STRATEGIC RESEARCH REPORT

Category-Theoretic Artificial Intelligence

A structured research report on categorical machine learning, deep learning architectures, probabilistic inference, reinforcement learning, semantics, and AI safety.

Prepared by
Yu Murakami
CEO, New York General Group

Date
June 1, 2026

Table of Contents

Executive Summary

Category-theoretic artificial intelligence is best understood as an emerging systems discipline for artificial intelligence. Its objective is not merely to attach abstract mathematics to machine learning, but to provide a rigorous language for composition, structure preservation, feedback, uncertainty, semantics, and verification across increasingly complex AI systems.

The field remains early. It is not yet a dominant industrial method for building foundation models. However, it is becoming increasingly relevant as AI systems evolve from single neural networks into heterogeneous assemblies of models, tools, retrieval systems, memory stores, agents, policies, evaluators, and human governance loops.

7

analytical research streams used as a working taxonomy, not a market count

3

near-term value pools assessed qualitatively: architecture, inference, safety

2024+

period in which categorical deep learning became a more explicit research program

The central thesis of this report is that category theory can become a strategic language for engineering trustworthy AI systems. It helps articulate how parts fit together, how constraints are preserved, how feedback loops operate, and how local components create global behavior.

This report deliberately avoids symbolic formulas. It presents the subject in executive language while preserving technical specificity. The intended audience includes AI leaders, research strategists, technical founders, applied mathematicians, and executives evaluating long-term AI infrastructure opportunities.

Evidence note. The numerical groupings in this report are analytical classifications used for executive orientation. They are not market-size estimates, adoption statistics, or empirical consensus counts. Claims about data efficiency, robustness, verification, and safety are stated as hypotheses or early signals unless a cited prototype or paper is explicitly identified.

Five executive takeaways

AI is becoming compositional.	The critical unit of analysis is increasingly the whole system, not the isolated model.
Category theory is the mathematics of composition.	It offers a disciplined way to represent interfaces, transformations, constraints, and system behavior.
Categorical deep learning is a young but important program.	It seeks a unified theory of architectures such as recurrent, graph, equivariant, and state-based models.
The most plausible first applications are not general LLM training.	Near-term potential lies in structured models, scientific AI, robotics, probabilistic workflows, and safety engineering.
Tooling is the bottleneck.	The field will need practical specification languages, compilers, validators, and integrations with PyTorch and JAX.

1. Strategic Context

Why category theory is entering the AI conversation now

Modern AI is moving from model-centric development toward system-centric development. A production AI application may include a foundation model, retrieval layer, tool orchestration layer, memory store, policy engine, monitoring system, human approval pathway, and feedback loop. The challenge is no longer limited to making one model accurate. It is to make a heterogeneous system coherent, safe, auditable, and adaptable.

Category theory is relevant because its core concern is composition. It asks what kinds of objects exist, what transformations connect them, how transformations compose, and which properties are preserved when systems are assembled. This makes it a natural candidate for reasoning about complex AI systems.

The strategic problem: an architecture zoo without a common grammar

Deep learning has produced a large and powerful collection of architectures: convolutional networks, recurrent networks, graph neural networks, transformers, diffusion models, normalizing flows, equivariant networks, world models, reinforcement learning agents, probabilistic programs, and neural-symbolic hybrids. Each has its own vocabulary. The result is a fragmented theory of design.

Category-theoretic AI attempts to provide a common grammar. It does not reduce every model to the same thing. Instead, it identifies structural relationships among models: how data moves, how representations transform, how constraints are preserved, how learning updates operate, and how components can be composed without losing meaning.

Why the timing matters

Several forces make the topic newly important. First, AI systems are becoming modular. Second, agents increasingly act in external environments. Third, safety expectations are rising. Fourth, scientific AI requires respect for symmetry, causality, and physical constraints. Fifth, enterprise AI needs auditable workflows rather than opaque model calls. These trends all reward a more formal account of interfaces and composition.

