The Computational Gauntlet of Human-Like Learning

Pat Langley
Institute for the Study of
Learning and Expertise
Palo Alto, California

This research was supported by ONR Grant N00014-20-1-2643 and by AFOSR Grant FA9550-20-1-0130. Thanks to many colleagues for discussions that led to these ideas.

The Problem

Despite its modest origins, machine learning has come to play a dominant role in artificial intelligence.

Statistical induction on massive data sets has led to impressive results in multiple areas, including:

- *Computer vision*
- *Natural language*
- *Game playing*

But in the process, the field has lost its intellectual diversity and abandoned its conceptual roots.

Claim: *We can remedy both drawbacks, and devise even more effective systems, by focusing on* ***human-like learning***.

Example: Learning Mathematics

Consider how students master mathematics in our educational system by learning, successively, to:

- *Recognize and write digits* 3 7 6 4
- *Retrieve and use arithmetic tables* $2 \times 1 = 2, 2 \times 2 = 4, 2 \times 3 = 6$
- *Carry out multi-column addition, subtraction* $\begin{array}{r} 327 \\ -\ \ 64 \\ \hline \end{array}$
- *Simplify complex fractions* $3/4 + 1/8 = ?$
- *Solve algebraic equations, word problems* $7x - 5 = 2x$

This curriculum takes years, but it does ***not*** require thousands of instances per concept or skill.

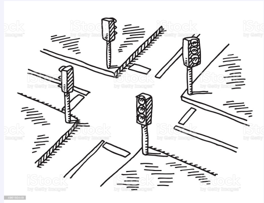
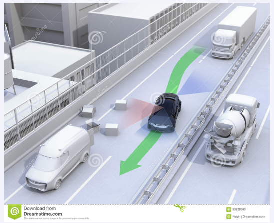
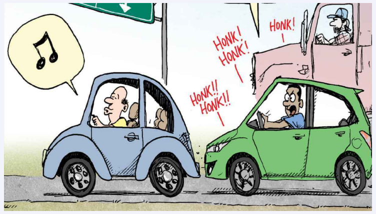
The trajectory of human learning here differs drastically from how we currently train machines.

Example: Learning to Drive

Now consider how people – often teenagers – learn to drive an automobile by acquiring:

- *Categories for roads, lanes, intersections, signs*
- *Skills for changing lanes, passing, turning, parking*
- *Social norms for driving, including laws and customs*

Mastering these elements requires training and practice, but most drivers are reasonably good after a short course.

Unlike statistical learners, humans do ***not*** need millions of miles' experience to acquire basic competence.

Machine Learning: A Brief History

Machine learning was founded more than four decades ago as a spinoff of mainstream AI:

- *First workshop 1980, Journal 1986, Conference 1988*
- *Focused initially on acquiring symbolic structures*
- *Concerned with automating creation of expert systems*
- *But also with modeling* ***high-level learning in humans***

This paradigm was successful, producing demonstrations of new capabilities and deployed systems.

During this early period, links to cognitive psychology played key roles in the field's aims and progress.

Machine Learning: More History

The new discipline of machine learning evolved rapidly and, by the mid-1990s, it had:

- *Redefined learning as improvement of performance*
- *Broadened to include statistical methods and neural nets*
- *Adopted controlled experiments for evaluation purposes*
- *Birthed the closely related discipline of data mining*

Each step seemed a positive one but also took the field further away from its psychological origins.

More recent results on learning with deep neural networks have only worsened the situation.

Current State of the Field

Machine learning is widely viewed as a great success, but the most popular approaches depend crucially on:

- *Collection of gigantic training sets*
- *Storage of these data in massive memories*
- *Processing them on arrays of CPU servers*

Progress is often measured using mindless ‘bake offs’ that can rely on questionable metrics.

- Recent results with large language models are impressive but they are fragile and depend on skilled prompting.

These ‘state of the art’ learning systems bear little resemblance to the way humans acquire expertise.

Constraints on Learning

To develop AI systems that learn like people, we must first identify the core features of human learning:

- *High-level regularities* observed in human cognition
- *Recurring phenomena* that hold across many settings
- *Laws of qualitative structure* (Newell & Simon, 1976)
- ***Not*** detailed models that fit specific experimental results

Insights about the character of human learning can serve as strong constraints on system design.

But how might researchers use such constraints effectively?

A Computational Gauntlet

A ***gauntlet*** is a passage, lined with armed adversaries, that one must traverse to survive a trial.

