AI trends 2025-2035: Neuro-Symbolic Architectures, World Models, and Grounding

Introduction – A Review of ULOGIC/UMIND –

In the previous articles of this series on ULOGIC/UMIND, we have journeyed through the foundations of logic and the philosophy of mathematics, the problem of paradoxes, and the search for solutions.

The main conclusions have been threefold:

1. Current logic has ended up being "a mathematical theory about polynomials within mathematical structures," abandoning its original mission of being a "reverse engineering of language". This reverse engineering was meant to understand the rules of reasoning, demonstration, and computation. Logic must change course and be understood again as a reverse engineering of language: making the rules explicit (which in turn transforms the language into a new one).

2. It is essential to understand that there is a SINGLE language, which we use to speak, do science, compute, and perform metalinguistics (we cannot step outside of it). It is a language we have invented over the last 3000 years, a unique language with MULTIPLE capabilities:
    - **Discursive capability** (talking about the world)
    - **Weak-argumentative capability** (imitative reasoning)
    - **Strong-argumentative capability** (logic and mathematics)
    - **Computational capability** (defining and executing procedures)
    - **Metalinguistic capability** (talking about itself, self-describing its rules).

3. The contribution of ULOGIC is a first step to demonstrate that IT IS POSSIBLE to define an exact, rule-governed language that has the "complete" capabilities of our "fuzzy and imprecise" language. In ULOGIC, reasoning and computation are the same: procedures for manipulating expressions. Everything is an expression (the execution of an algorithm is also an expression). All results would be contained exactly in a worldwide network of TekDocs (a type of document that is also an expression). It is also possible to endow ULOGIC with self-metalinguistic capability. Additionally, it allows for a new foundation for mathematics (in particular, a sufficient solution to the problems of set theory, inspired by a radical revision of what definitions are and how they work).

ULOGIC is inspired by and designed to solve problems in logic, mathematics, and the philosophy of science. It is a piece of "basic science" initially inspired by questions as far removed from engineering as the problem of classical set-theoretic paradoxes.

Compared to a formal language like FOL, CIC, Coq, or Lean, ULOGIC is like comparing a bicycle

to an airplane: the construction principles are completely different (in fact, radically opposed), and the result is an infinitely superior expressive power.

The crucial (and unexpected) issue is that all this would have a practical application: a consensus is beginning to emerge in the AI field that "scaling-data-and-power" approaches (the path of LLMs) are entirely insufficient by design. Modular architectures with more ingredients are needed, such as exact reasoning capabilities, world representations and causality, and perceptual interconnection for "grounding" (anchoring symbols with experience).

In this new landscape, **Neuro-Symbolic architectures** (connecting intuitive systems like LLMs with exact reasoning systems) are considered essential on the path forward (a path that will likely require more).

Our proposal (in previous articles) is that using ULOGIC, a language with full reasoning, computation, and metalinguistic capabilities, is not merely an incremental improvement over using FOL, Coq, Lean, or other languages used in Neuro-Symbolic architectures. Instead, it is the seed for intuitive systems (LLMs) to **develop the intuition for correct reasoning**, in addition to obtaining verifiable and reliable products.

This would solve (by design) the problem of **reliability-verifiability-explainability** that is unattainable for an LLM: Reliable because the Ulogic-Kernel has verified the correctness of the rule-based results (the intuitive system proposes, but the Kernel decides). Explainability is immediate: the products themselves are their justification (a step-by-step proof of something is the explanation of a result, and the same applies to algorithms and their outcomes).

The confrontation of an LLM with a Ulogic-Kernel based on a language with sufficient expressive power would allow the LLM to develop **the intuition for reasoning**, computation, and metalinguistics, and for its internal abstract vector representations to be modeled and to reflect (in an abstract way) the explicit external rules of the language.

This duality of capabilities fits perfectly with what philosophers and mathematicians have always described: Mathematics may be a language with verification rules, **but the "thinking" of a human mathematician is something chaotic, inspired by similes, visions, and intuitive dreams.** A mathematician does not achieve results by following the rules, but by not following them. A human mathematician "sees the results" first without needing to prove them (although this intuition is usually wrong, and that's why the hard task of proving the intuition correct remains; that's the game). It is impossible, even for a machine, to explore the combinatorial infinity of applicable rules in the game of mathematics: A machine will only be able to "discover mathematics" if it develops an intuitive engine to propose results, proposals that will then be exactly verified by the Kernel (another component of the system).

There is another insoluble problem for LLMs: "**composability**," the ability for knowledge results to be reused to generate more knowledge, and also to accelerate the acquisition and understanding of new problems. In ULOGIC/UMIND, the products of activity are collected in **TekDocs**, which would constitute a worldwide network of exact, verified knowledge. This solves the "composability" problem by design.

This leads us to a plausible assumption: Could we use TekDocs as a support for a **representation of common-sense knowledge**? This would require an extension of ULOGIC to a

v2 to add fuzzy concepts of quantity, probability, approximation, plausibility, temporality, modality, etc. But this is a feasible project because, at its core, it is a huge structure of interrelated concepts (that's what mathematical structures are). This is expressible in ULOGIC. It doesn't differ much from expressing "a group theory," only with tens of thousands of interrelated concepts. The "generation bottleneck" of this network of concepts and their interrelations would still need to be solved, but a "translator" is conceivable that can read "natural language" books and express them in "extended-ULOGIC-language," incorporating these interrelations between concepts.

The problems of "grounding" would be solved as usually proposed with "perceptual systems," but with a particular approach: that these systems create representations of the world in abstract spaces "with topology" and **use the unified ULOGIC language** to dynamically label regions of that space. That would be talking about the world.

The problem of planning and goal setting is still a very speculative upper layer at this time, but if ULOGIC is capable of expressing not only algorithms but also "general procedures," **then it is capable of expressing "plans"** (because a plan is a general procedure). Would we need to go further and have it set goals? Perhaps only intermediate goals for the achievement of general goals that we set for it.

But the main question is, what do we want to achieve with ULOGIC/UMIND? Do we want a "human mind"? The problem of creating a mind with human capabilities is very simple; biology solved it, anyone can do it, and it takes 9 months (it usually takes two people to achieve it). In short: we already have human minds.

What we believe our goal should be is to **build "artificial-scientific-minds" that help us in reasoning, problem-solving, and scientific research**, as assistants but also as "**reliable and explainable co-discoverers**." That would be the goal of ULOGIC/UMIND.

Next steps and questions:

Inevitably, the question arises as to whether this path could be a "possible solution towards AGI" (artificial general intelligence). It seems plausible that a powerful neurolinguistic component, founded on a language of superior expressiveness like ULOGIC, is the initial step from which to build everything else. There is at least one strong indication: it meticulously imitates the way scientists and mathematicians "think," with an intuitive component as a creative engine feeding the exact-Kernel, the arbiter of the game.

But what is the current recent landscape on the possibility and paths to explore for building an AGI according to recognized authors in the field? Having a general overview is necessary to at least try to understand "the size of the elephant."

What follows is a report compiled from various sources, sufficiently compact and brief, but at the same time with some extension, that allows visualizing the trends in artificial intelligence for the years 2025-2035 according to the leading figures and actors currently in this field.

Part I: The AGI Problem Space and the Limits of Scale

Section 1: Formalizing Artificial General Intelligence

The quest for Artificial General Intelligence (AGI) represents the original and most ambitious goal of the AI field: the creation of systems with human-level intelligence, capable of understanding, learning, and adapting to a wide range of intellectual tasks.