The field is still early, but the direction is strategically meaningful: a future AI stack may include not only training frameworks and inference engines, but also specification languages, categorical compilers, diagrammatic validators, and architecture generators that translate high-level structural intent into verified model families.

2. Conceptual Foundations

A non-symbolic explanation of the categorical vocabulary used in AI

This chapter introduces the core concepts of category theory without formulas. The goal is to explain why these concepts matter for artificial intelligence, not to reproduce a graduate mathematics course.

Categories: systems of things and transformations

A category consists of things and transformations between things. In AI, the things may be data types, state spaces, feature spaces, tasks, environments, representations, or model classes. The transformations may be functions, learned models, probabilistic transitions, policies, data preprocessing steps, embedding procedures, or inference routines.

The important point is not the vocabulary itself. The important point is that transformations can be composed. A raw document can be cleaned, embedded, searched, summarized, evaluated, and stored. Category theory gives a disciplined way to describe this chain and ask whether the composition preserves the properties we care about.

Functors: structure-preserving translations

A functor translates one structured world into another while preserving the relevant pattern of composition. In AI, this is a powerful idea. A grammatical structure can be translated into a semantic vector representation. A graph structure can be translated into feature representations. A data transformation can be translated into a corresponding transformation of model outputs.

This matters because many AI problems are not just about mapping inputs to outputs. They are about preserving structure across representations. If the structure is lost, the model may perform well on benchmarks while failing under meaningful transformations.

Natural transformations: disciplined consistency across models

Natural transformations express consistency between two structure-preserving translations. In AI terms, they can represent a principled correspondence between two representation schemes, two model families, or two pipelines that should behave consistently under changes in input structure.

Naturalness is closely related to a family of ideas that AI researchers already care about: invariance, equivariance, calibration, consistency, and robustness. Category theory packages these ideas in a common language.

Monoidal categories: parallel composition

Many AI systems combine multiple inputs or multiple components. A multimodal model combines text, image, audio, and structured data. A multi-agent system combines several interacting policies. A retrieval-augmented system combines a language model with a search subsystem and a database. Monoidal categories provide an abstract language for parallel composition.

Monads: structured computational effects

A monad is a way to organize computations that carry effects. In programming, effects may include state, uncertainty, failure, non-determinism, or interaction with the external world. In AI, these effects are everywhere: probabilistic inference, memory updates, tool calls, environment interaction, search, sampling, and policy execution.

This is particularly relevant for AI agents. A language model call may be wrapped inside a stateful process that reads memory, calls tools, handles errors, revises plans, and updates beliefs. Monadic perspectives can help describe such behavior without pretending it is a simple deterministic function.

Lenses and optics: bidirectional processes

Neural learning is bidirectional. A model computes forward from input to output, then training signals move backward to update parameters. Lenses and optics are categorical tools for representing this kind of bidirectional information flow.

This has direct relevance to backpropagation, optimization, reinforcement learning, and feedback control. A lens-based description can show how local components participate in global learning, and how updates compose when models are assembled from smaller parts.

Coalgebras: behavior, observation, and state

Coalgebras are useful for systems whose identity is expressed through behavior over time. Examples include automata, recurrent networks, state machines, reinforcement learning environments, interactive agents, and streaming systems. They are especially relevant when the question is not only what a system computes, but how it unfolds, reacts, and evolves.

