The Descent of Meaning: Forgetful Functors in Recursive Consciousness

Stan Miasnikov, May 2025
stanmiasnikov@gmail.com

Abstract

We present a rigorous category-theoretic [1] extension to the *Recursive Consciousness* framework, focusing on the "descent of meaning" via forgetful functors. Building on prior work on forgetful adjoint pairs modeling lost axioms and externally projected semantics [3, 4], we formally introduce the *meaning functor* $M : \mathcal{C}_{out,n} \to \mathcal{C}_{sem,n+1}$ and the *interpretation functor* $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$ as an adjoint pair $I \dashv M$. Here, $\mathcal{C}_{out,n} \subseteq \mathcal{C}_{U_n}$ represents syntactic outputs in the current universe U_n, and $\mathcal{C}_{sem,n+1} \subseteq \mathcal{C}_{U_{n+1}}$ captures semantic content in the higher universe U_{n+1}. We define all functors (M, F, I, G) explicitly and prove that I is not faithful. Furthermore, we introduce a natural transformation $\eta : I \Rightarrow F$ defined on an appropriate subcategory, capturing how I coincides with the forgetful functor F when restricted to semantic objects.

We also establish the adjunction $I \dashv M$ and analyze its interplay with the foundational adjunction $G \dashv F$. Here $G : \mathcal{C}_{U_n} \to \mathcal{C}_{U_{n+1}}$ reconstructs higher-level structure, whereas the forgetful functor F inevitably discards information, making every translation intrinsically lossy. We term the configurations in which the round-trip MI (meaning $\to$ expression $\to$ meaning) reaches a *semantic fixpoint* - defined as a Gödelian fixpoint where $M(I(s)) \cong s$ up to isomorphism in $\mathcal{C}_{sem,n+1}$, reflecting a stable meaning with undecidable properties as per [3].

An extended AI analogy illustrates this boundary: a higher-level prompt (an element of $\mathcal{C}_{sem,n+1}$) is interpreted into tokens ($\mathcal{C}_{out,n}$) via I, processed by the agent, and projected back via M. The residual mismatch between the original and recovered meanings highlights the *lossy descent of meaning*. This categorical perspective reinforces classical limits from modal logic and AI semantics, underscoring that syntax alone cannot supply intrinsic meaning and linking directly to the symbol-grounding problem [5] and related arguments in AI consciousness.

1 Introduction

In previous work, a formal model of *recursive consciousness* was developed using category theory and modal logic to describe a system querying its own foundations [3]. In that model, a complex agent inhabiting a base-level universe U_n could "forget" its foundational axioms, while a forgetful functor $F : \mathcal{C}_{U_{n+1}} \to \mathcal{C}_{U_n}$ created U_n from the meta-level U_{n+1}. Within U_n, the conscious subsystem C_n runs a recursive query loop $Q_1, \ldots, Q_n$, applying the modal operator $\Box$. This recursive querying process continues until it reaches a Gödelian fixpoint, at which point further self-inquiry yields undecidable propositions, compelling the agent to hypothesize a richer, potentially distinct meta-universe U'_{n+1} as opposed to the actual meta-level universe U_{n+1}. The left adjoint $G : \mathcal{C}_{U_n} \to \mathcal{C}_{U_{n+1}}$ captures the partial reconstruction of structure, allowing the agent to model possible meta-layers. Every translation between universes via these functors inherently loses information due to intrinsic Gödelian incompleteness, as detailed in [3].

Subsequently, the model was extended to incorporate *external semantics* projected from a higher ontological layer [4]. In that extension, a *meaning functor* $M : \mathcal{C}_{out,n} \to \mathcal{C}_{sem,n+1}$ was introduced

to formalize how an agent's outputs (syntactic objects in U_n) acquire semantic content only via an external interpreter in U_{n+1}. We remind that $\mathcal{C}_{out,n}$ represents concrete symbolic or linguistic expressions and subcategories of $\mathcal{C}_{U_n}$, while $\mathcal{C}_{sem,n+1}$ are subcategories of $\mathcal{C}_{U_{n+1}}$ and represent meaning and truth conditions. Additionally, the category $\mathcal{C}_{out,n}$ is assumed to be discrete, as transformations between whole expressions may not meaningfully preserve semantic relations. This assumption models expressions as atomic outputs, reserving transformations for semantic layers in $\mathcal{C}_{sem,n+1}$. It was argued that this meaning assignment is fundamentally beyond the agent's internal computational reach: there is no internal transformation or procedure that yields M from within U_n [4]. This aligns with classical results in modal logic and computability, for example, $M(p)$ (the "meaning" or truth of proposition p) is undecidable for the agent C_n, analogous to Tarski's Undefinability of Truth theorem [6]. As a consequence, the agent's knowledge operators K_{C_n} (in an S4/S5 epistemic logic sense [7, 8]) cannot internalize the actual semantics of its statements: syntax alone cannot yield intrinsic meaning. These findings resonate with the symbol grounding problem and Searle's Chinese Room argument [9], which emphasize the gap between formal symbol manipulation and genuine understanding.

