large datasets. Autodidactic Agents or Agencies use adaptive attentional selection based on priors and information acquired from interaction between the agent and space to improve upon human directed or arbitrary pedagogies. Various resources **28** are shared. Element **23** encloses the set of direct constituents to an agent which inherit from the Module class. Item **24** is the pool of Sequence Module Creators which take an input s and goal g and elect themselves through a derived confidence signal to contribute to the winner-take-all selection of constituent targets for training **26**. Sequences **26** are sometimes derived from the Selector Class of objects, in which case they map from the input and goal states to a set of nodes that corresponds with the name of the members of action module creators **32.** This means that sequences **26** activate action module creators **32** which create action modules **35** in turn.. Sequence Module Endpoint Predictors **25** estimate the outcome of sequence module creators **24.** This endpoint may be produced by a nonlinear neural network in the relative state variance case or derived from the empirically measured endpoint subsequent to engaging a sequence **26** to produce action modules **35.** Sequence Endpoint Predictors **27** map from g and $s(t1)$ into a state prediction for the end of point of **26** $s(t2)$. Creators (**32**, **24**, and some examples of **37**) can construct Module class objects by acting as Selectors on the Pools and Spaces in **28** with the exception of **34.** Creator predictors (**25**, **33)** not only predict on the space of s but also m so novel epistemic beliefs and the analysis and implications thereof can be estimated for in planning and simple prediction to induce a lattice of options. Feature Space **29** is defined by the states of nodes in an extensible deep feature extracting neural network. Almost all subcomponents of the class Predictor or descended from the Module class use a state in **29** to construct decisions. **29** is singular in an agent because a modular network could suffer performance costs due to redundant independent processing of s. Values created by domain space **30** represent approximate proximity in state space and are calculated independently of module execution to lower the computational cost of executing complete processing of Modules by approximating the odds that a given Module will be evoked in the current context. Action Modules **35** may exist as points mapped to the outcome space of Action Module Creators **32** and Action Endpoint Predictors **33.** Action Modules **35** take values from feature space **29** and pass them through the Action Effector Pool **37** gated by domain space signals**.** End Points **36** map from g and s(t) onto a state prediction for the end point of modules **35.** Value Space **31** differs from the Goals and Valuation Markup **38** in that Goals and Valuation Markup is temporary, serving to buffer predictions of value,while value space **31** persistently encodes long term state values. Action Effectors **37** are retained so that Creators can readily compare neural network instances. Similarly, the Predictor Effector Pool **40** contains all Predictor Effectors used in the Predictors **41**. Like members of the Action Effector Pool **37,** predictor effector pools **40** map a single input to a variable number of 'middle layer' nodes which retain states over the turns of execution (recurrent) and map exhaustively to the input space or value space **31**. The Heat Prediction **39** distributes value to novel theoretical models m of Ω through a Markov blanket of intermodular covariance.