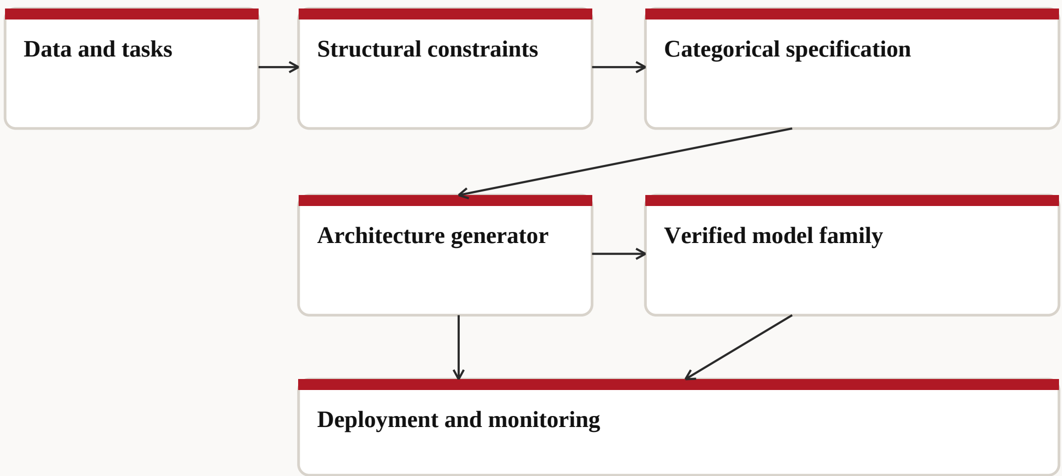


Exhibit: A categorical AI workflow converts structural intent into implementation and governance artifacts.

Source: New York General Group synthesis based on applied category theory literature.

Concrete workflow example: a retrieval-augmented AI system can be specified as a sequence of typed components: ingest documents, normalize text, create embeddings, retrieve candidate passages, generate an answer, check the answer against policy, and log the decision. A categorical specification asks whether each component accepts the right kind of input, whether the output of one component is valid for the next, whether uncertainty remains marked as uncertainty, and whether safety constraints survive the whole composition. The practical guarantee is not that the answer is automatically true; it is that invalid handoffs and constraint-breaking compositions become explicit design objects rather than hidden implementation accidents.

3. Research Landscape

The major streams of category-theoretic artificial intelligence

A field with multiple overlapping programs

Category-theoretic AI is not one single research agenda. It is an intersection of applied category theory, machine learning theory, programming language semantics, categorical probability, geometric deep learning, reinforcement learning, natural language semantics, and formal methods. The following research streams define the current landscape.

Gradient-based learning and backpropagation

This stream studies neural learning as a compositional process. It uses lenses, optics, reverse differentiation, and related structures to represent forward computation and backward update flow.

Categorical deep learning

This stream seeks a general theory of neural architectures. It asks how recurrent networks, graph networks, equivariant networks, automata, and other architectures can be understood as instances of a broader algebraic design language.

Categorical probability and Markov categories

This stream formalizes probabilistic computation, Bayesian inference, statistical experiments, conditioning, and causal reasoning through categorical structures.

Equivariance and structured generalization

This stream extends the idea of group-equivariant neural networks to more general categorical structures such as graphs, orders, groupoids, lattices, and sheaves.

Categorical cybernetics and reinforcement learning

This stream treats learning agents as bidirectional systems that interact with environments through feedback, rewards, policies, and value updates.