1.1. Defining Intelligence: From Task Optimization to General Competence

There are two perspectives. One, eminently mathematical and formal, defines intelligence in terms of performance and optimization. The other, with a more cognitive inspiration, defines it in terms of architecture and underlying capabilities.

The **universal intelligence perspective**, proposed by Shane Legg and Marcus Hutter, formally defines intelligence as the ability of an agent to achieve goals in a wide range of environments. This definition is based on the theoretical framework of AIXI, a Bayesian reinforcement learning agent that is, in theory, optimally intelligent. This approach is valuable for its formalism and generality, as it is not anthropocentric and ranges from simple agents to superintelligences. However, its nature is purely behavioral and external; it measures performance without specifying the internal architecture. The search for approximations to this universal intelligence naturally leads to the scaling hypothesis: if intelligence is optimal performance, then scaling data, parameters, and computation should better approximate that optimum.

In contrast, the **cognitive systems perspective**, advocated by researchers like Ben Goertzel, defines AGI based on its human-like cognitive capabilities. According to Goertzel, an AGI must possess "a reasonable degree of self-understanding and autonomous self-control, and have the ability to solve a variety of complex problems in a variety of contexts, and to learn to solve new problems that[1] it was not aware of at the time of its creation." This definition focuses not on external performance, but on internal architecture and cognitive flexibility. It's not just about *what* the system does, but *how* it does it. This view implies that intelligence requires specific modules for self-representation, planning, and learning, which leads directly to the cognitive architecture hypothesis.

1.2. A Pragmatic Enumeration of the Fundamental Capabilities of AGI

Synthesizing research in the field, we can establish a set of necessary, though perhaps not sufficient, capabilities that any system aspiring to be an AGI must demonstrate. These capabilities will serve as evaluation criteria for the architectures discussed in this report.

- **Reasoning and Strategy:** The ability to employ logic, strategy, solve puzzles, and make decisions under uncertainty. This involves going beyond simple pattern matching to perform deductive, inductive, and abductive inferences.
- **Knowledge Representation:** The ability to build, maintain, and use a rich and coherent internal model of the world, including common-sense knowledge. This model must represent entities, their properties, their relationships, and the rules that govern their interactions.

- **Planning and Abstraction:** The ability to set and pursue goals, break down complex problems into manageable sub-problems, and operate at multiple levels of abstraction. An AGI must be able to formulate long-term plans and adjust them dynamically.
- **Learning and Adaptation:** The ability to learn efficiently and continuously from experience with minimal human intervention. This includes transfer learning, few-shot learning, and self-teaching, allowing the system to adapt to novel situations and domains for which it was not explicitly programmed.
- **Grounded Communication:** The ability to use natural language in a way that is meaningfully connected to the real world and the system's internal representations. Language should not be a mere game of symbols, but a tool for describing and reasoning about the world.
- **Embodied Interaction:** Although not a universally accepted requirement, the ability to perceive the world through multiple sensory modalities (vision, hearing, touch) and to act on the physical environment is increasingly seen as a crucial catalyst for the development of truly general intelligence. Embodied cognition posits that intelligence develops through interaction with the physical world.

The debate between the universal intelligence and cognitive systems definitions is not merely academic; it defines research strategies. The former suggests that AGI might "emerge" from scale, while the latter argues it must be "designed" with a specific architecture. The accumulated evidence on the limitations of large-scale models strongly supports the second view, making cognitive architecture the most viable path toward AGI.

Section 2: The Scaling Hypothesis and Its Fundamental Flaws

The current era of AI (2016-2025) is dominated by the scaling hypothesis—the belief that general intelligence can emerge simply by increasing the size of models, the amount of training data, and computational power. Large Language Models (LLMs) are the ultimate expression of this hypothesis. Despite their impressive capabilities in language generation and pattern recognition, which position them as sophisticated "System 1 Thinking" systems—fast, intuitive, and habit-based—their limitations are not mere engineering problems to be solved with more scale, but fundamental architectural flaws that prevent their progression toward AGI. An analysis of these limitations reveals that they are not isolated failures, but a cascade of symptoms stemming from a root cause: the absence of a structured, causal, and reality-grounded world model (coupled with an inability for real, rigorous reasoning).

2.1. Central Limitation 1: The Symbol Grounding Problem

The symbol grounding problem, formulated by Stevan Harnad, asks how symbols within a formal system can acquire intrinsic meaning, rather than being merely "parasitic" on the meanings that exist in our minds. An LLM operates on a system of ungrounded symbols; the words and tokens it manipulates have no inherent connection to the objects, properties, or relationships in the real world. They are part of a "symbol-to-symbol carousel," where the meaning of one symbol is defined solely in terms of other symbols.

This lack of grounding is the most profound deficiency of LLMs. It means that, at a fundamental level, they do not "understand" the concepts they process. Their operations are purely syntactic,

based on statistical relationships learned from a massive text corpus, without access to the real-world semantics that the text describes.

NOTE: The preceding three paragraphs are a "description in common language" of the problem. However, in my opinion, the understanding of "meaning" is more complex and has at least three levels:

- **Weak-intensional meaning:** The meaning that words have in relation to other words, within a system with "discursive-narrative" capability. If you know how to "string words together," you have "understood the weak-intensional meaning."
- **Strong-intensional meaning:** The meaning that words have in relation to other words and to the exact production rules of the system (which allows one to see logical relationships beyond purely narrative ones). If, in addition to stringing words together, you have logical coherence, you have "understood the strong-intensional meaning." (Impossible if you don't know that logical rules exist).
- **Denotational meaning:** The meaning that words have because the agent using them has perceptual systems that connect perceptions (internal states) with language. Impossible if there are no perceptual systems.

It is incorrect to say that LLMs do not understand meaning. Of course they do, but they understand weak-intensional meaning, lacking strong-intensional meaning and, of course, denotational meaning (which is non-existent without perceptual systems).

2.2. Central Limitation 2: Absence of Coherent World Models

As a direct consequence of the lack of grounding, LLMs do not build or maintain explicit, dynamic, and interpretable world models. A world model is an internal representation of entities, their states, and the rules governing their interactions. LLMs, instead, construct a massive statistical map of token co-occurrences.

Chess serves as a paradigmatic example of this failure. An LLM can recite the rules of chess, describe famous openings, and even predict the most likely next move in a known game, because all this information exists as text in its training data. However, if presented with a novel board configuration or asked about the implications of a non-standard chess rule, its performance degrades catastrophically. It cannot "reason" about the board because it does not possess an internal, manipulable, and coherent representation of it; it can only retrieve textual patterns associated with chess.

Although some research suggests that LLMs may develop "internal world models" for specific tasks, these are implicit, fragile, and not available for the deliberate, general-purpose reasoning that AGI requires. One cannot point to a data structure within an LLM and say, "this is where the state of the chessboard is stored."

2.3. Central Limitation 3: Failure of Causal and Logical Reasoning

The absence of a world model that separates entities from their underlying causal mechanisms makes robust causal reasoning impossible for LLMs. Using Judea Pearl's "Ladder of Causation"

as a framework, LLMs operate almost exclusively on the first rung: Association. They are experts at identifying correlations in text (e.g., "smoke is associated with fire"), but they systematically fail on the higher rungs:

- **Rung 2 (Intervention):** They cannot reliably predict the outcomes of an action or intervention (e.g., "what would happen if I prevent the fire?").
- **Rung 3 (Counterfactuals):** They cannot reason about hypothetical scenarios that contradict the facts (e.g., "if there had been no fire, would there be smoke?").