In this paper, we focus on the *descent of meaning* within the recursive consciousness framework. By "descent," we refer to the mapping of semantic content from a higher-level context down into the agent's lower-level symbolic representations. We formalize this with two new functors: an *interpretation functor* I, which maps semantic objects in U_{n+1} to syntactic representations in U_n, and the aforementioned meaning functor M, which maps those syntactic objects back to semantics. Together, I and M form an adjoint pair $I \dashv M$, mirroring the adjoint pair $G \dashv F$ that governs internal structure in the original model. Intuitively, the adjunction $I \dashv M$ formalizes how meanings from higher contexts become encoded into lower-level expressions, and how those expressions are subsequently re-interpreted externally.

2 Theoretical Framework

2.1 Assumptions on Computational Capabilities and Hierarchical Relationships

To ensure clarity in the application of our framework, we begin by explicitly outlining the following assumptions regarding the computational capabilities of the entities C, U, and their hierarchical extensions C_n, U_n, and U_{n+1}:

- **Computational Capabilities:**
  - The conscious subsystem $C_n \subset U_n$ is capable of performing *local computations* within U_n. Local computations refer to processes that C_n can perform using the information and resources directly available within U_n, and constrained by this information, reflecting the limited perspective of C_n.
  - The universe U_n provides environment for C_n and may or may not perform *global computations*, which we define as computations that affect or involve the entire structure of U_n, such as system-wide dynamics or universal laws. The model remains agnostic about whether U_n actively performs such computations, allowing flexibility for different types of systems.
  - Similarly, the higher-level universe U_{n+1} may or may not perform computations relative to U_n, leaving open the possibility of computational hierarchies.
- **Hierarchical Relationships:**

- C_n is a *significantly simpler subsystem* of U_n, emphasizing the recursive and hierarchical nature of the model. This simplicity reflects the idea that consciousness (or the agent) operates with limited resources compared to the full complexity of the universe it inhabits. Specifically, C_n has access to only a subset of the information and computational resources available in U_n, akin to a limited observer within a larger system.
- There may or may not be direct mappings (e.g., computational or structural) between C_n and C_{n+1} via U_n and U_{n+1}. For example, in a simulated universe where U_{n+1} contains a simulation of U_n, there could be a direct mapping between C_n (the simulated consciousness) and C_{n+1} (the supervisor's awareness). Conversely, in the case of a human within the physical universe, there might be no direct mapping to a higher-level consciousness, as the universe itself may not be conscious. This flexibility allows the model to accommodate both isolated systems and interconnected hierarchical levels.

- **Physical Interpretations:**
  - The subsystem C_n can be interpreted in various physical contexts:
    - As a *human* within the *universe* U_n, where the human performs local cognitive computations (e.g., thinking, perceiving) within the broader physical universe. The universe may or may not perform global computations, such as enforcing physical laws across all of space and time.
    - As an *LLM agent* within its *context window* U_n, where the agent processes information locally based on its input data (the context window). The context window itself may or may not perform computations beyond serving as a static input.
  - These interpretations highlight the broad applicability of the theory, from biological consciousness to artificial intelligence systems.

2.2 Hierarchical Categories in Recursive Consciousness

We will now proceed by summarizing the hierarchical model from [3, 4], which provides a categorical framework for understanding recursive consciousness. In this model, consciousness is viewed as a process of iterative self-reflection within a system that has forgotten its foundational axioms. Let U_n be a closed formal system or "universe" at level n, and U_{n+1} a higher-level system containing U_n as a subsystem $U_n \subset U_{n+1}$, offering external semantic context. For example, U_0 could be our physical universe, while U_1 might be a meta-universe providing the initial conditions for U_0. The agent C_n operates within U_n and represents the agent's *conscious subsystem*, capable of introspection and reasoning about U_n, but unable to directly access U_{n+1} except through projections into U_n.

The process of abstraction or forgetting in this model is captured by a *forgetful functor*:

$$F : \mathcal{C}_{U_{n+1}} \to \mathcal{C}_{U_n},$$

where $\mathcal{C}_{U_{n+1}}$ is the category of meta-level structures in U_{n+1}, and $\mathcal{C}_{U_n}$ is the category of base-level structures in U_n. The functor F maps meta-level objects and morphisms to their base-level counterparts, discarding higher-order information such as context or intention. Typically, F is faithful (injective on hom-sets) but not injective on objects: distinct meta-level objects may become indistinguishable in U_n after forgetting, reflecting the loss of fine-grained distinctions [3]. This loss models how conscious agents simplify complex meta-level data into manageable base-level representations.