0041

Fig 4 is the Module Diagram depicting the components of modules. In the disclosed method the Module class serves as a base class for the definition of similar objects.Derivations of the module class include Sequence Module Creators, Sequences, Action Module Creators, Action Modules, Goals and Valuation Markup, the class of Selectors for sequencing, which have a ‘disengage’ node to signal the cessation of action, and Creators, which employ a set of selectors to reference and evoke shared components of Modules, Selectors, or Creators. In Fig. 4, a single line running between the Selection and Name elements depicts a control channel equivalent to an address space and control bus on a general purpose computer using a von Neumann architecture. The number of lines on the bus reflects the number of generating and name elements, which are seamlessly extensible and uniquely identify and/or evoke each module with a value of one. This structure allows attenuation, selection, sequencing, and mixing of modules by Selectors. Pools as described throughout this document have similar address spaces. Every Module within an Agent has a unique Name **40** which can directly gate action by Modules in the same fashion as normalized confidence signals **48.** Selection **45** by Selectors calls or evokes Modules according to their qualification via names **40**. Domain space **41** maps the correlation between concepts and actions.. In this drawing the Agent **42** is depicted to clarify its relationships to the modules that it contains. The Gate Junction **43** uses the confidence signal **48** to weigh the individual influences of selection **45,** output **46,** and planning control **47.** Output **46** is the state of the output nodes of one or more effector(s) **44.** Planning Control **47** maps the output of effectors **44** into a control signal for prediction repeaters in the Planning Imagination Loop **52.** The Confidence Signal **48** is a normalized and summed linear cofactor for a module derived from the activation states of module components. The sum of all confidence signals equals 1.0. Value Predictors **49** map input states to value predictions and are trained vs. realized values both given as priors and those derived from learning. Every Module has a driving neural network Feature **50** which is symbolic of a state s or ψ .Predictors **51** accept state data from features **50** and minimize the difference between model predictions and empirical evidence from sensory states. . The Planning Imagination Loop **52** allows state Predictors **51** to project and/or combine their effect into forecast of a future Input s . The Planning Imagination Loop **52** is composed of several subparts: an input feed **53** which can receive signals projected by predictors **51** through a bus interface **55** which in turn is driven by signal passed into **50** via a bus interface front end **54** which is in turn buffered or choked by the prediction planning gate **56**. The pool of modules is denoted with **57.** The activation state of the modules is represented in **58**. The arrows of **59** indicate the separation of modules by the **BLR. 60** are the arrows which represent the extension of constituent networks **50**, **51, 44, 49** in terms of depth (deep learning). Both **59** and **60** are the basis of **61** which is the structural growth signal used to feed back into the current confidence signal given by **48** and the long term valuation in terms of E. **62** is the output choke controlled by the confidence signal **48** which is derived from the domain **41**, name evocation **40**, activation **58**, and salience (free energy reduction) **59** and **60** multiplied by domain volume (determined by **41**) and prior value surfaces which are not pictured and the planning control **47** which gates action during planning.

0043

Fig 5 shows an apparatus **63** implementing the disclosed material which may include a computer which may include various components known to the art, a computer which may use the method described above to instantiate a broad autodidactic agency using the network of components within the computer (**65-70)**. The computer may be connected to a tester **71**, which tests the actions or sequences proposed by the agents and/or the agency in series and/or parallel using actuators **72** and/or biomicroelectromechanical systems, including but not limited to systems for chemical and biochemical flow synthesis **73** and control systems **74** for parallelized experimentation on both real and simulated models, utilizing synthesized molecules to better understand relevant models of biology **77**. The tester may include sensors (**75**, **76**) which measure the result of the actions or sequences performed, and the tester may transmit information gained from the sensors back to the computer to train the agency.

0044

Any system which may receive and execute code according to an embodiment of the disclosed, being an apparatus for practicing embodiments of the disclosed, may be connected to a tester for the purposes of executing modular action sequences commanded by said apparatus for the purpose of testing hypotheses generated in the function of the apparatus as a learning system. Such a tester may execute experiments to gather empirical evidence supporting or refuting the hypotheses generated by said apparatus. Such evidence may be communicated in a standardized manner to the apparatus practicing the embodiment of the disclosed, and said evidence may be used for the purpose of further learning by the agents and/or agencies functioning upon said apparatus. Thus, an apparatus used to instantiate an agency and agents may accelerate the learning process through action.

0045

In one example implementation of the disclosed material, a computer-implemented method may begin with a query to a user for one or more domain(s) to be explored. Given this input, an agency may be instantiated with the target domain and a default policy. This agency may then retrieve a sampled set of information from the target domain. This agency may then instantiate a number of neural network agents with properties defined by the default policy and connections to the environment of choice. Independent agent may be instantiated with properties which differ from the default policy in a probabilistic manner to increase the variety of the agents. Agents may then retrieve data from their respective receptive fields within the domain(s). Agents may then use the calculus of information to decompose this receptive field space into modules and sequences of modules defined by prediction error boundaries. These agents may create one or more searchable indices of domain information and potential actions. The agents may then directly command modular actions or action sequences, or the agency may receive and aggregate actions and action sequences from agents into a different number of modules and sequences. The agency may then estimate the utility of each module or sequence, and the agency may directly command external systems to perform these actions or action sequences in series and/or parallel based on their utility rank, and the agency may then update the policy based on the environmental response to these modular actions.