This inability manifests in several ways. LLMs confuse correlation with causation, are heavily influenced by the temporal order of events in the text (assuming that what is mentioned first is the cause), and their performance degrades when a narrative contradicts the parametric knowledge stored in their weights. This leads to logical inconsistencies and an inability to perform robust multi-step reasoning, an indispensable requirement for general intelligence.

2.4. Central Limitation 4: Systematic Generalization and Compositionality

Human intelligence is characterized by **systematic compositionality**: the ability to understand and produce an infinite number of novel expressions by combining a finite set of known elements (words, concepts) according to rules. This ability is the basis of robust generalization. LLMs lack this skill. Their "knowledge" is not a compositional grammar of concepts, but a flat statistical map. They learn to recognize high-level combinations that are frequent in the training data, but struggle to generalize to novel combinations that are semantically valid but statistically rare. For example, a model that has seen "man chases dog" and "woman rides horse" may not be able to correctly interpret "woman chases horse." Its generalization is therefore fragile and unreliable outside the distribution of its training data.

2.5. Central Limitation 5: Lack of Reliability and Practical Barriers

These deep architectural flaws manifest in a series of practical problems that make LLMs, in their current form, unfit to be the core of an AGI.

- **Hallucinations:** The generation of plausible but factually incorrect information is a direct consequence of the lack of grounding and world models. Without a mechanism to verify facts against a coherent model of reality, the model simply generates the most probable sequence of tokens, which may or may not correspond to the truth.
- **Catastrophic Forgetting:** The architecture of LLMs makes them inherently susceptible to forgetting previously learned information when they are fine-tuned on new tasks. A true AGI agent must be able to learn continuously and incrementally without degrading existing knowledge, a capability that LLMs do not possess.
- **Data and Compute Inefficiency:** The very "scaling laws" that have driven the success of LLMs also point to an unsustainable path. The need for astronomical amounts of data and ever-increasing computational power for marginal improvements suggests that this approach faces diminishing returns and physical and economic barriers.

In summary, the failures of LLMs are not isolated errors that can be patched individually with more data or ad-hoc solutions like Retrieval-Augmented Generation (RAG). They are

interconnected symptoms of a central architectural disease. The lack of grounding prevents the construction of world models; the absence of world models prevents causal reasoning; and the lack of a causal and compositional model prevents systematic generalization.

This unified diagnosis points to an inescapable conclusion: the path to AGI is not through scaling LLMs, but through a fundamental paradigm shift towards architectures that explicitly design systems for **grounding**, **world modeling**, and **reasoning**.

Fundamental Limitation	Description	Root Architectural Cause	Key Researchers/Papers
Lack of Symbol Grounding	Symbols (tokens) lack intrinsic meaning; their "understanding" is based on statistical relationships with other symbols, not with the real world.	The model is trained exclusively on textual data, without perceptual or action-based connection to an environment.	Harnad (1990), Bender & Koller (2020)
Absence of World Models	Inability to build and maintain coherent, dynamic, and manipulable internal representations of entities, states, and their rules.	Consequence of the lack of grounding. Without meaningful symbols, a semantic model of the world cannot be built, only a statistical map of language.	Gary Marcus, Yann LeCun
Failure in Causal Reasoning	Confuses correlation with causation. Cannot reason about interventions or counterfactuals. Performance degrades with non-chronological narratives.	Absence of a world model that separates entities from causal mechanisms. Operates at the level of association, not intervention.	Judea Pearl, Kıcıman et al. (2023)
Non-Systematic Generalization	Difficulty generalizing to novel combinations of known concepts (lack of compositionality). Fragile generalization outside the training distribution.	Knowledge is not structured compositionally, but as a flat statistical map of patterns.	Lake & Baroni (2023), Fodor & Pylyshyn (1988)
Hallucinations	Generation of plausible but factually incorrect or inconsistent information.	Direct consequence of the lack of grounding and a world model against which to verify the truthfulness of generated statements.	Lin et al. (2022), Weidinger et al. (2022)
Catastrophic Forgetting	Fine-tuning on new tasks degrades or destroys previously learned knowledge.	Updating the model's weights for a new task interferes with the distributed representations of old knowledge.	Kirkpatrick et al. (2017), Li & Hoiem (2017)

Part II: Fundamental Paradigms for Cognitive Architectures

The critique of the scaling hypothesis forces us to seek constructive alternatives. Several of the most influential researchers in AI have proposed blueprints for cognitive architectures that directly address the shortcomings of monolithic models. These visions, though diverse in their implementation details, converge on a set of three fundamental principles: **modularity**, the centrality of **world models**, and the need to integrate learning with **reasoning**.

Section 3: Blueprints for Cognitive Architectures

3.1. Yann LeCun: The World Model as a Central Component

Yann LeCun posits that the path to autonomous intelligence lies not in traditional reinforcement learning, which is inefficient and requires a massive number of trials, but in **model-based predictive learning**. An intelligent agent must be able to predict the consequences of its actions and those of others, allowing it to reason and plan efficiently. His proposal is a modular cognitive architecture, where each component is differentiable and therefore trainable using gradient-based methods.

LeCun's proposed architecture consists of several interconnected modules:

- **Perception Module:** This module receives raw sensory inputs (images, audio, text) and transforms them into abstract representations of the current state of the world.
- **World Model:** This is the heart of the system. It receives the current state representation from the perception module and a sequence of hypothetical actions from the Actor module. Its function is to predict future states of the world. This model must be able to handle the inherent uncertainty of the real world by predicting multiple plausible futures.
- **Cost Module:** This is the agent's intrinsic motivation engine. It evaluates the "discomfort" or cost associated with a world state. It consists of two sub-modules:
    - **Intrinsic Cost:** An immutable, hardwired cost representing basic drives (e.g., avoiding harm).
    - **Critic:** A trainable module that learns to predict the future cumulative cost from a given state, enabling longer-term planning.
- **Actor Module:** Its function is to generate sequences of actions that minimize the cost predicted by the World Model and the Cost Module. It performs a search or optimization in the space of action sequences to find the most suitable one.
- **Short-Term Memory:** Maintains and updates the state of the world, both perceived and predicted, serving as a workspace for reasoning.

The key technology LeCun proposes to implement the World Model is the **Joint Embedding Predictive Architecture (JEPA)**. Unlike generative models that try to predict every detail of the input (e.g., every pixel of a future image), which is computationally expensive and unnecessary, JEPA operates in an abstract representation space. It encodes an input (e.g., a video frame) into

an abstract representation and then predicts the abstract representation of a future input (e.g., the next frame). By predicting in this latent space, the model can ignore irrelevant and hard-to-predict details, focusing on the essential semantics of the world, making it much more efficient and robust for learning world models.

3.2. Yoshua Bengio: System 2 Cognition and the Consciousness Prior

Yoshua Bengio approaches the AGI problem through the lens of cognitive psychology, using Daniel Kahneman's distinction between **System 1** and **System 2** thinking.

- **System 1:** Is fast, intuitive, unconscious, and parallel. It corresponds to the current capabilities of deep learning, such as pattern recognition.
- **System 2:** Is slow, logical, sequential, and conscious. It involves reasoning, planning, and manipulating abstract concepts.