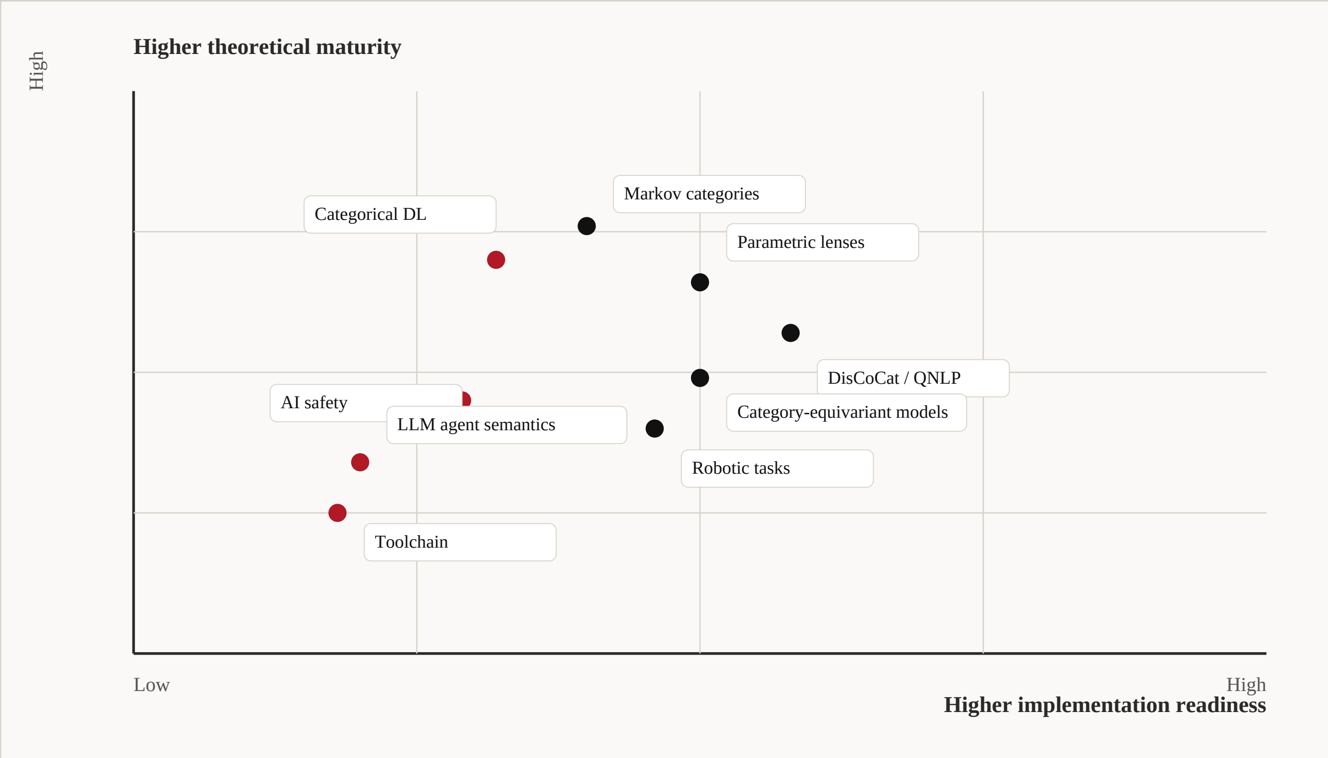
Categorical compositional semantics

This stream connects grammar, meaning, tensor structure, and distributional semantics, with DisCoCat as the leading example.

AI safety, systems engineering, and compositional governance

This emerging stream asks how formal composition can help verify the behavior of complex AI systems, agents, tools, and safety filters.

Exhibit 1: maturity of selected categorical AI areas



Interpretation: the map is qualitative. It compares theoretical maturity and implementation readiness, not commercial adoption.

4. Categorical Deep Learning

Toward a unified theory of neural architectures

Categorical deep learning is one of the most ambitious parts of the field. Its premise is that neural architectures should not be treated as isolated engineering recipes. Instead, they should be described by abstract specifications that generate families of valid implementations.

The architecture problem

Modern deep learning relies on architectures designed around structural assumptions. Convolutional networks assume local spatial structure. Graph neural networks assume relational structure. Recurrent networks assume temporal state. Transformers assume contextual interaction across token positions. Equivariant networks assume symmetry. Yet the field lacks a universal language for expressing these assumptions at the architecture level.

Categorical deep learning addresses this problem by treating architectures as structured mathematical objects. A model family is not just a set of layers. It is a system of allowable transformations, compositions, parameter sharing rules, and constraints.

Specification versus implementation

A central distinction is the difference between specification and implementation. A specification may say that a model should preserve a symmetry, respect a graph structure, maintain causality, or compose across tasks. An implementation says how to realize this in tensors, layers, parameters, and computation graphs.