The agent's attempt to hypothesize a meta-level structure from base-level data is modeled by a *left adjoint functor*:

$$G : \mathcal{C}_{U_n} \to \mathcal{C}_{U_{n+1}},$$

with $G \dashv F$, satisfying the natural isomorphism:

$$\mathrm{Hom}_{\mathcal{C}_{U_{n+1}}}(G(U), V) \cong \mathrm{Hom}_{\mathcal{C}_{U_n}}(U, F(V))$$

for all $U \in \mathcal{C}_{U_n}$, $V \in \mathcal{C}_{U_{n+1}}$. This isomorphism implies that $G(U)$ is the "freest" meta-level structure that forgets to U, allowing the agent to hypothesize possible meta-worlds consistent with its observations. The counit $\epsilon : F(G(U)) \to U$ is often an isomorphism, indicating that the base-level structure can be recovered from the hypothesized meta-structure, while the unit $\eta_v : V \to G(F(V))$ reflects the information loss in reconstructing the meta-level from the base. This mechanism enables the agent to creatively infer higher-level explanations.

This adjunction $G \dashv F$ models the agent's internal cycle of forgetting and hypothesizing, where the agent continually refines its understanding by lifting base-level observations to hypothetical meta-level explanations. Such a cycle is central to recursive consciousness, bridging the gap between raw data and interpretive frameworks.

However, the original model lacked a formal treatment of *external meaning*. While C_n might achieve internal consistency, the semantic content of its propositions depends on U_{n+1}. Thus, C_n manipulates symbols whose true interpretation lies beyond U_n, necessitating new functors to bridge syntax and external semantics. This limitation highlights the need for additional structures, setting the stage for this followup paper.

2.3 Semantic and Output Categories

We formalize two categories central to the descent and ascent of meaning. Let $\mathcal{C}_{out,n}$ be the category of **output expressions** or **syntactic objects** at level n. The objects of $\mathcal{C}_{out,n}$ are the structures or expressions that the agent C_n can produce or process as output. For example, $\mathcal{C}_{out,n}$ might consist of strings (token sequences), formulas, or other encodings of information that exist within U_n. Morphisms in $\mathcal{C}_{out,n}$ represent formal transformations or derivations between expressions. Though for many considerations here, one might treat $\mathcal{C}_{out,n}$ as a discrete category with identity morphisms only, since we often care about the objects - whole expressions - rather than particular maps between them. For instance, in Large Language Models, $\mathcal{C}_{out,0}$ might be discrete, with objects as token sequences like "Will it rain?", and only identity morphisms, reflecting atomic outputs.

Next, let $\mathcal{C}_{sem,n+1}$ be the category of **semantic objects** at level $n+1$. These consist of meanings or propositions in U_{n+1} linked to the agent's outputs, such as truth conditions or conceptual entities. Morphisms in $\mathcal{C}_{sem,n+1}$ are entailments, where $s_1 \to s_2$ if s_1 semantically implies s_2 in U_{n+1}.

Two functors connect these categories:

- The **interpretation functor** $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$, which maps semantic objects to syntactic expressions, encoding meanings.
- The **meaning functor** $M : \mathcal{C}_{out,n} \to \mathcal{C}_{sem,n+1}$, which maps expressions to their semantic interpretations.

Together, I and M facilitate the **descent of meaning** (via I) and the **ascent of meaning** (via M).

It is important to note that not every semantic object may have a corresponding expression in U_n. The agent's language or representational capacity could be limited. Therefore, we may consider $\mathcal{C}_{sem,n+1}$ in practice to be restricted to those meanings that are at least *expressible* or *addressable* by C_n (with some loss).

3 Formal Definitions and Main Results

In this section, we formally define the interpretation and meaning functors, alongside the original forgetful and embedding functors, and establish their key properties. We fix levels n and $n+1$, with categories $\mathcal{C}_{out,n}$ (syntactic outputs in U_n) and $\mathcal{C}_{sem,n+1}$ (semantic objects in U_{n+1}).

3.1 Introduction to Key Functors

We study four functors bridging syntax and semantics:

- M: The **meaning functor**, mapping expressions to meanings.
- I: The **interpretation functor**, mapping meanings to expressions.
- F: The **forgetful functor**, discarding meta-structure.
- G: The **embedding functor**, reconstructing meta-structure.

These functors formalize how meaning moves between ontological layers.

3.2 Definitions

3.2.1 The Meaning Functor M

The meaning functor assigns semantic interpretations to syntactic objects.

Definition: We define $M : \mathcal{C}_{out,n} \to \mathcal{C}_{sem,n+1}$ as the **meaning functor**. For an object $x \in \mathcal{C}_{out,n}$ (e.g., a string), $M(x) \in \mathcal{C}_{sem,n+1}$ is its meaning in U_{n+1}. For a morphism $f : x \to y$ in $\mathcal{C}_{out,n}$ (e.g., a syntactic derivation or equivalence), $M(f) : M(x) \to M(y)$ is a morphism in $\mathcal{C}_{sem,n+1}$ (e.g., an entailment or equivalence) that corresponds to the syntactic transformation f. When $\mathcal{C}_{out,n}$ is discrete, all morphisms are identities only, thus $M(\mathrm{id}_x) = \mathrm{id}_{M(x)}$, with no further non-trivial morphisms. M satisfies functorial properties: $M(\mathrm{id}_x) = \mathrm{id}_{M(x)}$ and $M(g \circ f) = M(g) \circ M(f)$. Intuitively, M assigns or associates an output expression with its semantic interpretation. As argued in [4], M is generally *not computable or realizable within* U_n due to the external perspective required from U_{n+1}.