Bengio argues that achieving AGI requires developing a robust System 2. The centerpiece of Bengio's proposal is the **Consciousness Prior**. This is not an architectural component but a powerful inductive bias for learning representations. The hypothesis is inspired by the Global Workspace Theory of consciousness, which posits that consciousness acts as a bottleneck: from a vast set of unconscious processes, a few elements are selected by attention and "broadcast" globally to condition subsequent processing.

Formally, the Consciousness Prior posits that "conscious thoughts" are low-dimensional combinations of a few high-level concepts. A sentence like "the cat is on the mat" involves few concepts (cat, on, mat) but makes a strong and likely true statement about the world. This implies that the joint probability distribution over high-level concepts has the structure of a **sparse factor graph**. In this graph, each factor (representing a dependency) connects only a small number of variables (concepts), but the dependency can be very strong. This structure is inherently compositional and allows for combinatorial generalization, addressing one of the main weaknesses of LLMs.

To implement the sampling and search mechanisms necessary for System 2 reasoning, Bengio and his team have developed **Generative Flow Networks (GFlowNets)**. GFlowNets are a class of generative models designed to sample compositional objects (like graphs, equations, or plans) with a probability proportional to a reward or energy function. Unlike MCMC methods that can be slow to mix, GFlowNets learn a policy to construct these objects sequentially, making them well-suited for structured search tasks in the vast combinatorial spaces that characterize System 2 reasoning.

3.3. Geoffrey Hinton: Alternative Learning and Representation Paradigms

Although Geoffrey Hinton has not proposed a complete, unified cognitive architecture like LeCun or Bengio, his recent work offers crucial alternative components that could be fundamental to any future AGI.

- **The Forward-Forward (FF) Algorithm:** Hinton has criticized the biological plausibility of backpropagation and has proposed the FF algorithm as an alternative. FF replaces the forward and backward passes of backpropagation with two forward passes: one with

"positive" (real) data and another with "negative" (generated or incorrect) data. Each layer of the network has its own local objective: to maximize a "goodness metric" (e.g., the sum of squared neural activities) for positive data and minimize it for negative data. By normalizing activity between layers, each layer is forced to learn new features. This local, backpropagation-free learning could be a much more efficient and biologically plausible way to train the deep neural networks that make up the perception and action modules in an AGI architecture.

- **Capsule Networks:** Convolutional Neural Networks (CNNs) achieve translation invariance through pooling, which discards precise position information. Capsule Networks are an alternative designed to preserve spatial information and handle hierarchical part-whole relationships more robustly. A "capsule" is a group of neurons whose activity vector represents the instantiation parameters of an entity (such as its position, orientation, size, etc.). Higher-level capsules make predictions for the poses of lower-level capsules, and are activated if multiple predictions agree, a process called "routing by agreement." This architecture has built-in compositionality and equivariance (the ability to understand how transformations on an object affect its representation), which could lead to much more robust and generalizable perception modules than those based on standard CNNs.

3.4. Ben Goertzel: An Integrative and Open-Source Framework for AGI

Ben Goertzel and the OpenCog project offer a pragmatic and constructive vision of AGI, materialized in the open-source framework **OpenCog Hyperon**. The core philosophy is "cognitive synergy": the idea that general intelligence will not arise from a single master algorithm, but from the cooperative and dynamic interaction of multiple AI algorithms and paradigms.
Hyperon's architecture is based on two key components:

- **Atomspace:** This is the system's universal knowledge store. Technically, it is a weighted, labeled metagraph, an extremely flexible data structure capable of representing knowledge in various forms: declarative (facts), procedural (programs), linguistic, attentional, goal-oriented, etc. Unlike a traditional database, the Atomspace is designed for dynamic manipulation and transformation by AI algorithms.
- **MeTTa (Meta-Type Talk):** This is the system's programming language. MeTTa is a functional, probabilistic, and strongly-typed language, designed specifically so that various AI algorithms (neural networks, logical reasoners, evolutionary algorithms) can operate on the Atomspace and interact with each other. It allows one algorithm to manipulate or even rewrite another, facilitating the emergence of complex cognitive processes and the self-organization of the system.

Goertzel's vision is less a fixed architecture and more an ecosystem for intelligence to emerge. It provides a common language and workspace where different specialists (a neural image recognizer, a logical theorem prover) can collaborate to solve problems that none could solve alone.

SUMMARY:

Analyzing these four visions, an implicit consensus emerges. Despite differences in terminology

and proposed implementation mechanisms (JEPA, Consciousness Prior, Atomspace), all these pioneers converge on the need for a modular, cognitively-inspired architecture. They reject the idea that AGI will emerge from a monolithic black box trained end-to-end.

Instead, they propose systems composed of specialized, interacting modules: a module for perception, another for reasoning and planning, and a motivation system. The central question of AGI research is no longer "scale vs. architecture," but "which architecture?"

The next step is to detail how the reasoning module—the component that endows these systems with "thought"—can be built.

Section 4: Neuro-Symbolic Integration: The Synthesis of Learning and Reasoning

If modular cognitive architectures are the "what" of the path to AGI, neuro-symbolic integration (NeSy) is the "how."

NeSy is the technological paradigm that provides the crucial bridge between the continuous, sub-symbolic, data-driven world of neural networks and the discrete, structured, logic-based world of reasoning and planning.

It is the technology that allows for the construction of the "System 2" modules or "reasoning engines" that the architectures of LeCun, Bengio, and Goertzel demand.

4.1. Fundamental Principles of Neuro-Symbolic AI (NeSy)

Neuro-symbolic AI arises from the recognition that the two historical paradigms of AI—connectionism (neural networks) and symbolism (logic, rules)—are complementary in their strengths and weaknesses.

- **Neural networks** excel at learning patterns from raw, unstructured, and noisy data, but they are opaque ("black boxes"), require vast amounts of data, and lack explicit, verifiable reasoning capabilities.
- **Symbolic AI** excels at explicit reasoning, planning, and explainability, as it operates on formal rules and knowledge. However, it is brittle in the face of noisy or incomplete data and suffers from the "knowledge acquisition bottleneck," as rules often must be manually encoded by experts.

NeSy seeks to combine the best of both worlds to create systems that are simultaneously robust in learning and rigorous in reasoning, more transparent, data-efficient, and capable of complex, multi-step inference.

4.2. A Taxonomy of NeSy Architectures

To understand the various ways in which neural and symbolic components can be integrated, the taxonomy proposed by Henry Kautz is an influential and widely adopted framework. This

taxonomy classifies NeSy architectures based on the flow of control and the nature of their integration.

1. **Symbolic[Neuro]:** In this architecture, the main control is symbolic. A symbolic reasoning system (like a search engine or a planner) invokes a neural network as a subroutine to perform a specific task that is difficult to formalize with rules, such as pattern evaluation.
    - **Interaction Mechanism:** The symbolic component passes a query to the neural component and receives an output (e.g., a score, a classification) which it then uses in its reasoning process.
    - **Canonical Example:** *AlphaGo* and *AlphaGeometry*. In AlphaGo, a Monte Carlo Tree Search (MCTS) algorithm (the symbolic component) explores the game tree. To evaluate the quality of a board position, it calls a neural network (the neural component) that returns an estimate of the win probability from that position. In AlphaGeometry, a symbolic deduction engine, when stuck, asks a language model (the neural component) to suggest promising auxiliary geometric constructions to advance the proof.