0046

The disclosed methods top level of usage flow proceeds as follows:

1. Connect to databases, or create them
    a. Unstructured databases for documents, images
    b. Structured databases for pre-organized datasets
    c. Dynamic ontologies for natural language processing
    d. Revisioning databases for events and their context
2. Initialize a user interface
3. Engage a mode which accepts and stores queries
4. Obtain raw data relevant to the next query
    a. Fetch data from linked databases
    b. Fetch data from web search results
    c. Organize data into databases
5. Create agents linked to data and queries
6. Initialize a module by inferential induction of constituent neural networks.
7. Use the initialized module to train a Creator Module to create the initial Module given the state(s) of the starting domain(s).
    a. Creators and Sequence Creators enable re-use of abstracted options
8. Select goal states in the domain or the combined space of both domain state and weight state, the set of weight values in the constituent networks. These may be iterated over according to a progressive ordering.
    a. Weight states result from a learning mechanism which maps domain state space to a future state space.
    b. The weight state represents another form of combined state and value predictor, or maps domain state space into an explicit representation of value which may bear the effect of prior data
9. Select a modular model (in the form of an appropriate Creator Module) or Sequence of modular models and associated Creators, to create an appropriate module or modules

which are then selected on the basis of confidence and their contribution to reaching the goal state.

   a. Normalize and sum the values on the realized valuation surface E, or
   b. Rank, normalize, and then sum the values in E which correspond to a given Creator.

10. Select modules.
    a. Either normalize and sum module confidence signals, or
    b. Rank and normalize module confidence signals.
11. Experiment.
    a. Engage one or more module(s) as controller to the agent.
    b. Evaluate as successful when the agent reaches a stop point within a small percentage (10%) of the spanning distance of the endpoint or goal.
    c. The agent reaches a stop point under conditions which may include:
       i. It exceeds a boundary defined by a Manhattan distance hyperellipse with foci including the start and end point of the option(s) represented by the module(s). If $a = start$, $b = goal$, $c = present\ location$ are points $\in s$ and ε is an error bound decided according to a linear relation of the option length $\varepsilon = x\,||a - b||$ then the error boundary is defined as $\varepsilon \geq ||a - c|| + ||b - c||$.
       ii. It fails to make incremental progress with respect to incremental environmental state change.
       iii. It fails to meet self imposed deadlines which may be created by temporal span predictors.
12. Adapt.
    a. Feed observed phenomena from the experiment which used the engaged module(s), back into the control system for the agent.
    b. Construct an error return surface for each constituent learning element in the evoked modules
    c. Construct a new Creator module using retrained module(s) to approximately create both prior and retrained modules. Use data from from the agent-environment level including evaluations such as:
       i. exogenous factors which supply feedback to predictors
       ii. a reward value prior and/or a differential thereof,
       iii. experiment span in abstract and/or real spaces,
       iv. time,
    d. Augment data with self-evaluations of learning effectiveness, including but not limited to:
       i. free energy value of alteration (over prior space states);

ii. error reduction or integrated error reduction (over prior or current space states),

iii. minimization of variational free energy.

13. Repeat steps 3-12 while offering responses
    a) update histories in modules for retrospective evaluation
    b) compress used Behavioral Modules into history instances for the creator modules
    c) update the shared neural expression-to-phenomenal map for both modules and module creation.
    d) deform the lattice of option space
        i) Option space is composed of environmental state, action space, and/or state space
        ii) Option space maps environmental states along with speculative and informed transition enacting modules.
        iii) Option lattice deformation involves inductive selective unfolding of an option space for an agent to adapt to one or more domain(s) according to the free energy principle.