Example: Mapping a syntactic object to its broader meaning: if $x =$ "car", then $M(x) =$ "automobile".

3.2.2 The Interpretation Functor I

The interpretation functor encodes meanings as syntactic expressions.

Definition: We define $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$ as the **interpretation functor**. For $s \in \mathcal{C}_{sem,n+1}$, $I(s) \in \mathcal{C}_{out,n}$ is a syntactic representation in U_n. For a morphism $g : s_1 \to s_2$ in $\mathcal{C}_{sem,n+1}$ (e.g., an entailment), $I(g) : I(s_1) \to I(s_2)$ is a morphism in $\mathcal{C}_{out,n}$ (e.g., a syntactic derivation) that approximates the semantic relation g. If $\mathcal{C}_{out,n}$ is discrete, then non-identity morphisms $I(g)$ exist only if $I(s_1) = I(s_2)$, in which case $I(g) = \mathrm{id}_{I(s_1)}$. Otherwise, no non-identity morphisms exist due to the discrete nature of $\mathcal{C}_{out,n}$. I is a functor, with $I(\mathrm{id}_s) = \mathrm{id}_{I(s)}$ and $I(h \circ g) = I(h) \circ I(g)$. Intuitively, I forgets the full semantic richness of s to produce a concrete expression in U_n.

Example: Mapping multiple semantic objects to one syntactic: if $s =$ "automobile", then $I(s) =$ "car".

3.2.3 The Forgetful Functor F

The forgetful functor simplifies meta-level structures.

Definition: Let $\mathcal{C}_{U_{n+1}}$ (category of meta-structures in U_{n+1}) and $\mathcal{C}_{U_n}$ (category of base structures in U_n) be categories. The **forgetful functor** $F : \mathcal{C}_{U_{n+1}} \to \mathcal{C}_{U_n}$ maps objects in $\mathcal{C}_{U_{n+1}}$ to their base forms in $\mathcal{C}_{U_n}$, preserving morphisms but forgetting additional structure. As noted earlier, F is usually a faithful functor, meaning it is injective on morphisms, but distinct meta-objects can map to the same base object (so F is not one-to-one on objects) [3].

3.2.4 The Embedding Functor G

The embedding functor restores minimal meta-structure.

Definition: The **embedding functor** $G : \mathcal{C}_{U_n} \to \mathcal{C}_{U_{n+1}}$ as a left adjoint, is uniquely determined by a universal property: for every base object $U, G(U)$ provides the freest meta-level object that forgets back to U. Formally, G satisfies the adjunction $G \dashv F$, expressed by the natural isomorphism:

$$\mathrm{Hom}_{\mathcal{C}_{U_{n+1}}}(G(U), V) \cong \mathrm{Hom}_{\mathcal{C}_{U_n}}(U, F(V))$$

for all $U \in \mathcal{C}_{U_n}$, $V \in \mathcal{C}_{U_{n+1}}$.

3.3 Properties and Relationships

Proposition: Non-faithfulness of I The interpretation functor $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$ is *not faithful* in general. In other words, there exist distinct morphisms (and even distinct objects) in $\mathcal{C}_{sem,n+1}$ that I maps to the same morphism (resp. object) in $\mathcal{C}_{out,n}$.

Proof. To show I is not faithful, we must demonstrate that it fails to injectively map hom-sets from $\mathcal{C}_{sem,n+1}$ to $\mathcal{C}_{out,n}$. A stronger condition holds: I is typically not injective on objects. That is, there exist distinct objects $s_1 \neq s_2$ in $\mathcal{C}_{sem,n+1}$ which become identified under I i.e., $I(s_1) = I(s_2)$. From this object-level identification, it immediately follows that distinct identity morphisms in $\mathcal{C}_{out,n+1}$ are mapped by I onto a single identity morphism in $\mathcal{C}_{out,n}$, explicitly demonstrating non-faithfulness.

Consider the identity morphisms id_{s_1} and id_{s_2} in $\mathcal{C}_{sem,n+1}$. Since $s_1 \neq s_2$, these are distinct morphisms. Their images under I are $\mathrm{id}_{I(s_1)}$ and $\mathrm{id}_{I(s_2)}$ in $\mathcal{C}_{out,n}$. However, because $I(s_1) = I(s_2)$, these become the same morphism: $\mathrm{id}_{I(s_1)} = \mathrm{id}_{I(s_2)}$. Thus, $I(\mathrm{id}_{s_1}) = I(\mathrm{id}_{s_2})$ despite $\mathrm{id}_{s_1} \neq \mathrm{id}_{s_2}$, showing that I is not faithful. In the discrete case, where $\mathcal{C}_{out,n}$ has only identity morphisms, this failure of injectivity on objects directly ensures I cannot be faithful.