- We can use characteristics of human learning to devise a *computational* gauntlet.
- Each constraint introduces a new threat that AI systems must encounter and overcome.
- To reach the end, they must make it past each obstacle along the dangerous path.

This offers a radical alternative to the performance-oriented ‘bake offs’ that now guide the field.

But what aspects of human learning can serve this purpose?

Modular Structures

One basic feature of human learning (Bower, 1981) concerns the nature of acquired content:

- ***Learning involves the acquisition of modular cognitive structures.***

This does not specify the structures' details; only that expertise consists of ***discrete*** mental elements.

This contrasts sharply with the idea that learning only revises ***parameters*** in an existing monolithic structure.

E.g., most neural networks alter the weights on links between nodes that are given in advance, rather than acquired.

Composable Elements

A second characteristic is enabled by the first one and often associated with it closely:

- ***Learned cognitive structures can be composed during performance.***

That is, relevant elements of expertise are accessed and then combined *as needed* to produce behavior.

E.g., planning systems and sentence parsers compose learned structures to address multi-step tasks.

Neural networks propagate activations over links, but many question their capacity for compositional reasoning.

Examples of Composable Structures

There have been many proposals for modular, composable structures from psychology, AI, and linguistics:

- *Chunks* (Miller, 1956)
- *Exemplars / Cases* (Schank, 1982)
- *Grammar rules* (Chomsky, 1965)
- *Production rules* (Newell, 1966)
- *Planning operators* (Fikes & Nilsson, 1971)
- *Stimulus-response pairs* (Skinner, 1953)

These differ in details, but all are composable at performance time, qualifying them as ***generative models***.

Piecemeal Acquisition

Another feature involves how people process experiences and create new structures. In particular:

- ***Expertise is acquired in a piecemeal manner, with one element added at a time.***

Humans learn one cognitive structure, then another, continuing until they achieve broad coverage.

E.g., they acquire each concept and skill for mathematics and driving in a reasonably independent manner.

They do not create complex models *en masse*, as done by most methods for statistical induction.

Incremental Learning

Another processing constraint focuses not on the knowledge elements but on handling training cases:

- ***Learning is an incremental activity that processes one experience at a time.***

This is linked to on-line approaches that interleave learning tightly with performance mechanisms.

E.g., people process the training events for mathematics and driving in an ongoing stream, not all at once.

Incremental and piecemeal learning can co-occur, but they are distinct; e.g., most rule induction is piecemeal but batch.

Guidance from Knowledge

The sequential nature of human learning also means that later processing builds on previous results:

- ***Learning is guided by knowledge that aids the interpretation of new experiences.***

Because acquisition is piecemeal and incremental, it occurs in the context of existing mental structures.

E.g., complex skills for both mathematics and driving build on simpler ones acquired earlier in training.

Knowledge is central to human learning but it receives limited attention in data-intensive paradigms.

Rapid Acquisition

A final characteristic of human learning, enabled by piecemeal, incremental, and knowledge-guided processing, is that:

- ***Cognitive structures are acquired and refined rapidly, each from small numbers of training cases.***

The claim is ***not*** that ***all*** expertise comes from a few instances, but that we learn modular elements this way.

Human learning curves in mathematics and driving, which plot performance vs. training cases, are very steep.

Again, this diverges from statistical induction’s dependence on thousands or millions of items.

Critiques and Responses

- Why change paradigms when deep learning works so well?
  - Because it is ***not*** as data efficient as human learners. And we should understand the ***entire space*** of learning methods.
- Why build AI systems that learn like people? (planes ≠ birds)
  - Birds offer many insights into flight (e.g., *lift*, *thrust*, and *drag*). And we now have small drones that fly very much like birds.
- Is this about structure learning vs. parameter estimation?
  - No, the question is whether a learner relies *only* on parameter estimation or, like humans, acquires new structures.
- Are you saying that human learning never involves statistics?
  - No, but the rapid acquisition of new structures is a distinctive feature of human learning; statistics is a *background* process.

Examples of Human-Like Learning

The literature contains some cases of human-like learning that count as positive instances:

- Fisher’s (1987) *Cobweb* – constructs a probabilistic conceptual taxonomy from unsupervised training cases
- Minton’s (1990) *Prodigy* – acquires control rules from planning traces to guide search on future problems
- McClure et al.’s (2015) *SAGE* – invokes structural analogy to learn relational concepts from training sequences
- Muggleton et al.’s (2018) *meta-interpretive learning* – acquires logic-based concepts very rapidly

These systems fare well on the gauntlet and offer useful role models for the research community.