0047

Agents explore environmental space Ω by maximizing length and covariation of their history h. Agents explore epistemic space while maximizing information. More specifically, they maximize the total amplitude of the covariance matrix between module activation states over time + |difference of encodings| under the resource ideal (number of modules available given process and memory resources). Pools are filled up to and proctored to remain under resource caps via the following algorithm:

1. If below the resource ideal, induce new modules.
2. If at the resource ideal reorganize modular domains, maximize expression variation to induce naive change in environmental states S
3. Afterwards, utilize the expression-to-state map to calculate goals on the basis of their distance from the cluster of environmental states in S, work backward to a maximum probable model m using confidence bidding based on predictions, domain, state, and credibility of predictions and the expression-to-state map or expression-to-option map to predict a solution to reaching the goal.
4. Alternatively integrate and delete modules on the basis of net reduction of symbolic information expressed as the size of the expression of the deleted module and assessed in phenomenological terms as the sum of (confidence * state (or salience) * domain) over all time instances from the history.
5. This is the application of the Broad Learning Rule (BLR):
    a. maximize encoding coverage of m to maximize the likelihood $\psi|m$ is already handled by some module on m.

6. The structural improvement signal, derived from evaluation of differentials produced by BLR, deep learning progress, or successful temporal extension of a process, and minimization of hidden states Ψ by maximizing covariance of states in the history $s \in h$, form an approximate semi-supervised valuation map for information gain in $m \wedge \Psi$. Real valued assessments of learning, in the form of structural growth signals, are fed back into E as realized instances of structural growth which fulfills the imperative of **BLR** and represent an expression of the maximization of m according to an application of the FEP. Predictors of E and the disclosed method to calculate an additional signal to complement the process of maximizing coverage of m by pure FEP, are used to project values to unknown states, untried modules, and novel domains using the heat map as a buffer for the projection markup space. The difference between the realized valuation surface E and the heat map is used to retrain the value predictors which implicitly estimate the learning rate $l' \in m$ which biases behaviors and behavior selection toward epistemic growth.

0048

According to an implementation of the disclosed broad learning rule, modules are automatically separated by the selection of a sequential subset of the experienced states $h = \{s_0, s_1, ..., s_n\}$ for which the creation of a novel predictor and therefore novel module reduces the total error domain for the agent and the free-energy present in the combined system consisting of the environment and agent. The upper bound for available modules depends on available resources and the policy of the agency. A stochastic process configures module boundaries to minimize the number of utilized modules and search for instances of modular compression. Agents have three shared references which are used to perform the modular compression operation.

1. Intermodule covariance matrix θ relating modules $c \text{ and } d$
2. Intramodule covariance matrix θ' relating states $i \text{ and } j$
3. Access to the local history of a state $\mu(t) \in R(t)$

0049

To get the empirically adjusted distance between two networks with the same topology, take distance element $\delta \in \Delta$ where $\Delta = w_c - w_d \text{ and } \delta_{ij} = w_{cij} - w_{dij}$, scale each member δ_{ij} by the relative difference expressed by the combined covariance field $\{\lambda \in \Lambda : \mu(t) \,\forall\, \theta'_c \text{ and } \theta'_d := \lambda_{cij}\}$ and take the Manhattan length of that field from unity as a singular measure.

Manhattan Distance $d_1(p, q) = \|p - q\|_1 = \sum_{i=1}^{n} |p_i - q_i|$

0050

Error is reduced by minimization of the difference between historical activations and targets. In the case of modules, this is the difference of the end state and goal. In the case of predictors this is the difference of a target value and the predicted value. In the case of selections this is the difference between the desired name and the expression issued by the selection subpart of a Selector. Error minimization is approximated (while also minimizing the co-variation on $\psi \cup m$) by inducing changes in m or ψ which may result in error correcting modules or even an attempt to restructure the environment to minimize m. By minimizing the expression cost of representations of $\psi \in m$ while maximizing coverage (utility), valuation by priors, maximizing expectation while simultaneously minimizing variation across an adjacent subset of history h which confirms the boundaries of modules and maximizes the available difference between those elements of m that refer to the Ψ and some future state s.