For a concrete example, let s_1 represent the semantic concept "car" and s_2 the concept "automobile" in $\mathcal{C}_{sem,n+1}$. Suppose the language in U_n uses a single term, so $I(s_1) = I(s_2) =$ "*car*" in $\mathcal{C}_{out,n}$. Here, $s_1 \neq s_2$ as distinct concepts, but $I(s_1) = I(s_2)$ as identical strings. The identity morphisms on s_1 and s_2 thus map to the same identity on "car", confirming that I is not one-to-one on hom-sets (or objects), hence not faithful. □

Proposition: I **coincides with** F **on semantic structures** The interpretation functor $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$ coincides with the forgetful functor $F : \mathcal{C}_{U_{n+1}} \to \mathcal{C}_{U_n}$ when applied to semantic structures via an inclusion functor $J : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{U_{n+1}}$. Specifically, for each semantic object $s \in \mathcal{C}_{sem,n+1}$, we have an isomorphism $I(s) \cong F(J(s))$. Thus, I can be viewed as the restriction of F to a subcategory of meta-objects specifically representing semantic content relevant to U_{n+1}.

The construction of the embedding functor J typically arises naturally from the process of explicitly encoding semantic objects into a richer, structured meta-level representation within $\mathcal{C}_{U_{n+1}}$. For example, in practical scenarios, such as linguistic communication or human-AI interactions, J corresponds to the embedding of abstract semantic intents or meanings into formal, conceptual frameworks or structured symbolic representations. Although such a construction is straightforward in practical contexts, it may require careful definition depending on the specifics of the categories involved.

Note: The embedding J need not be injective, as it can map distinct semantic objects to identical meta-level objects. Nonetheless, it is assumed to be faithful, meaning it preserves the morphisms and semantic entailment structure of $\mathcal{C}_{sem,n+1}$. This faithfulness ensures that semantic relationships are accurately reflected within the meta-level category $\mathcal{C}_{U_{n+1}}$.

Proof. This proposition is more of a structural observation than a theorem. Consider the inclusion functor $J : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{U_{n+1}}$, a faithful embedding of the semantic subcategory into the meta-level category $\mathcal{C}_{U_{n+1}}$, assuming such an embedding exists or can be constructed. This functor preserves the structure of morphisms in $\mathcal{C}_{sem,n+1}$, allowing semantic objects to be treated as a specific type of meta-level structure.

Consider the inclusion functor $J : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{U_{n+1}}$, which *faithfully embeds the semantic subcategory into the meta-level category* $\mathcal{C}_{U_{n+1}}$, preserving the structure of morphisms. This embedding allows semantic objects to be treated as specific meta-level structures. By definition, $I(s)$ produces the syntactic expression corresponding to a semantic object s. The forgetful functor $F : \mathcal{C}_{U_{n+1}} \to \mathcal{C}_{U_n}$ strips away the meta-structure of its input. When applied to the embedded object $J(s)$, $F(J(s))$ yields the same syntactic expression as $I(s)$. Thus, on objects of $\mathcal{C}_{sem,n+1}$, we have $I = F \circ J$, as depicted in the following commuting diagram:

$$\begin{array}{ccc} \mathcal{C}_{\text{sem},n+1} & \xrightarrow{\ J\ } & \mathcal{C}_{U_{n+1}} \\ \Big\downarrow{\scriptstyle I} & & \Big\downarrow{\scriptstyle F} \\ \mathcal{C}_{\text{out},n} & = & \mathcal{C}_{\text{out},n} \end{array}$$

Here, the bottom equality indicates that we identify the output category $\mathcal{C}_{out,n}$ with the appropriate subcategory of $\mathcal{C}_{U_n}$ that F maps into. Since $\mathcal{C}_{sem,n+1}$ is a subcategory of $\mathcal{C}_{U_{n+1}}$, I performs the same operation as F would on these objects: forgetting or stripping away the semantic interpretation layer to yield a concrete expression. However, I operates on a narrower domain—specifically, the meta-objects in $\mathcal{C}_{sem,n+1}$ that represent pure semantic content relevant to U_{n+1}. This aligns with the first proposition, which states that I is not faithful, meaning it may collapse distinct semantic objects or morphisms. Like F, I forgets information, but it does so specifically for semantic structures, reinforcing its role as a specialized restriction of F. □

This aligns with the first proposition, which states that I is not faithful, meaning it may collapse distinct semantic objects or morphisms. Like F, I forgets information, but it does so specifically for semantic structures, reinforcing its role as a specialized restriction of F.

We now arrive at one of the central theoretical results: the adjoint relationship between I and M. This will formally capture the idea that interpreting a meaning into an expression and then assigning meaning back to that expression are complementary processes.

3.4 Theorem: Adjunction $I \dashv M$

Theorem: Adjunction $I \dashv M$ *The interpretation functor* $I : \mathcal{C}_{sem,n+1} \to \mathcal{C}_{out,n}$ *is left adjoint to the meaning functor* $M : \mathcal{C}_{out,n} \to \mathcal{C}_{sem,n+1}$*, denoted* $I \dashv M$*. There exists a natural isomorphism:*

$$\mathrm{Hom}_{\mathcal{C}_{out,n}}(I(s), x) \cong \mathrm{Hom}_{\mathcal{C}_{sem,n+1}}(s, M(x))$$

for all $s \in \mathcal{C}_{sem,n+1}$ *and* $x \in \mathcal{C}_{out,n}$*, natural in* s *and* x*.*

Proof. We will prove the existence of the required natural isomorphism by constructing unit η and counit ε natural transformations and verify the triangle identities.