Probabilistic Concept Hierarchies

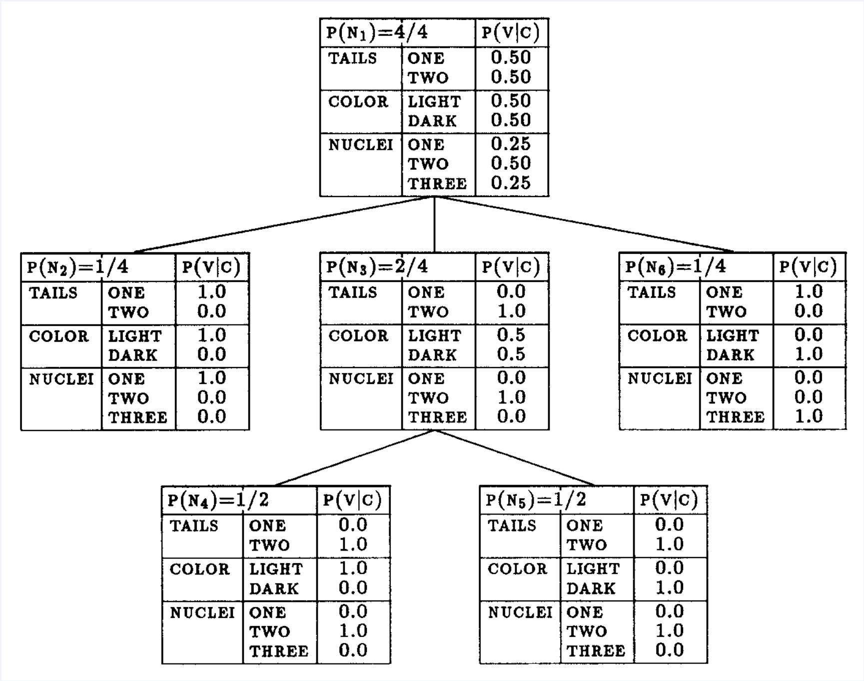
Fisher's (1987) Cobweb is a process model of categorization and category learning that:

- Constructs a taxonomy of probabilistic concept descriptions
  - Terminal nodes are cases; nonterminals summarize descendants
- Sorts new cases down the hierarchy guided by *category utility*
  - On halting, uses selected concept node to predict missing values

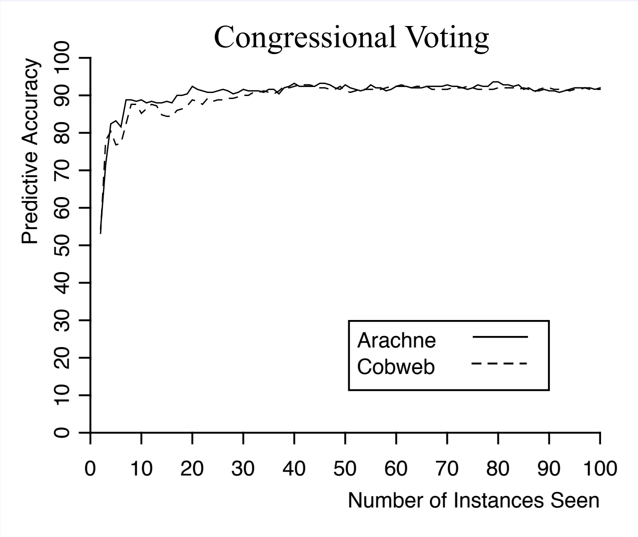
Cobweb unifies ideas from *decision trees*, *naive Bayes*, and *nearest neighbor* classifiers.

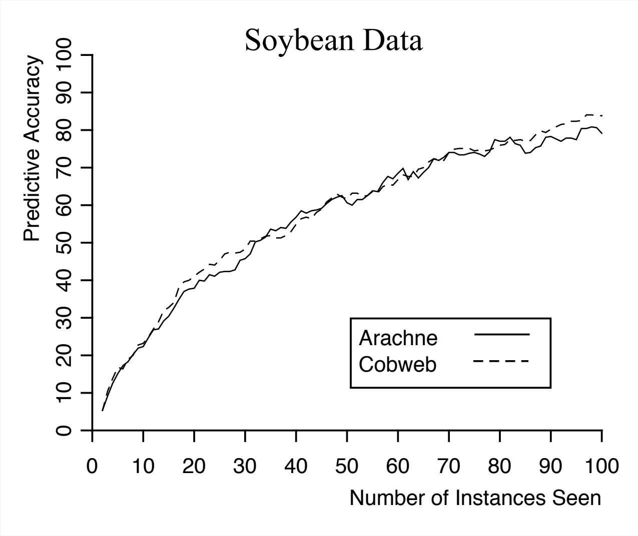
The system replicates well-known psychological phenomena like basic-level categories and typicality effects.