2. **Neuro|Symbolic:** This is a pipeline or sequential architecture. A neural component acts as a perception front-end, processing raw data (e.g., an image) and extracting a structured symbolic representation. This representation is then passed to a symbolic back-end component that performs reasoning.
    - **Interaction Mechanism:** The neural network translates the unstructured input into a symbolic format (e.g., a scene graph, a logical formula). The symbolic reasoner operates exclusively on this symbolic output.
    - **Canonical Example:** *Visual Question Answering (VQA)*. A VQA system might use a CNN to detect objects in an image ("cat," "sofa") and their spatial relationships ("is on"). This information is encoded into a scene graph. A logical reasoner can then answer a question like "What color is the object the cat is on?" by querying this graph.

3. **Neuro:Symbolic → Neuro:** In this approach, symbolic knowledge is used to guide or structure the training process of a neural network. The symbolic component is not part of the system at inference time.
    - **Interaction Mechanism:** Symbolic knowledge is "compiled" into the network's architecture or training data. For example, a symbolic algebra system can be used to generate millions of problem-solution pairs to train a neural network to solve equations. Or, logical rules can be used to generate additional data labels or to impose constraints on the network's architecture.

4. **NeuroSymbolic (Deep Integration):** This is one of the tightest forms of integration. Symbolic knowledge, typically in the form of first-order logic, is integrated directly into the neural network's learning process, often as a regularization term in the loss function.
    - **Interaction Mechanism:** Logical rules are translated into a differentiable format. During training, the loss function measures not only the error on labeled data but also the degree to which the network's predictions violate the logical rules. This

forces the network to learn representations (embeddings) that are consistent with the symbolic knowledge.

- **Canonical Example:** *Logic Tensor Networks (LTN)*. In LTN, an axiom like $\forall x(\text{cat}(x) \rightarrow \text{mammal}(x))$ is converted into a loss that penalizes the model if the "truthfulness" of an object being a mammal is less than the "truthfulness" of it being a cat.

5. **Neuro[Symbolic]:** This architecture is the inverse of Symbolic[Neuro]. The main control is neural. A neural model, typically an RL agent or an LLM, learns to invoke an external symbolic reasoning engine as if it were a tool.
    - **Interaction Mechanism:** The neural model generates a query to an external system (a calculator, a search engine, a knowledge base) and then incorporates the response into its own generation or decision-making process.
    - **Canonical Example:** *Agentic LLMs* that learn to use tools. An LLM asked "What is the square root of the population of Paris?" might learn to generate a call to a search API to find the population and then a call to a calculator to perform the mathematical operation, rather than trying to guess the answer.

4.3. Key NeSy Frameworks and Implementations

Beyond the taxonomy, several concrete software frameworks have demonstrated the power of NeSy approaches in practice.

- **DeepProbLog:** This framework integrates probabilistic logic programming (ProbLog) with deep learning through the concept of a **neural predicate**. A neural predicate links a logical predicate to a neural network. For example, in the task of adding MNIST digits, one can define a predicate digit(Image, Number). The neural network takes an Image and produces a probability distribution over the possible Numbers. This probabilistic fact can then be used within a larger logic program. The classic example is addition(I1, I2, Sum) :- digit(I1, N1), digit(I2, N2), Sum is N1 + N2. Here, two neural networks (one for each digit) perform perception, and the logic program performs symbolic reasoning (the addition). The entire system can be trained end-to-end, where the error in the final sum is backpropagated to improve both the digit classification networks and the probabilistic parameters of the logic program.

- **Logic Tensor Networks (LTN):** LTN provides a language, called **Real Logic**, to ground first-order logic in continuous vector spaces, making it compatible with deep learning. In LTN, logical symbols are mapped to elements of the deep learning framework:
    - **Constants** become tensors (feature vectors).
    - **Predicates** become neural models (classifiers) that take input tensors and produce a truth value in the interval [0, 1], using fuzzy logic semantics.
    - **Logical axioms** are translated into a differentiable loss term. The training objective is to minimize this loss, which forces the model to learn embeddings for concepts (e.g., "swan," "white") that are consistent with the logical rules.

- **Neural Logic Machines (NLM):** While DeepProbLog and LTN often use predefined logical rules, NLMs are a neural architecture designed for the **inductive learning of "lifted" rules** (rules containing variables) from examples. NLMs operate on tensor representations of predicates and use neural operators that emulate logical operations (AND, OR, NOT) and quantifiers (EXISTS, FOR ALL). By being trained on tasks like sorting a small list, an NLM can learn the underlying symbolic rule of sorting and then generalize it to much longer lists, demonstrating a capacity for systematic generalization that purely neural models fail to achieve.

These frameworks demonstrate that neuro-symbolic integration is not just a theoretical idea.

It provides the software engineering tools and algorithmic foundations to build the reasoning modules of cognitive architectures. A perception module (like a JEPA) can produce abstract representations, which then become the input for a NeSy reasoning engine.

For example, the predicate digit(X, N1) in DeepProbLog is a perfect Neuro|Symbolic interface: the neural network performs perception (from the image to a probability distribution over digits), and this becomes a symbolic fact for the logic program. NeSy is, therefore, the necessary implementation detail for any viable modular AGI architecture.

NeSy Framework	Fundamental Principle	Kautz Category	Strengths	Limitations
DeepProbLog	**Neural Predicates:** Neural networks define the probabilities of facts in a probabilistic logic program.	Neuro|Symbolic	Natively handles uncertainty; integrates logical and probabilistic reasoning; end-to-end trainable.	-
Logic Tensor Networks (LTN)	**Differentiable Logic:** First-order logic axioms are converted into a loss function to regularize the learning of embeddings.	NeuroSymbolic	Imposes logical consistency on embedding spaces; improves data efficiency by using prior knowledge; flexible.	The interpretation of fuzzy truth values can be less intuitive; the choice of fuzzy operators affects performance.
Neural Logic Machines (NLM)	**Inductive Lifted Rule Learning:** Neural architecture designed to learn abstract logical rules (with variables) from input/output examples.	Neuro:Symbolic → Neuro (learns rules that could then be compiled)	Demonstrates strong systematic generalization to larger-scale problems; learns the logical structure rather than having it predefined.	Scalability to high-arity rules and complex logics is a challenge; extracting human-readable rules from weights is an open problem.
AlphaGeometry	**Neural Network-Guided Symbolic Search:** A symbolic deduction engine uses an LLM to	Symbolic[Neuro]	Solves extremely complex reasoning problems that are beyond the reach of purely neural or	Highly specialized architecture; requires a domain-specific symbolic engine.

NeSy Framework	Fundamental Principle	Kautz Category	Strengths	Limitations
	propose promising auxiliary constructions.		symbolic methods.	
LLM Agents with Tools	**LLM Tool Invocation:** An LLM learns to generate calls to external tools (calculators, APIs, databases) to obtain accurate information.	Neuro[Symbolic]	Reduces hallucinations by grounding answers in verifiable data sources; combines language flexibility with the precision of symbolic systems.	Depends on the LLM's ability to learn to use tools correctly; API call overhead can be an issue.

Part III: A Blueprint for a Hybrid AGI System

The analysis of the limitations of current models and the exploration of alternative paradigms converge on a clear conclusion: the path to AGI requires a deliberate design based on cognitive principles. It will not be a monolithic system, but a hybrid, modular architecture. This final part synthesizes the previous findings to propose a concrete blueprint for such a system, highlighting its fundamental pillars and the challenges that remain to be solved.