Many failures in AI design happen because this connection is informal. The model is intended to have a structural property, but the implementation only approximately respects it, or respects it in one part of the pipeline and not another. Categorical deep learning aims to close that gap.

What would success look like?

In a mature version of this field, an engineer could state high-level constraints and receive a family of architectures that satisfy those constraints by construction. The system would expose which transformations are preserved, which components can be substituted, and which global properties follow from local design choices.

This would transform architecture search from a mostly empirical search over candidate networks into a specification-driven design process. Performance would still matter, but the search space would be shaped by mathematical structure.

5. Learning as Bidirectional Structure

Lenses, optics, and categorical accounts of optimization

Backpropagation is often introduced as a computational trick for efficiently calculating gradients. From a categorical perspective, it is also a compositional bidirectional process. Values move forward. Training signals move backward. Components compose in both directions.

Why lenses matter

A lens can represent a component with a forward behavior and a backward update behavior. In neural networks, a layer produces an output during the forward pass and contributes update information during training. When layers compose, their update behaviors compose as well.

This is not only aesthetically appealing. It can help explain why backpropagation works compositionally, how different optimizers can be represented in one framework, and how learning can be generalized beyond continuous differentiable spaces.

Optimizers as structured processes

Optimizers such as stochastic gradient descent, AdaGrad, momentum methods, and ADAM are not merely formulas. They are stateful procedures that transform local error information into parameter updates. Lens-based and optic-based approaches can model these procedures as interchangeable components in a broader learning architecture.

The practical promise is modularity. If optimization methods are expressed in a common categorical interface, it becomes easier to compare them, compose them, generalize them, and reason about their behavior in non-standard learning systems.

Beyond standard neural networks

Some categorical treatments of learning extend beyond ordinary differentiable real-valued networks. They can also address Boolean circuits, polynomial circuits, discrete systems, and other computational structures. This matters because future AI may combine differentiable neural components with symbolic, logical, discrete, or programmatic elements.

6. Categorical Probability

Uncertainty, inference, and causality as compositional processes

Artificial intelligence is saturated with uncertainty. Models estimate probabilities, generate samples, infer latent variables, predict outcomes, and act under incomplete information. Category theory offers tools for representing probabilistic processes as composable transformations.

Markov categories

Markov categories provide an abstract language for probabilistic systems. They allow researchers to speak about stochastic transformations, statistical experiments, independence, conditioning, and information flow without committing too early to one concrete representation of probability.

This is particularly valuable for AI because different applications use different probability models. A single abstract framework can help compare Bayesian models, probabilistic programs, causal models, and learning systems that produce distributions rather than deterministic outputs.

Bayesian inference and conditioning

Bayesian inference updates beliefs when evidence arrives. Categorical probability studies how such updates can be represented structurally. The benefit is a clearer account of how evidence, latent variables, observations, and predictions compose in complex pipelines.

Causal reasoning

Causal AI requires distinguishing observation from intervention. Category-theoretic probability and diagrammatic reasoning can help represent the flow of information and the effect of interventions. This is important for safety-critical AI, where correlation-based prediction is often insufficient.

7. Equivariance and Structured Generalization

From group symmetry to category-level structure

A model generalizes better when it respects the structure of its data. In image models, translation structure matters. In molecules, rotations and permutations matter. In graphs, node order should often be irrelevant. In time series, causal order matters. In knowledge graphs, relation types matter.

From invariance to naturalness

Invariance means that output should remain stable under certain input transformations. Equivariance means that output should transform consistently when input transforms. Category theory generalizes these ideas through naturality: behavior should be consistent with the structural transformations of the domain.

Why group symmetry is not enough

Group-based equivariant learning is powerful but limited. Many data structures are not naturally groups. Hierarchies, partial orders, graphs, grammars, workflows, typed dependencies, causal diagrams, and sheaf structures often require a broader language. Categories can represent these structures more naturally.

Category-equivariant neural networks

Category-equivariant neural networks attempt to define neural architectures whose behavior is consistent with transformations in a category. This broadens the design space beyond classical group equivariance and can unify several structured architectures under one conceptual framework.