Cobweb Examples and Results



A Cobweb Hierarchy





Category Utility

$$\frac{\sum_{k=1}^{K} P(C_k) \sum_i \sum_j P(A_i = V_{ij} | C_k)^2 - \sum_i \sum_j P(A_i = V_{ij})^2}{K}$$

Early extensions to Cobweb let it handle numeric attributes and structured data, as well as mitigate dependence on the order of training cases.

Learning Concept Hierarchies

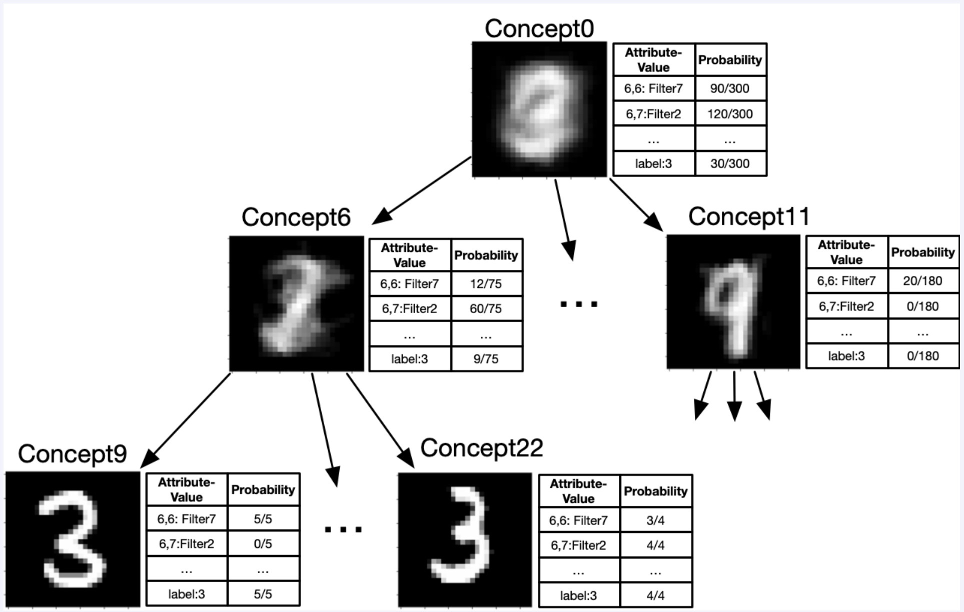
Cobweb interleaves learning with categorization to construct its hierarchy from unsupervised data by:

- Updating the distribution for concepts to which a case is sorted
- Extending the taxonomy downward on reaching a terminal node
- Adding a new branch when no children are similar enough
- Merging / splitting a node's children if category utility improves

The system learns categories very rapidly in an incremental, piecemeal way that builds on prior acquisitions.

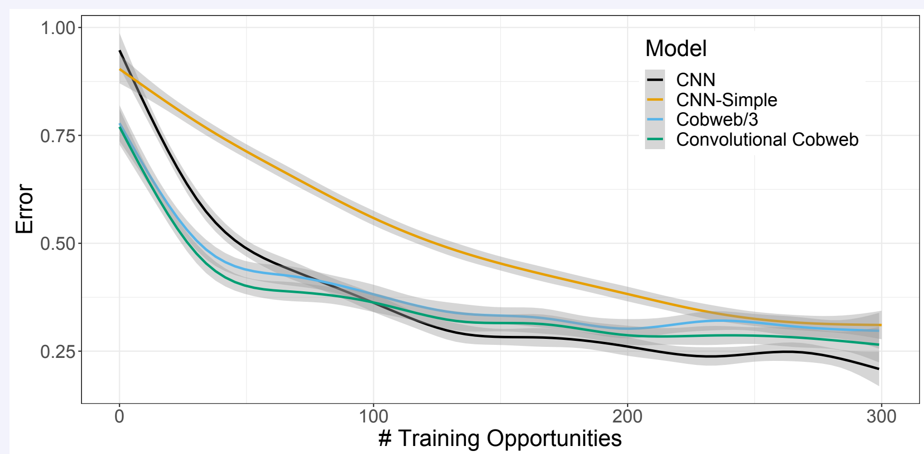
Each Cobweb training case leads to both the ***creation of new cognitive structures*** *and* ***revision of statistical summaries****.*

Recent Extensions to Cobweb



The original Cobweb dealt with 'tabular' encodings, but an extension incorporates *convolutional* processing of images.