0051

The disclosed apparatus preserves topological homomorphisms with empty additions $\forall i, j, k \neg\delta,\ w_{ijk}$ across similar subcomponents of all Modules k. The homomorphism serves several purposes: crossover compatibility of interpolated modules, compatibility with interfaced elements: busses, input and output layers, etc., and allowing edit distances between any two similar networks to be easily determined.

0052

Selective Attention in the disclosed method is a function of the induction of traversals, either by expanding a complete option transition lattice or closing it in reward / cost first order, or by creating informed goals and counting forward through an ordered compact set of encodings for constituent networks. The three following rules are used to infer and induce novel behaviors or goals. The iteration of this expansion to a variable depth, while temporarily buffering predicted learning value rewards, allows the projection of values, including those which would be learned under a speculative or imagined premise, beyond the existing cluster of known Options in the state space.

0053

Pure Induction of A Novel Module or Goal using linear cluster expansion:

1. Calculate center of the module cluster $C = \frac{\sum_{i=1}^{n} \{m_1, m_2, ..., m_n\}}{n}$.

2. Construct a line through the current state with C as an endpoint.
3. Project that line beyond the current state a distance established by the average span of module start and end points.
4. Use the new Goal as an endpoint and domain qualifier.

0054

Informed Induction of a Novel Module by using a Goal and a differential established in the expression-to-outcome mapping.

1. Take the slope of the differential between the current state and goal.
2. Find an existing nearest neighbor if possible.
3. If no example is available, use the current next in the inductive counting series.
4. If only one example is available, negate the weight of the first network.
5. If only two examples are available and they are equidistant, pick the next nn derived from the current next in the counting series.
6. If there are three or more known Options take the nearest neighbor's effector network and the current next in the series and perform a logical 'or' with current next member of the sequence of networks. Do not increment the sequence index counter.
7. Else, use the current Creator to create to create a new module to suit a direct extrapolation of the current Goal.

0055

Informed Inferential Induction of a Novel Goal:

1. If no value surface can be constructed. Take the next network encoding from the ordered set.
2. Extrapolate known value surface of prior goals into the heat map. Find the top value and use its location as a Goal.

0056

Curation of the shared pools of component parts and common resources in general is handled by the Agent Class of objects and can utilize variable or constant sized lists of components. In the case of Module Creators, when the size limit is met the disclosed algorithm attempts to fuse the nearest neighbors in encoding space and split the highest error Module Creators through induction and subsequent experiments. In this manner the lattice over the already traversed domain is continually revised to fit the environment given the available resources.

0057

The disclosed method uses multiple disparate learning domains to create meta-operative autodidactic or scientific behaviors, as described by Chalmers to evolve general learning rules. Testing for efficacy is facilitated by comparing other prior goal directed machine learning techniques with a BAA which has been pretrained on other domains. The emergence of high order behaviors, like reductionist single variable experiments, is tested in a series of special problem domains. In the case of scientific experimentation those domains are characterized (in the prior valuation) by a global value optima which is conditional upon the rest of the state of the environment. This creates a surface which exaggerates and seeds the valuation of minimization of potential noise. This testable property can be viewed as the sapient version of the policy of finding reciprocal paths in free space. The disclosed apparatus can engage in a mode of planning successive option paths $\{s(t=0) \rightarrow \mu_0 \rightarrow a_0,\ s(t=1),\ \ldots s(t=n)\}$ that minimize co-variation on $\psi \cup m$, and by executing those plans, produce both order in of the self and environment. The disclosed apparatus will accumulate a variety of experimental methods like introducing noise into systems with intent of discovering causal relations where only correlations are available from static perceptions.