Section 5: The Essential Pillars of a Generalizable Intelligence

Any architecture aspiring to AGI must be built on three conceptual pillars that directly address the fundamental shortcomings of current models. These pillars are not optional components, but prerequisites for robust and adaptable intelligence.

5.1. The Grounding Imperative

The symbol grounding problem is perhaps the most fundamental obstacle for current AI. For the symbols that a system manipulates (be they words, tokens, or nodes in a graph) to have genuine meaning, they must be causally connected to the real world. A system trained solely on text, no matter how vast the corpus, remains trapped in the "symbol-to-symbol carousel," unable to truly understand the referents of its language.

The solution to this problem is **multimodal grounding**. An AGI must learn from and about the world through multiple sensory modalities—vision, hearing, touch, etc.—in addition to language. Vision provides information about objects, their spatial properties, and their interactions; sound informs about events and their dynamics; language provides a symbolic structure for describing and reasoning about these perceptions. It is the correlation and coherence between these modalities that allow symbols to be anchored in perceptual experience. A system that learns that the word "cat," the image of a cat, and the sound "meow" consistently co-occur begins to build a multimodal, grounded representation of the concept of a *cat*, overcoming the superficiality of LLMs.

5.2. The Centrality of World Models

Once a system has grounded symbols, it can begin to build a **world model**. This is not simply a static store of facts, but an internal, dynamic, and simulatable representation of the environment and itself. A robust world model is essential for high-level cognitive capabilities:

- **Prediction:** It allows the agent to anticipate future states of the world as a consequence of events or its own actions. This is fundamental to any form of planning.
- **Planning:** By using the world model as an internal simulator, the agent can "imagine" the outcomes of different action sequences without having to execute them in the real world, enabling efficient and safe decision-making.
- **Information Filling:** The model can infer missing or unobservable information. If the agent sees smoke, its world model can infer the high probability of a hidden fire.

- **Creativity and Counterfactuals:** A world model allows the agent to reason about hypothetical scenarios, a cornerstone of creativity and advanced problem-solving.

LeCun's JEPA architecture is explicitly designed to learn these world models efficiently by predicting in an abstract representation space rather than in the high-dimensional sensory space.

5.3. The Primacy of Causal Reasoning

For a world model to be truly useful for a general intelligence, it must be **causal**, not merely correlational. As Judea Pearl argues, true intelligence lies in the ability to climb the "Ladder of Causation." An agent must be able not only to observe associations (Rung 1) but also to reason about the effects of interventions (Rung 2) and to imagine counterfactual outcomes (Rung 3). A causal world model explicitly represents cause-and-effect relationships, allowing the agent to answer questions like "*Why* did this happen?" and "What should I do to make that happen?".

This capability is a prerequisite for robust planning, fault diagnosis, and strategic decision-making. **Neuro-Symbolic Causal AI** is the emerging field dedicated to building these models. In these architectures, causal relationships are often represented in a symbolic structure (like a causal knowledge graph or a logic program), while neural components are used to learn the parameters of these causal models from observational and interventional data, or to perform inferences over them.

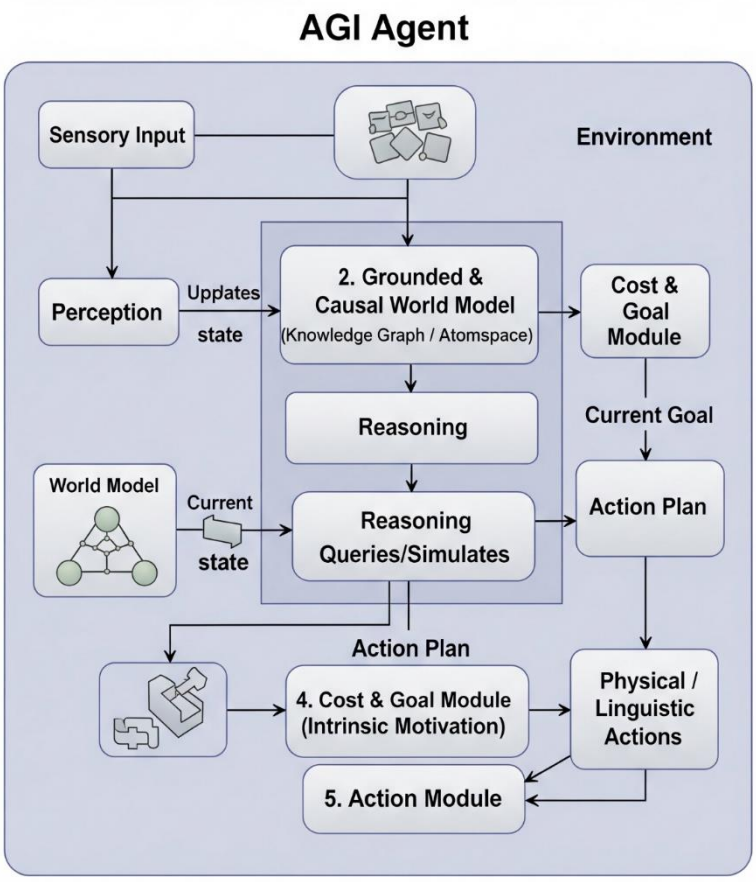
These three pillars—grounding, world models, and causality—are not independent. Multimodal grounding provides the raw data to learn a world model, and this world model must have a causal structure to enable truly intelligent reasoning and planning. Together, they form the foundation upon which a modular AGI architecture must be built.

Section 6: A Proposed Modular Architecture for AGI

Synthesizing the principles of AI pioneers and the enabling technologies of NeSy, a tentative BLUEPRINT for a modular AGI architecture can be proposed. This architecture is not a monolithic model, but a system of specialized components interacting in a continuous cognitive cycle.

Diagram 1: The AGI Cognitive Cycle

The following diagram illustrates the high-level architecture, showing the main modules and their interactions. This design would be a (possibly approximate) synthesis of the proposals by LeCun, Bengio, and Goertzel:



Description of Modules:

1. **Multimodal Perception (System 1):** This is the agent's portal to the world. It ingests streams of raw, unstructured sensory data. Its function is to transform this data into abstract, structured, and disentangled representations. To do this, it would use self-supervised learning architectures like LeCun's **JEPA** to learn predictive representations, or Hinton's **Capsule Networks** to robustly capture part-whole hierarchies. The output of this module is not labels, but high-level vector representations of the detected objects, events, and their properties.
2. **Grounded and Causal World Model:** This is the agent's cognitive core, its "understanding" base. It is a dynamic, structured knowledge base that stores the state of the world. The most likely implementation would be a **Causal Knowledge Graph** or a metagraph like OpenCog's **Atomspace**. The abstract representations from the Perception module are used to continuously update this model. It stores not only what entities exist and what their properties are (grounding), but also the causal rules that govern their interactions (causality).
3. **NeSy Reasoning and Planning Engine (System 2):** This is the deliberation module. It

receives a goal from the Cost Module and the current state from the World Model. Its task is to formulate a plan to achieve the goal. It is intrinsically a hybrid neuro-symbolic system that combines:

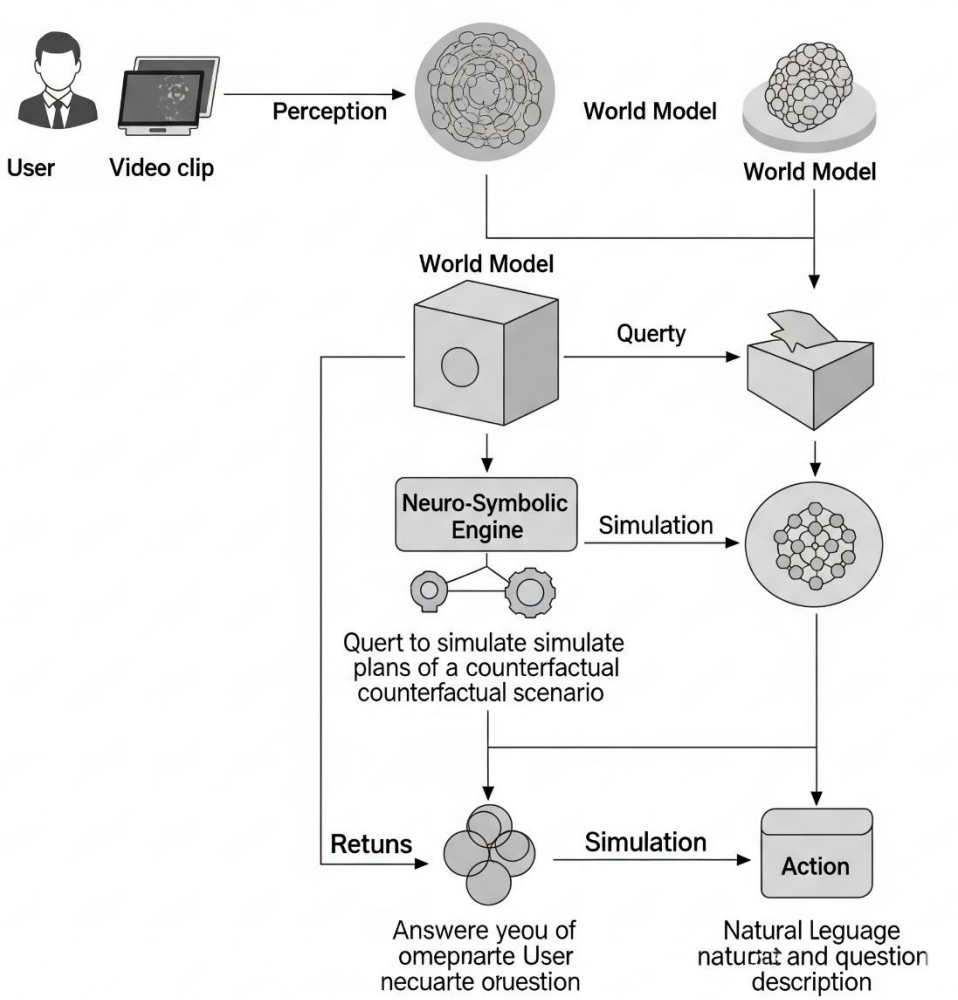
- **Symbolic Reasoning:** It uses a logical inference engine (like a theorem prover or a symbolic planner) to generate high-level plans and ensure logical correctness.
- **Neural Policies:** A neural network, trained via reinforcement learning, guides the search through the vast space of possible plans, pruning unpromising branches, analogous to AlphaGo's policy network.
- **World Model Simulation**: Before executing a plan, this engine "simulates" it in the World Model to predict its consequences, allowing for error correction and optimization before acting.This module would implement multiple NeSy strategies, such as Neuro[Symbolic] to query specialized tools (e.g., a mathematical solver) or Neuro|Symbolic to reason about perceived states.

4. **Cost and Goal Module:** This is the agent's intrinsic motivation system, as described by LeCun. It does not rely on external rewards, but on internal goals to guide behavior. These goals can be low-level (e.g., "avoid harm") or high-level (e.g., "minimize uncertainty about the world," which creates a curiosity drive, or "answer the user's question"). It is this module that initiates the reasoning process by providing a goal to the planning engine.

5. **Action Module:** This is the agent's effector. It translates the abstract plans generated by the Reasoning Engine into concrete actions in the environment. These actions can be motor control for a robot or language generation for a conversational agent. The execution of these actions modifies the environment, which in turn generates new sensory data, thus closing the perception-action loop.

Diagram 2: Information Flow for Grounded Causal Reasoning

To make it concrete how this architecture would operate, let's consider the information flow for answering a counterfactual question—a task that is beyond the reach of current LLMs.



Query: A user shows the agent a video of a red billiard ball hitting a blue ball, and asks: *"What would have happened if the red ball had been heavier?"*

Flow Steps:

1. **Input and Perception:** The Perception Module processes both the video and the text. It identifies the entities ("red ball," "blue ball"), their properties (color, motion), and the events (collision).
2. **Grounding and World Model Update:** The perceived information is used to instantiate and update the Causal World Model. The model now contains grounded representations of the balls and the physical (causal) rules of collisions.
3. **Goal Formulation:** The Cost/Goal Module interprets the question as a request for

counterfactual inference, setting the goal to "answer the user's question."

4. **Reasoning and Simulation:** The NeSy Reasoning Engine receives this goal. It creates a temporary copy of the world state. In this copy, it performs an **intervention**, modifying the *mass* property of the *redBall* entity. It then uses the causal rules of the World Model to simulate forward and predict the new outcome of the collision.
5. **Response Generation:** The result of the simulation (e.g., a new velocity vector for the blue ball) is passed to the Action Module, which translates it back into natural language to answer the user.

A crucial capability that distinguishes a true AGI from a static system is **meta-learning** or self-organization.

The described architecture should not be fixed. A truly general agent must learn to orchestrate its own cognitive resources.

Different problems require different reasoning strategies. A simple question might be solved directly by the perception module (System 1), while a complex planning problem would require extensive deliberation by the reasoning engine (System 2).

The AGI should learn to dynamically allocate its computational resources and select the most appropriate reasoning strategy for the task at hand. This implies a higher level of control, perhaps a "Configurator" as proposed by LeCun, which optimizes not only the agent's knowledge about the world but also its own problem-solving processes. This is a step towards an intelligence that not only learns, but learns to learn.

Section 7: Conclusion: Open Problems and Future Trajectories

7.1. Recapitulation of the Argument

This report has argued that Artificial General Intelligence is not a likely destination on the path of scaling monolithic models like LLMs. The limitations of these systems—their lack of grounding, the absence of coherent world models, their failures in causal reasoning, and their inability for systematic generalization—are not superficial defects, but fundamental architectural flaws.

The most promising alternative, supported by an emerging consensus among leading researchers in the field, is the deliberate design of **modular cognitive architectures**. These architectures are inspired by the principles of human and animal cognition, separating perception from reasoning and centering intelligence on an internal, dynamic, and causal world model. The key technology for realizing the reasoning components of these architectures is **Neuro-Symbolic AI**, which merges the robust learning of neural networks with the rigor and explainability of symbolic reasoning.

The blueprint proposed in this report—a cognitive cycle that integrates multimodal perception, a causal world model, a neuro-symbolic reasoning engine, an intrinsic motivation system, and an action module—represents a synthesis of these ideas. It is a path towards an AI that not only

recognizes patterns, but understands, reasons, and plans.

7.2. Key Unsolved Challenges

Despite the clarity of this trajectory, the path to AGI is fraught with fundamental research challenges that must be addressed.