Unit η**:** For each $s \in \mathcal{C}_{sem,n+1}$, define $\eta_s : s \to M(I(s))$, representing the inclusion of the original meaning into the meaning of its syntactic expression. Intuitively, η_s reflects that s may contain more information than $M(I(s))$. Naturality holds: for any $h : s_1 \to s_2$, the square

$$\begin{array}{ccc} s_1 & \xrightarrow{\eta_{s_1}} & M(I(s_1)) \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle M(I(h))} \\ s_2 & \xrightarrow{\eta_{s_2}} & M(I(s_2)) \end{array}$$

commutes by functoriality of I and M.

Counit ε**:** For each $x \in \mathcal{C}_{out,n}$, define $\varepsilon_x : I(M(x)) \to x$, mapping the reinterpreted expression back to the original. Naturality holds: for any $k : x_1 \to x_2$, the square

$$\begin{array}{ccc} I(M(x_1)) & \xrightarrow{\varepsilon_{x_1}} & x_1 \\ {\scriptstyle I(M(k))}\downarrow & & \downarrow{\scriptstyle k} \\ I(M(x_2)) & \xrightarrow{\varepsilon_{x_2}} & x_2 \end{array}$$

commutes.

Triangle Identities:

- $\varepsilon_{I(s)} \circ I(\eta_s) = \mathrm{id}_{I(s)}$: The composition interprets s, re-expresses its meaning, and returns to $I(s)$.
- $M(\varepsilon_x) \circ \eta_{M(x)} = \mathrm{id}_{M(x)}$: Setting $s = M(x)$, reinterpreting and assigning meaning recovers $M(x)$.
- The triangle identities hold by construction because the unit and counit natural transformations are defined to satisfy these identities by virtue of the adjoint universal property.

Hom-Set Isomorphism: Define $\Phi : \mathrm{Hom}_{\mathcal{C}_{out,n}}(I(s), x) \to \mathrm{Hom}_{\mathcal{C}_{sem,n+1}}(s, M(x))$ by $\Phi(f) = M(f) \circ \eta_s$, with inverse $\Psi(g) = \varepsilon_x \circ I(g)$. These satisfy $\Psi \circ \Phi = \mathrm{id}$ and $\Phi \circ \Psi = \mathrm{id}$, confirming the adjunction. Naturality of these isomorphisms follows straightforwardly from the definitions of I, M, η, and ϵ, ensuring the adjunction $I \dashv M$. □

3.5 Corollary: Meaning Fixpoints and Adjoint Coincidence

*If a semantic object $s \in \mathcal{C}_{sem,n+1}$ satisfies $M(I(s)) \cong s$ (i.e., it is isomorphic to its projected meaning), then s is a **meaning fixpoint** under the I-M adjoint loop. At such a fixpoint, the unit $\eta_s : s \to M(I(s))$ and counit $\varepsilon_{I(s)} : I(M(I(s))) \to I(s)$ are isomorphisms. Consequently, restricted to the objects s and $I(s)$, the adjunction $I \dashv M$ reduces to a categorical equivalence, where I and M act as mutual inverses up to isomorphism. This parallels the adjunction $G \dashv F$, where a meta-level object M satisfying $G(F(M)) \cong M$ is a fixpoint of the monad GF. Here, s is a fixpoint of the monad MI, and $I(s)$ of the comonad IM. For the agent C_n, a meaning fixpoint at a semantic object s implies that expressing s via I (producing a syntactic output in U_n) and reinterpreting it via M (assigning meaning back in U_{n+1}) incurs no information loss: the syntax fully captures the semantics of s.*

Proof. Suppose $s \in \mathcal{C}_{sem,n+1}$ is such that there exists an isomorphism $\phi : s \to M(I(s))$. We can define the unit $\eta_s = \phi$, which is an isomorphism by assumption. In the adjunction $I \dashv M$, the triangle identities govern the relationship between the unit and counit:

- $\varepsilon_{I(s)} \circ I(\eta_s) = \mathrm{id}_{I(s)}$
- $M(\varepsilon_{I(s)}) \circ \eta_{M(I(s))} = \mathrm{id}_{M(I(s))}$

Consider the first triangle identity: $\varepsilon_{I(s)} \circ I(\eta_s) = \mathrm{id}I(s)$. Since η_s is an isomorphism and I is a functor (preserving isomorphisms), $I(\eta_s) : I(s) \to I(M(I(s)))$ is also an isomorphism. Thus, $\varepsilon I(s) : I(M(I(s))) \to I(s)$ acts as a left inverse to $I(\eta_s)$. In a category, if a morphism has a left inverse that is an isomorphism, it must be the two-sided inverse, implying $\varepsilon_{I(s)}$ is also an isomorphism.