MacLellan and Thakur (2021) report comparisons of the extended Cobweb with a convolutional neural network on the MNIST image repository.



We have also developed a *contextual* version of Cobweb that distinguishes word senses in textual sequences.

Recent efficiency improvements should let it acquire a *large language model* from a corpus with millions of words.

Knowledge-Guided Planning

Minton's (1990) Prodigy offers an architecture for knowledge-guided planning that:

- Encodes knowledge as domain operators and control rules for selecting or rejecting goals, operators, or bindings
- Invokes means-ends analysis to carry out goal-directed search in a space of problem decompositions
- Uses control rules to reduce search by blocking poor choices and favoring good ones

Prodigy's reliance on means-ends analysis is consistent with studies of human problem solving.

The system unifies AI's four key ideas: *reasoning*, *heuristic search*, *knowledge*, and *learning*.

Learning Search-Control Knowledge

Prodigy acquires planning expertise from traces of its own search processes by:

- Using a generic theory of problem solving to explain why each choice led to success or failure
- Compiling each explanation into a control rule for selecting or rejecting a goal, operator, or binding
- Collecting statistics on these rules' utilities to determine which ones to retain or abandon

The system substantially reduces both nodes examined and CPU time to solve new problems in many domains.

*Prodigy combines **rapid generation** of new structures through knowledge with their **gradual evaluation** by statistics.*

Prodigy Examples and Results

A Prodigy Rejection Rule

(FAILS *goal node*)
if (AND (CURRENT-GOAL *node goal*)
(MATCHES *goal* (ON *x y*))
(OR (AND (KNOWN (ON-TABLE *y*))
(EQUAL *x y*))
(AND (KNOWN (ON *y z*))
(EQUAL *x y*))
(AND (KNOWN (HOLDING *y*))
(EQUAL *x y*))))

A Prodigy Selection Rule

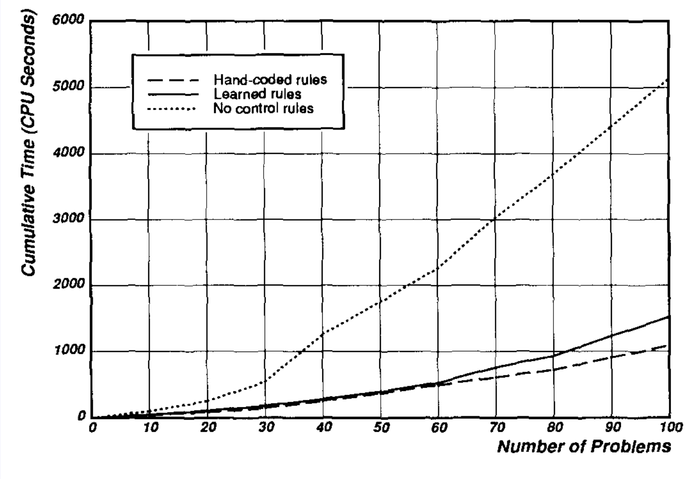
(SUCCEEDS *goal node*)
if (AND (CURRENT-GOAL *node goal*)
(MATCHES *goal* (HOLDING *x*))
(KNOWN *node* (IS-BLOCK *x*))
(KNOWN *node* (ON-TABLE *x*)))