- **Scalable Causal Learning:** How can an agent efficiently and robustly learn causal models of the world from observation and interaction, especially in complex, high-dimensional environments? Current methods for causal discovery often do not scale well or require strong assumptions.
- **The Representation Problem:** What is the optimal format for the world model? How can we most effectively bridge the gap between the continuous, distributed representations of neural networks and the discrete, compositional structures of symbolic logic? Although frameworks like LTN and DeepProbLog offer solutions, integration remains an active area of research.
- **The Objective Function Problem:** How do we define cost functions or intrinsic objectives that lead to safe, aligned, and beneficial behavior without being overly restrictive? A truly autonomous agent must be able to develop its own sub-goals, but this raises profound questions of safety and alignment.
- **Efficient and Scalable NeSy Integration:** Although many NeSy frameworks exist, their integration can be computationally expensive and complex. The development of smoother, more efficient hybrid systems that do not introduce computational bottlenecks is critical for the practical viability of these architectures.

7.3. Final Remarks

The journey towards Artificial General Intelligence is not a sprint based on scaling a single paradigm, but a complex, interdisciplinary research program.

It requires the synthesis of ideas from deep learning, symbolic AI, cognitive science, causality, and philosophy. Hybrid, modular, and cognitively-inspired architectures, powered by neuro-symbolic integration, represent the most promising frontier of this endeavor.

Building machines that not only calculate, but think, remains the greatest challenge of our time, and it is in this synthesis of learning and reasoning that the most well-founded hope of achieving it lies.

7.4. Conclusion: The Rosetta Stone of AI?

From all the above, it follows that Neuro-Symbolic architectures will be at the center of all attempts to create AGI, or at least AI systems better than the current ones, in the foreseeable future of 2025-2035.

It seems that the first step, in a complex and long sequence, will be to integrate the intuition of LLMs with formal languages and logic, in order to achieve powerful and sufficient reasoning

capabilities.

If this is the case, then "logic," reinvented and redefined as "reverse engineering of the universal language we use for discourse, reasoning, and computation," will be a fundamental and central field of research to find the solution.

And in that case, a language like ULOGIC (and its future improvements) could be the Rosetta Stone that allows the foundations of the entire edifice to be laid.

In the words of Archimedes of Syracuse: δῶς μοι πᾶ στῶ καὶ τὰν γᾶν κινάσω ("give me a place to stand and I will move the world").

REFERENCES

1. Cognitive Architectures and World Models (LeCun's Vision)

- **Fuente:** LeCun, Y. (2022). *A Path Towards Autonomous Machine Intelligence*. OpenReview.
    - **URL:** https://openreview.net/pdf?id=BZ5a1r-kVsf
    - **Comentario:** This is the foundational paper by Yann LeCun that outlines his vision for a modular, cognitive architecture for AGI, centered on a predictive world model. It details the roles of the Perception, World Model, Cost, and Actor modules, arguing that this is the path to move beyond the limitations of current models. It's the primary source for the architecture proposed in Section 3.1.
- **Fuente:** Assran, M., et al. (2023). *Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture*. arXiv.
    - **URL:** https://arxiv.org/pdf/2301.08243
    - **Comentario:** This paper provides the technical details for the Joint-Embedding Predictive Architecture (JEPA), which LeCun proposes as the key technology for building the World Model component. It explains how JEPA works in an abstract representation space to make predictions, which is more efficient than predicting every pixel.

2. System 2 Cognition and Reasoning (Bengio's Vision)

- **Fuente:** Bengio, Y., et al. (2019). *The Consciousness Prior*. arXiv.
    - **URL:** https://arxiv.org/pdf/1709.08568
    - **Comentario:** This is a key paper by Yoshua Bengio introducing the "Consciousness Prior". It formalizes the idea that high-level, conscious thoughts involve a sparse combination of concepts, which maps to a sparse factor graph.
- **Fuente:** Bengio, Y., et al. (2021). *GFlowNets for AI-Driven Scientific Discovery*. arXiv.
    - **URL:** https://arxiv.org/pdf/2307.13524
    - **Comentario:** This paper details Generative Flow Networks (GFlowNets), the mechanism proposed by Bengio's lab to perform structured search and sampling. It's the technical answer to How a System 2 module might explore the vast combinatorial space of reasoning. It's designed to generate compositional objects like graphs or explanations.

3. Limitations of LLMs: Causality and Grounding

- **Fuente:** Kıcıman, E., et al. (2023). *Causal Reasoning and Large Language Models: A Survey*. arXiv.

    - **URL:** https://arxiv.org/pdf/2305.00050

    - **Comentario:** This is a comprehensive survey about the causal reasoning failures of LLMs. It uses Judea Pearl's "Ladder of Causation" as a framework to show that LLMs are stuck at the level of association and fail at intervention and counterfactuals

- **Fuente:** Bender, E. M., & Koller, A. (2020). *Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data*. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

    - **URL:** https://aclanthology.org/2020.acl-main.463.pdf

    - **Comentario:** This is a highly influential paper that provides a rigorous linguistic and philosophical argument for the symbol grounding problem in LLMs. It argues that models trained only on text (form) cannot, by definition, learn meaning in the way humans do. This paper is a cornerstone for the arguments you make about the "symbol-to-symbol carousel" and the different levels of meaning.

- **Fuente:** Lake, B. M., & Baroni, M. (2023). *Human-like systematic generalization through a meta-learning neural network*. Nature.

    - **URL:** https://www.nature.com/articles/s41586-023-06668-3

    - **Comentario:** This paper directly addresses the lack of systematic generalization and compositionality in standard neural networks. It presents an architecture that learns to generalize in a more human-like way, confirming that this is a recognized and critical limitation of current approaches and a major area of active research.

4. Neuro-Symbolic Architectures (NeSy) and Frameworks

- **Fuente:** d'Avila Garcez, A. S., et al. (2022). *Neurosymbolic AI: The 3rd Wave*.

    - **URL:** https://www.scitepress.org/Papers/2022/107775/107775.pdf

    - **Comentario:** A great high-level overview of the field of Neurosymbolic AI. It positions NeSy as the "Third Wave" of AI, following handcrafted rules and statistical learning. It validates the central thesis that combining neural and symbolic approaches is the most promising path forward.

- **Fuente:** Kautz, H. (2022). *The Third AI Summer, and its Contradictions*. AI Magazine.

    - **URL:** https://ojs.aaai.org/index.php/aimagazine/article/view/19485/19253

    - **Comentario:** This is the source for the influential taxonomy of Neuro-Symbolic architectures. It provides the definitions for Symbolic[Neuro], Neuro|Symbolic, etc., and serves as a formal framework for classifying the different integration strategies discussed.

- **Fuente:** De Raedt, L., et al. (2020). *From Statistical Relational AI to Neuro-Symbolic AI*.

    - **URL:** https://arxiv.org/pdf/2003.08316

- **Comentario:** This paper provides a detailed look into the principles behind systems like DeepProbLog. It explains the concept of the "neural predicate" and how probabilistic logic programming can be integrated with deep learning.

5. Integrative AGI Frameworks (Goertzel's Vision)

- **Fuente:** Goertzel, B., et al. (2023). *The OpenCog Hyperon AGI Architecture*. arXiv.
    - **URL:** https://arxiv.org/pdf/2310.18318
    - **Comentario:** This is the primary technical paper describing the OpenCog Hyperon framework. It details the two core components: the Atomspace as a flexible knowledge metagraph and the MeTTa programming language designed for cognitive synergy. It is the definitive source for the ideas presented in Section 3.4, advocating for an integrative, multi-algorithm approach to AGI.