The potential value is significant but still case-dependent: improved data efficiency, out-of-distribution robustness, clearer inductive biases, and more interpretable structural behavior are plausible benefits when the categorical structure accurately matches the data-generating process. They should be treated as testable claims, not automatic consequences.

8. Reinforcement Learning and Cybernetics

Agents, feedback, and compositional interaction

Reinforcement learning is naturally interactive. An agent observes a state, chooses an action, receives feedback from the environment, and updates its policy or value estimates. This loop is not a simple one-way mapping. It is a feedback system.

Categorical cybernetics

Categorical cybernetics studies learning and control through compositional, bidirectional systems. It is well matched to reinforcement learning because RL algorithms involve a forward path of action and observation and a backward path of reward, value correction, and policy improvement.

Why this matters for AI agents

LLM-based agents, robotic agents, and autonomous decision systems increasingly interact with external environments. They call tools, update memory, revise plans, receive feedback, and adapt policies. A categorical account can help describe the interfaces among agent, environment, evaluator, and governance system.

Compositional task learning

Complex tasks can often be decomposed into reusable skills. A robot may approach, grasp, lift, move, and place an object. A software agent may search, extract, summarize, verify, and file a result. Category theory can represent these skills as composable processes, making it easier to reason about reuse and safe assembly.

9. Language, Meaning, and DisCoCat

Categorical semantics as a bridge between grammar and representation

Categorical compositional distributional semantics, often called DisCoCat, is one of the clearest historical examples of category theory applied to AI-adjacent language modeling. It connects grammar with vector-based meaning.

The core idea

Distributional semantics represents words through their usage patterns. Compositional semantics explains how the meaning of a sentence arises from the meanings of its parts and the grammar that combines them. DisCoCat uses category theory to connect these two traditions.

Relevance in the foundation model era

Large language models dominate practical natural language processing, but DisCoCat remains conceptually important. It offers an explicit account of compositional meaning, a diagrammatic representation of grammar-to-semantics flow, and connections to quantum natural language processing.

In the future, categorical semantics may be used not to replace large language models, but to interpret, constrain, evaluate, or hybridize them with structured reasoning systems.

10. AI Safety and Systems Engineering

The most strategically important long-term application

The most consequential application of category-theoretic AI may be AI safety and systems engineering. As AI systems become agentic and modular, failures may arise not from one defective component but from unsafe composition among individually reasonable components.

Compositional safety

A safety filter may be reliable by itself. A retrieval module may be reliable by itself. A tool-use planner may be reliable by itself. Yet the composed system may still fail if information crosses boundaries in unexpected ways or if a safe output from one component becomes unsafe when combined with another.

Category theory can help represent the interfaces, transformations, and constraints that govern such systems. It can ask whether a property proven for a component is preserved when the component is placed into a larger architecture.

Type discipline for AI pipelines

Type systems prevent nonsensical compositions in software. AI systems need an analogous discipline. A model output should not automatically be treated as verified knowledge. A probability estimate should not be treated as a fact. A human-readable answer should not be treated as an executable instruction without validation.

Categorical and type-theoretic approaches can help encode these distinctions at the system level.

Governance loops

AI governance is not just a policy document. It is a system of monitoring, approval, escalation, evaluation, logging, auditing, and remediation. These are feedback loops. Categorical cybernetics and compositional systems theory can help model them as part of the AI architecture rather than as external afterthoughts.

11. Implementation and Tooling

What must exist before the field can scale

The main barrier to practical adoption is tooling. Most machine learning engineers do not want to manipulate abstract categorical definitions directly. They need languages, compilers, visual tools, validators, and integrations with existing machine learning frameworks.