Prodigy's Utility Criterion

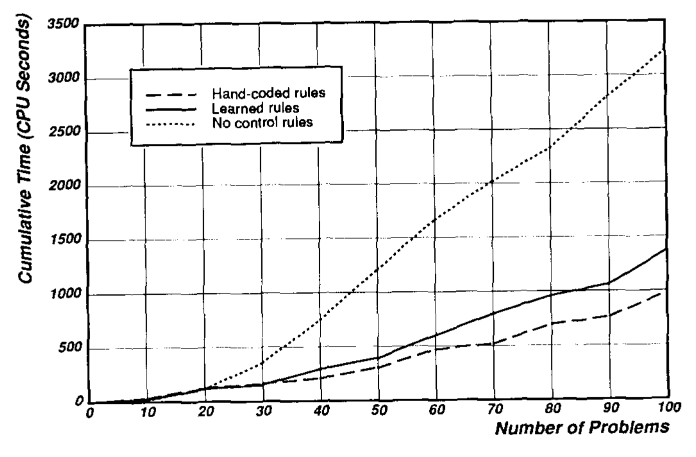
$$Utility = (AvrSavings \times ApplicFreq) - AvrMatchCost \text{ ,}$$

Prodigy's many successors supported planning by abstraction, analogical problem solving, and learning for plan quality rather than efficiency.

STRIPS Robot



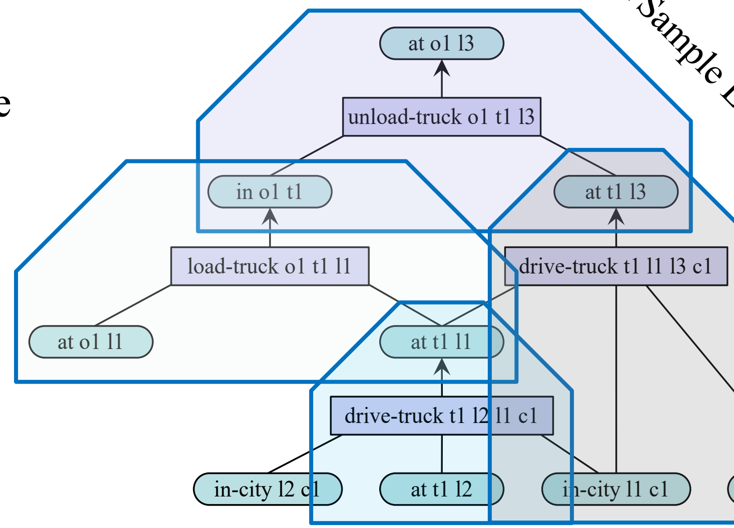
Scheduling



Hierarchical Problem Networks

HPNL (Langley, 2022) learns *hierarchical methods* from sample solutions to decompose complex problems into simpler ones.

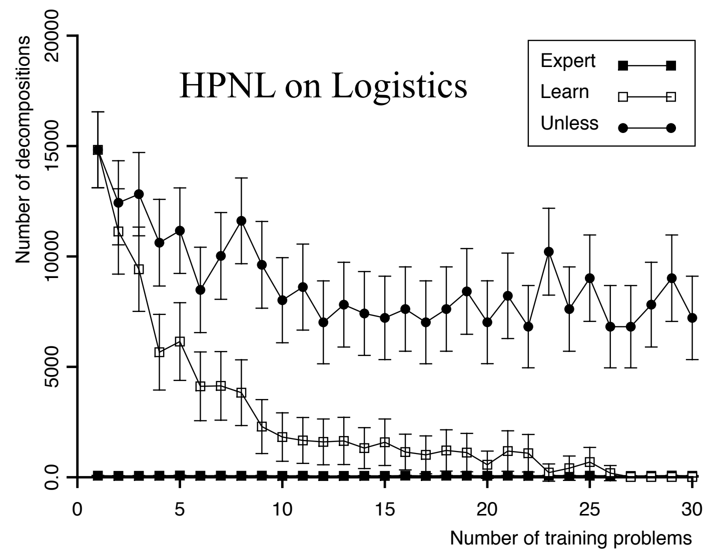
The system uses *domain constraints* to identify conditions on its methods, not classic EBL or ILP techniques.