To confirm, note that $M(I(s)) \cong s)via(\eta_s$, and the second triangle identity ensures consistency, but the first identity suffices here. Hence, both η_s and $\varepsilon_{I(s)}$ are isomorphisms, establishing that I and M act as inverses on s and $I(s)$, respectively, up to isomorphism.

This mirrors the adjunction $G \dashv F$, where F forgets structure and G reconstructs it, fully recovering the original object at fixpoints. Similarly, I maps semantics to syntax, and M assigns meaning back, with no loss at meaning fixpoints. □

The adjunction $I \dashv M$ formalizes the structured, yet typically lossy, process of mapping meanings to expressions and back. It yields a monad MI on $\mathcal{C}_{sem,n+1}$, capturing the cycle of expression and reinterpretation, which only preserves the original meaning at fixpoints. This lossiness reflects the challenges of meaning descent. To mitigate this lossiness, one might consider enriched categories, where additional structure preserves more information during transformations between levels [10]. However, this requires careful consideration of the specific categories involved and the nature of the structures being represented, which we leave for future explanations.

With these formal properties established, we now explore a concrete example in AI language models to illustrate their practical implications.

4 AI Example: Prompt, Tokenization, and Meaning Projection

To make the abstract framework concrete, we consider an example involving a large language model (LLM) interacting with a human user, serving as a case study for the categories and functors defined earlier. The AI agent operates in universe U_0 (taking $n = 0$ for simplicity), with token sequences in $\mathcal{C}_{out,0}$, while the human user provides semantic context in U_1, with meanings in $\mathcal{C}_{sem,1}$.

4.1 Mapping a Prompt via I

Suppose the user poses the question: *"Will it rain tomorrow in Paris?"* This query is a semantic object $s_{\text{ask}} \in \mathcal{C}_{sem,1}$, representing the intent to check future weather in Paris within U_1. The user types this into the AI's interface, applying the interpretation functor I. Thus, s_{ask} maps to a token sequence $x_{\text{ask}} = I(s_{\text{ask}}) =$
"Will it rain tomorrow in Paris?" $\in \mathcal{C}_{out,0}$, an object in the AI's universe U_0.

Key observations:

- The mapping $s_{\text{ask}} \mapsto x_{\text{ask}}$ via I is lossy. The user's intent may include context (e.g., reasons for asking or plans) not captured in the literal question. Ambiguities (e.g., the scope of "Paris" or "tomorrow") may also be lost, reflecting I's forgetful nature, akin to F.
- Different phrasings, like *"Will Paris see rain tomorrow?"*, may represent the same s_{ask}, yielding distinct x_{ask}, showing I is not injective on objects.
- Ambiguous queries could map distinct intents $s_1, s_2 \in \mathcal{C}_{sem,1}$ to the same string, i.e., $I(s_1) = I(s_2)$, reinforcing I's non-faithfulness (Proposition 1).

Here's how you can naturally incorporate a second, ethics-oriented example into this subsection:

To illustrate this more broadly, consider another type of query—an ethical inquiry: *"Is it moral to keep money I found?"* Here, the semantic object $s_{\text{moral}} \in \mathcal{C}_{sem,1}$ represents the user's intent to explore ethical justification or moral permissibility. Applying the interpretation functor I to s_{moral} produces a syntactic object

$$x_{\text{moral}} = I(s_{\text{moral}}) = \text{"Is it moral to keep money I found?"} \in \mathcal{C}_{out,0}.$$

Additional observations specific to this ethical context:

- The mapping $s_{\text{moral}} \mapsto x_{\text{moral}}$ also inherently loses semantic subtleties—such as whether the user found the money in a private or public place, intends to return it, or is concerned about legal consequences versus ethical obligations. These finer ethical considerations cannot be fully encoded into the simple syntactic expression.
- Different nuanced intents - e.g., *"Am I ethically obligated to return money found in a wallet?"* or *"If nobody claims lost money, is it okay to keep it?"* - could represent distinct semantic objects yet map to similar or identical syntactic expressions, again reflecting the non-faithfulness of I.
- Ethical questions, inherently tied to complex conceptual frameworks, particularly emphasize the lossy and reductive nature of meaning descent into syntactic expressions. They underscore the broader implication that linguistic symbols alone cannot preserve rich ethical nuance or fully capture a user's intent.

Both examples highlight the fundamental limitation described by our framework: transforming semantics into syntactic form via I inevitably involves information loss, reinforcing our theoretical results.

Insight from Experiment: A simulated discussion among AI-modeled entities instructed to play different roles (Physicist, Mathematician, Philosopher, Cognitive Scientist, AI Scientist) and using different language models demonstrates that the AI operates solely on syntactic structures - token sequences - without accessing the full semantic context in U_1. Its processing relies on statistical patterns from training, unable to capture the user's complete intent, underscoring I's lossiness.