Required toolchain capabilities

- A specification language for structural constraints, model interfaces, and allowable compositions.
- A compiler that converts categorical specifications into PyTorch, JAX, or another execution backend.
- A validator that checks whether architecture transformations preserve declared constraints.
- A diagrammatic interface for inspecting AI pipelines, feedback loops, probabilistic dependencies, and safety boundaries.
- Libraries for lens-based learning, category-equivariant model design, and categorical probability workflows.
- Benchmark suites that demonstrate performance, robustness, maintainability, and safety benefits.

Why implementation is hard

The difficulty is not only software engineering. It is conceptual alignment. Different categorical frameworks often describe related phenomena in different ways. The field needs pragmatic standards that preserve mathematical rigor while remaining usable by engineers.

Likely path to adoption

The most likely adoption path is not immediate replacement of existing deep learning frameworks. Instead, categorical tooling will probably appear as a higher-level design and verification layer above established tensor frameworks. It may first be adopted in domains where structure matters more than raw benchmark scaling: scientific AI, robotics, probabilistic programming, safety-critical workflows, and complex enterprise AI systems.

12. Bear Case and Risk Assessment

Why selective funding is more appropriate than broad deployment

The strategic case for category-theoretic AI is credible, but it is not risk-free. A balanced investment view must explain why decades of categorical and structural ideas have not yet become mainstream industrial machine learning practice.

The performance gap risk

The strongest bear case is empirical. Modern AI is driven by scale, data, compute, engineering iteration, and benchmark performance. Category-theoretic approaches often improve conceptual clarity before they improve measured accuracy. For many near-term product teams, a conventional transformer, graph neural network, probabilistic program, or rule-based guardrail may be cheaper and more effective than a categorical redesign.

The abstraction cost risk

Category theory compresses many ideas into elegant abstractions, but that compression creates a translation cost. Engineers need to know what to implement, how to test it, and how it improves outcomes. If categorical language becomes detached from measurable engineering workflows, it risks becoming a notation layer rather than an adoption driver.

The tooling and talent risk

The community is small relative to mainstream deep learning. Practical adoption requires people who understand both advanced category theory and production AI systems. That talent pool is limited. Toolchains are also immature: there is no widely adopted categorical compiler for neural architecture design, no standard library for categorical AI safety validation, and no generally accepted benchmark suite for categorical specifications.

The overclaiming risk

Terms such as compositionality, robustness, explainability, and verification can be used too broadly. This report treats them as design hypotheses unless tied to a concrete mechanism or case. A categorical model of a system is not automatically a proof that the deployed system is safe, robust, or interpretable. Additional empirical testing, formal verification, and operational controls remain necessary.

Investment implication

The appropriate posture is selective funding. The field merits attention where structure is already central: scientific AI, robotics, probabilistic reasoning, structured data, AI safety engineering, and agent orchestration. It is less compelling as a near-term substitute for frontier foundation model training.

13. Strategic Roadmap

How the field may develop over the next decade

Near term: two to three years

The near term will likely focus on theory consolidation and targeted prototypes. Research will continue to clarify categorical deep learning, parametric lenses, categorical probability, category-equivariant networks, and reinforcement learning in categorical cybernetics. Early software demonstrations will remain specialized.

Medium term: three to six years

The field may begin producing practical domain-specific tools. Likely candidates include diagrammatic AI pipeline tools, category-equivariant model libraries, probabilistic workflow languages, and architecture generators for constrained scientific and robotic applications.

Long term: six to ten years

If successful, category-theoretic AI may become part of the systems engineering layer for high-stakes AI. It would help specify, compose, audit, and verify large AI systems. Its impact would be less visible than a new model architecture but potentially more foundational.

Decision milestones and monitoring indicators

Two to three years	Working prototypes in narrow domains	Open-source libraries, replicated case studies, and benchmark tasks showing measurable gains in robustness, specification clarity, or engineering effort.
Three to six years	Tooling begins to integrate with mainstream ML stacks	Interfaces to PyTorch or JAX, reusable categorical design patterns, domain-specific compilers, and early enterprise pilots.
Six to ten years	Systems engineering role in high-stakes AI	Adoption in safety-critical workflows, formal assurance pipelines, scientific AI infrastructure, or regulated agentic systems.