A Sample Logistics Plan

HPN Methods for Logistics

```
((at ?o1 ?l3))
 conditions:    ((object ?o1) (truck ?t1) (location ?l3) (location ?l1)
                 (in-city ?l3 ?c1) (in-city ?l1 ?c1) (at ?t1 ?l3) (at ?o1 ?l1))
 operator:      (unload-truck ?o1 ?t1 ?l3)
 subproblem:    ((at ?t1 ?l3) (in ?o1 ?t1))
((at ?t1 ?l1))
 conditions:    ((truck ?t1) (location ?l3) (location ?l1)
                 (city ?c1) (in-city ?l3 ?c1) (in-city ?l1 ?c1)
                 (in-city ?l2 ?c1) (at ?t1 ?l3))
 operator:      (drive-truck ?t1 ?l3 ?l1 ?c1)
 subproblem:    ((at ?t1 ?l3))
 unless-goals:  ((in ?o ?t1)))
((in ?o1 ?t1))
 conditions:    ((object ?o1) (truck ?t1) (location ?l1) (location ?l3)
                 (in-city ?l1 ?c1) (in-city ?l3 ?c1) (at ?t1 ?l3) (at ?o1 ?l1))
 operator:      (load-truck ?o1 ?t1 ?l1)
 subproblem:    ((at ?t1 ?l1) (at ?o1 ?l1))
((in ?o1 ?t1)
 :conditions    ((object ?o1) (truck ?t1) (location ?l1) (airport ?l1)
                 (location ?l2) (location ?l3) (in-city ?l1 ?c1) (in-city ?l2 ?c1)
                 (in-city ?l3 ?c2) (at ?t1 ?l2) (at ?o1 ?l3))
 :operator      (load-truck ?o1 ?t1 ?l1)
 :subproblem    ((at ?t1 ?l1) (at ?o1 ?l1))
```



Analogical Concept Learning

McClure et al.'s (2015) SAGE can acquire complex concept descriptions from labeled training cases by:

- Representing each concept as a set of relational literals with associated probabilities
- For each new training case T:
  - Using structural analogy to retrieve descriptions similar to T and selecting the best candidate C
  - If C and T match well enough, then using T to update C's probabilities and to add new relations
  - Else storing a new disjunctive description based on case T

SAGE learns geographical concepts, musical genres, and object shapes far more rapidly than statistical methods.

Meta-Interpretive Learning

Muggleton et al. (2018) report a new *abductive* approach to learning relational logic programs that:

- Uses domain-independent knowledge stated as logical rules
- Searches for simple explanations that cover each case
- Posits new predicates that may be reused in the explanation
- Transforms the explanations into domain rules for later use

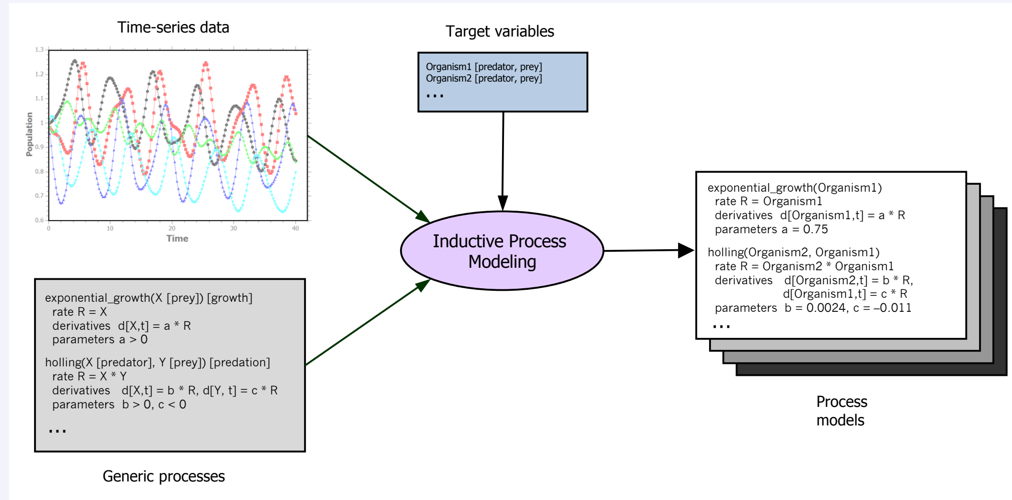
Meta-interpretive learning (MIL) masters visual concepts and control programs very rapidly, often from single cases.

*This work demonstrates that **representation learning** is not limited to deep neural networks.*

Inductive Process Modeling

Inductive process modeling (Langley, 2019) constructs explanations of time series from background knowledge.

Discovered models comprise sets of *differential equations* organized into higher-level *processes*.