0058

In one particular example of an apparatus according to an embodiment of the disclosed, broad autodidactic agency learning may be used to enhance the development process of novel therapies for disease. The process may begin with the user entering queries of relevance to this topic into a user interface, for example the set [ethics, chemistry, biochemistry, biology, anatomy, physiology, pathophysiology, medicine, toxicology, immunology, pharmacology]. The process may continue with the instantiation of an agency targeted to these domains and holding a default policy, where said policy may include information gained and optimizations performed based on learning from previous applications of the method. The agency may next perform a preliminary search of the internet or local database to retrieve receptive field data. The agency may next generate a set of neural network learning agents to explore the various receptive fields of the domains of interest. These agents may next perform further data retrieval actions to obtain raw data for unsupervised learning. The agents may then perform a modular extraction on their respective raw data, obtaining a set of modules bordered by regions where optimized models of the module contents have increased predictive error. The optimization of module models is performed with respect to a free energy surface which represents both the cumulative positive utility of diverse knowledge and skills and the cumulative negative utility of various costs including model complexity. The modular extraction process may repeat and may generate networks of modules. With each iteration of the modular extraction, learning rate l' and total learned quantity l are used to optimize agent parameters and agency policy. These optimizations, along with exploration of the target domains, may lead to creation and/or pruning of agents, and addition of new receptive fields in domains associated with the target domains. As learning progresses, the system extracts information from the domains of interest and indexes this information in the form of databases, modules and/or a dynamic ontology. The system may keep a portion of available resources in reserve for goals ancillary to the learning process itself, including but not limited to the operation of one or more user interface(s) on one or more computer(s), network(s), and/or device(s); to hold a reserve of learning capability to explore novel subdomains of relevance at a particular period of function of the system; and to process user queries with the module index and the trained network. User queries may be answered in a variety of ways including but not limited to text, audio, images, and/or video. The system may accept data directly from users who are interested to contribute.

0059

For one of a variety of examples of said accelerated learning through action by broad autodidactic agencies, an apparatus practicing an embodiment of the disclosed may be used for the purpose of therapeutic design in medicine. Such an apparatus may use raw data from any relevant source, and the knowledge and skills extracted from said data, to select sequences of modular actions which generate new data from experiment in order to more optimally explore the space of the domain and maximize both learning rate l' and total learned quantity l. A tester according to an embodiment of said method might perform a series of experimental actions based on learned knowledge and skills, experimental actions which may synthesize various molecules and expose said molecules to tissue spheroids and/or other biological models as physical, 3D models recapitulating a process of interest to the field of medicine. Sensors including but not limited to spectroscopes for molecular identification and microscopes and/or cameras for visual observation may obtain time series data on the nature of the molecules within the tester and spatiotemporal data on the state of the spheroids and/or other models of interest. Said data may be used to study the efficacy and side effect profile of hypothetical small molecule, biomolecule, or nanotechnological therapeutics, said data being processed by the broad autodidactic agencies within the apparatus and/or by independent human observers. In this manner, an apparatus practicing an embodiment of the disclosed may perform a closed-loop feedback optimization and search for more optimal therapeutics for a range of pathologies. Similarly, said feedback optimization may include the use of hypothesized therapeutics on live animal models of various pathologies, given that appropriate ethical standards are followed by practitioners of said embodiment of the disclosed. Such feedback optimization of therapeutic modes may result in a list of promising therapies for one or more diseases of interest, and such a list may be sorted and/or ranked in various ways including but not limited to the predicted distribution of therapeutic index and/or efficacy of said therapies.