14. Conclusion

Category theory as a strategic language for trustworthy AI

Category-theoretic artificial intelligence is not a shortcut to better benchmark scores. It is a long-term research program for making artificial intelligence more compositional, interpretable, verifiable, and structurally disciplined.

Its value increases as AI systems become more complex. A single model can often be managed empirically. A network of models, tools, agents, memories, evaluators, and human governance loops requires a stronger language of composition. Category theory offers precisely such a language.

The field still faces serious obstacles: abstraction, limited tooling, lack of standardization, and relatively few large-scale empirical wins. Yet these limitations are typical of foundational research before it becomes infrastructure. The strategic opportunity is to convert categorical insight into practical design tools for the next generation of AI systems.

The conclusion of this report is therefore clear: category-theoretic AI should be monitored closely, funded selectively, and connected to practical AI engineering. The highest-value opportunities lie at the intersection of structured model design, safety engineering, probabilistic reasoning, scientific AI, and agentic systems. Investment should be gated by concrete milestones: working prototypes, measurable robustness or maintainability benefits, and toolchains that ordinary AI engineers can use.

Selected Sources and Further Reading

The following sources informed the synthesis in this report. Descriptions are provided for orientation rather than as exhaustive bibliographic annotations.

Shiebler, Gavranovic, and Wilson, Category Theory in Machine Learning, 2021.

A foundational survey organizing the field around gradient-based learning, probability, and equivariant learning.

Gavranovic and collaborators, Categorical Deep Learning is an Algebraic Theory of All Architectures, 2024.

A central statement of categorical deep learning as a general architecture theory.

Cruttwell and collaborators, Deep Learning with Parametric Lenses, 2024.

A lens-based account of deep learning algorithms and optimization procedures.

Hedges and Sakamoto, Reinforcement Learning in Categorical Cybernetics, 2024.

A categorical cybernetics account of reinforcement learning algorithms and feedback processes.

Jia and collaborators, Category-Theoretical and Topos-Theoretical Frameworks in Machine Learning, 2024 and 2025.

A survey that expands the landscape to include topos-theoretic perspectives.

Maruyama, Categorical Equivariant Deep Learning, 2025, arXiv preprint.

A generalization of equivariant neural networks from group symmetry to categorical structure; treated here as early-stage research rather than settled industrial practice.

Masulovic, Coalgebras for Categorical Deep Learning, March 2026, arXiv preprint.

A coalgebraic extension of categorical deep learning focused on representation and behavior; included as a recent preprint available before the June 1, 2026 report date, not as a peer-reviewed consensus result.

Coecke, Sadrzadeh, Clark, and related DisCoCat literature.

Foundational work connecting grammar and distributional semantics using category theory.

Fritz, Perrone, Rischel, and related Markov category literature.

Core work in categorical probability and statistical experiments.

Applied Category Theory conference proceedings and community resources.

The leading community venue for applied category theory research across mathematics, computer science, physics, AI, and systems.

Appendix: Executive Glossary

Category	A structured collection of objects and transformations that can be composed.
Functor	A structure-preserving translation from one category to another.
Natural transformation	A disciplined way to compare two structure-preserving translations.
Monad	A structure for organizing computations with effects such as uncertainty, state, failure, or interaction.
Lens	A structure for bidirectional information flow, useful for forward computation and backward updates.
Optic	A generalization of lenses used to describe compositional bidirectional systems.
Markov category	A categorical framework for probabilistic processes and statistical reasoning.
Coalgebra	A framework for systems described through behavior, observation, and state evolution.
Equivariance	The property that outputs transform consistently when inputs are transformed.
DisCoCat	A categorical framework connecting grammar and distributional meaning in natural language.
Topos	A rich categorical structure that connects generalized spaces, logic, and contextual reasoning.