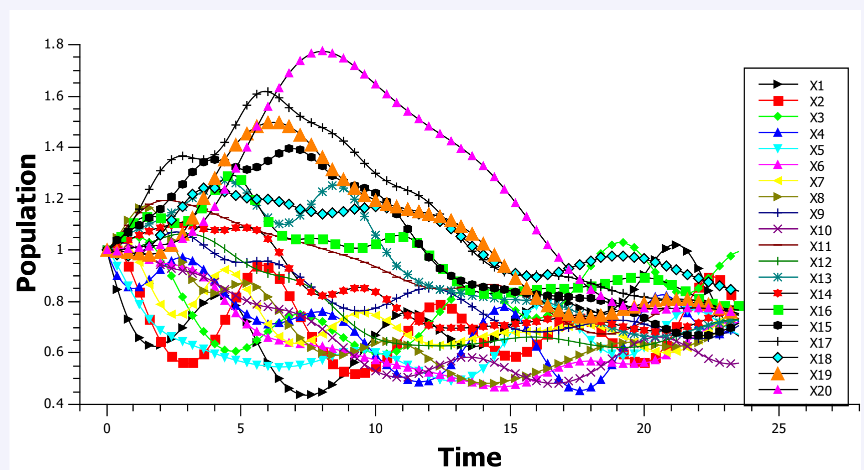


A Quantitative Process Model

(a) organism_loss[phyto, detritus]
  rate $r = phyto$
  equations $d[phyto, t] = -0.05 * r$
  $d[detritus, t] = 0.05 * r$
nutrient_uptake[phyto, nitro]
  rate $r = phyto * nitro$
  equations $d[phyto, t] = 0.5 * r$
  $d[nitro, t] = -0.005 * r$
remineralization[detritus, nitro]
  rate $r = detritus$
  equations $d[detritus, t] = -0.04 * r$
  $d[nitro, t] = 0.04 * r$

(b) $d[phyto, t] = -0.05 * phyto + 0.5 * phyto * nitro$
$d[nitro, t] = -0.005 * phyto * nitro + 0.04 * detritus$
$d[detritus, t] = 0.05 * phyto + -0.04 * detritus$

Trajectories for 20-Organism Food Chain



Comparison of Characteristics

We can compare how these systems – and the most popular class of statistical learners – fare on the computational gauntlet.

Characteristics	Deep Net	Cobweb	Prodigy	SAGE	MIL
Modular structures	○	●	●	●	●
Composable elements	⊙	○	●	○	●
Piecemeal learning	○	●	●	●	●
Incremental processing	○	●	●	●	○
Knowledge guidance	⊙	●	⊙	●	⊙
Rapid acquisition	○	●	●	●	●

The table shows that Cobweb, Prodigy, SAGE, and meta-interpretive learning all pass most of its challenges.

Can Neural Nets Learn Like Humans?

Some may believe neural networks cannot exhibit human-like learning, but there are counterexamples:

- Neural networks can support *transfer* of expertise from earlier training to produce rapid learning on related tasks.
- *Cascade correlation* (Fahlman & Lebiere, 1990) learns network structure in a piecemeal way, adding one node at a time.
- *Adaptive Resonance Theory* (Grossberg, 1987) is incremental and piecemeal, adding nodes when none match well enough.

The latter two combine the creation of new structures with statistical updates, much as Cobweb, Prodigy, and SAGE.

These results suggest the issue lies not with neural networks, but with how most developers *instantiate* them.

Fostering Work on Human-Like Learning

Research on human-like learning was once widely accepted by the AI community. How can we restore this vision?

- *Broaden education to cover classic methods*
- *Expand funding to support human-like approaches*
- *Establish publication venues that value such work*
- *Champion evaluation with* ***computational gauntlets***

Together, these steps can help create a *Zeitgeist* that recaptures the spirit of early AI and machine learning.

This call to arms echoes similar appeals by Fahlman (2012), Marcus and Davis (2021), and others.

Making the Gauntlet Operational

Before we can use the computational gauntlet for evaluation of learning systems, we must:

- Specify a dependent measure for each hurdle
  - *Some qualitative but others a matter of degree*
- Provide training sets that allow cumulative learning
  - *To demonstrate ability to benefit from knowledge*
- Encourage reporting of learning curves
  - *To show rates of improvement and asymptotes*

We can then compare these to the characteristics of human learning in chosen target domains.

Summary Remarks

Machine learning, despite impressive advances, has abandoned many of its early, profound insights.

A promising alternative is to develop AI systems that learn in a more human-like manner by:

- *Acquiring modular, composable structures in a piecemeal, incremental way, aided by knowledge, from little data.*

We can treat these features as design constraints that define a *computational gauntlet* for learning systems.

I call on audacious AI researchers to tackle this challenge.

The Computational Gauntlet