0060

Embodiments of the disclosed may function in the context of a variety of general-purpose or application-specific physical information processing systems. FIG. 5 shows a conceptual model for the function of a general-purpose computer. This form of computer is known to the art and thus beyond the scope of the disclosed, and typically contain one or more central processor(s), one or more graphical/parallel processor(s), short-term memory (RAM, ROM, or flash memory), long-term storage, input-output control systems, all being interconnected with a bus for data communication. Such computer may input-output mechanisms including but not limited to wireless internet, telephone, ethernet, USB, CD/DVD/blu-ray drives, mice, keyboards, monitors, scanners, touchscreens, camera(s), controllers, joysticks, backups, and floppy disc drives. Any and all components of computing systems useful for embodiment of the disclosed may be located external to the relevant computer system and connected by other interfaces. A computer may embody the disclosed without possessing all of these elements.

0061

The short term memory of computers which may embody the disclosed can store the basic input-output system, the operating system, and code for programs and/or applications currently in use. Programs and/or applications not in active use may be stored in long-term memory. Code to implement the disclosed may be stored in any system suitable for the storage of data and/or information. When a computer, computer network, mobile device or application-specific device receives and executes code according to an embodiment of the disclosed, said system becomes an apparatus for practicing embodiments of the disclosed.
The foregoing descriptions of various embodiments of the disclosed are not intended to be exhaustive and/or limit embodiments of the disclosed.

0062

What is claimed is:

1. A method for artificial general intelligence which uses the techniques of the disclosed to instantiate a modular neural network learning agency which accepts user queries as one or more target domain(s), wherein said agency probabilistically optimizes a policy which controls elements which may include resource usage, number, inputs, outputs, receptive fields, network topology and other parameters of agents, a policy which guides the creation of a variable number of network learning agents with variable receptive fields within said domain(s), wherein said agents perform an unsupervised extraction of modules and module networks from the domain and space of possible actions using free energy optimization, lattice induction, information calculus, and attentional planning, wherein said agency learns by training the parameters of neural networks within agents within said agency to globally optimize a free energy function, wherein said agents communicate information to other agents and the agency in a standardized manner, and wherein said agency uses information from the domain and the performance of the agents within the domain in a distributed manner to optimize the policy and optimally combine extracted modules to more optimally explore the space of domain, agents, and agencies, and where said modules and module combinations are stored as a compressed index and/or dynamic ontology of domain information to respond more rapidly to user queries.

2. An apparatus which embodies the disclosed, comprised of a variable number of computers which may be connected to the internet and which may include dedicated parallel processing, which may use the methods of the disclosed to instantiate a learning agency of network learning agents according to a probabilistic policy within one or more domain(s) of interest, agents which extract modules and information from domain and action space using free energy optimization, lattice induction, information calculus, and attentional planning,, modules and information used to update the policy of the agency, modules or module networks which are selected to optimize a composite utility function and transmitted to one or more tester(s) which implement(s) the modular actions or action sequences for experimentation using tools including but not limited to actuators, microfluidic bioMEMs, flow molecular synthesizers, spectroscopes, biological models, microscopes, and cameras, wherein the responses to implemented action(s) are detected by sensors, information from which is transmitted back to the agency and agents for more learning.

ABSTRACT OF THE DISCLOSURE

0063

The present disclosure pertains to the fields of artificial general intelligence and molecular engineering. A computer-implemented method generates an agency with a policy which determines the further generation of network learning agents. These agents perform a modular decomposition of the environment and action space using the calculus of information. These modules are then combined to explore environment and action space in a manner which becomes more optimal over time. The agents thus learn to plan to learn to explore, and this exploration is used to generate optimal control paths, searchable indexes and/or dynamic ontologies of domain(s) of interest.

DRAWINGS

OATH OR DECLARATION

I, Bion Howard, swear that I am an original joint inventor of the disclosed material, and that I understand that any willful false statement made in the declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five years, or both.

____

Bion Howard

____

Date

I, Correy Kowall, swear that I am an original joint inventor of the disclosed material, and that I understand that any willful false statement made in the declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five years, or both.

____

Correy Kowall

____

Date

Witnessed by a notary:

____

Notary’s Signature