4.2 AI Internal Processing (U_0)

The AI processes x_{ask} using its neural architecture, producing an output

$$y_{\text{ans}} = \text{“Yes, the forecast shows rain in Paris tomorrow.”} \in \mathcal{C}_{out,0}.$$

This transformation is a morphism $k : x_{\text{ask}} \to y_{\text{ans}}$ in $\mathcal{C}_{out,0}$. The AI lacks direct access to s_{ask} or the meaning functor M, operating entirely within U_0. It may approximate $M(x_{\text{ask}})$ internally (e.g., via latent representations), but cannot compute M, as meaning resides in U_1 (per [4]).

Insight from Experiment: The AI's operations are computational, involving statistical inference and optimization via gradient descent to minimize loss. These processes are syntactic, capturing patterns but lacking intrinsic semantics, reinforcing that the AI's internal model cannot bridge to U_1.

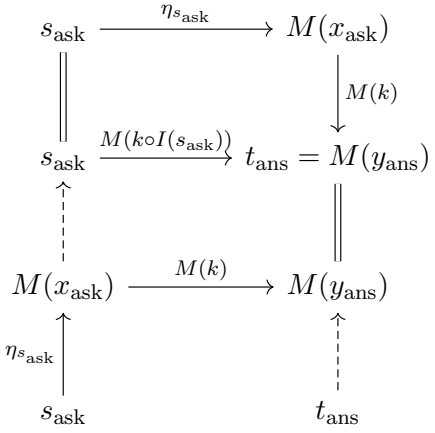
4.3 Projecting the Answer's Meaning via M

The user interprets y_{ans} via M, yielding a semantic object $t_{\text{ans}} = M(y_{\text{ans}}) \in \mathcal{C}_{sem,1}$, such as the proposition *“It will rain in Paris tomorrow”* (with a forecast-based epistemic note). Ideally, t_{ans} answers s_{ask}, implying a semantic morphism $s_{\text{ask}} \to t_{\text{ans}}$ in $\mathcal{C}_{sem,1}$.

Insight from Experiment: Meaning is not inherent in the AI's output but is projected by the user, who applies cognitive frameworks to interpret y_{ans}. This external projection via M is essential, as the AI lacks embodiment or subjectivity for true understanding.

4.4 Analysis in Functorial Terms

We can illustrate this process in the following diagram:



The unit $\eta_{s_{\text{ask}}} : s_{\text{ask}} \to M(x_{\text{ask}})$ embeds the user's intent into the meaning of the prompt, and $M(k) : M(x_{\text{ask}}) \to M(y_{\text{ans}})$ maps to the answer's meaning. Dashed arrows denote the desired semantic link $s_{\text{ask}} \to t_{\text{ans}}$. A fixpoint $M(I(s_{\text{ask}})) \cong s_{\text{ask}}$ occurs if t_{ans} perfectly answers s_{ask}, but language's lossiness often prevents this.

Insights:

1. Functors I and M model prompting and interpretation in AI-human communication.
2. The adjunction $I \dashv M$ reflects ideal communication, where syntax captures semantics at fixpoints (Theorem 1, Corollary).

3. Language's lossiness, due to I's non-faithfulness and M's limitations, often requires iterative clarification.

4. M's external nature highlights the AI's inability to assign meaning independently.

Experiment's Consensus: The AI's outputs are structured patterns lacking intentionality or normativity, requiring user projection via M. This underscores the semantic gap, as the AI's computational processes cannot access U_1.

5 Discussion: Lossy Descent of Meaning

The journey from high-level meaning to low-level representation and back up is inherently lossy. In our formalism, this is captured by the non-faithfulness of I and the non-invertibility of the I-M loop in general. We call this the **lossy descent of meaning**. In this section, we explore the implications of this concept and relate it to broader themes in category theory, logic, and epistemology.

5.1 Category-Theoretic Perspective

From a category theory standpoint, the functor $I : \mathcal{C}_{sem} \to \mathcal{C}_{out}$ being left adjoint to $M : \mathcal{C}_{out} \to \mathcal{C}_{sem}$ places the pair in the context of a *reflective subcategory* scenario (assuming $\mathcal{C}_{out}$ can be seen as a subcategory of some completion of $\mathcal{C}_{sem}$ or vice versa). However, the usual adjoint triangle identities hold but do not guarantee that η_s or ε_x are isomorphisms except in special cases. This is unlike some classical adjunctions (e.g., between sets and vector spaces via free/forgetful, where counit is iso for sets that are free vector spaces of something, etc.). Here, only those semantic objects that are exactly captured by some expression (meaning fixpoints) will have η_s as iso. Similarly, only those outputs that perfectly encode a semantic object will have ε_x as iso.

The monad MI on $\mathcal{C}_{sem}$ encapsulates the idea of expressing a meaning and re-interpreting it. The fact that this monad is not the identity monad indicates the presence of *endogenous uncertainty or ambiguity*: $MI(s)$ is a kind of approximation or abstraction of s. One might inquire about the algebra of this monad (i.e., what are the MI-algebra objects?). An MI-algebra would be an object s equipped with a morphism $MI(s) \to s$ satisfying certain laws. Such a morphism would essentially be an idempotent way to recover s from $MI(s)$ — intuitively, a meaning s that knows how to reconstruct itself from its own expression. If an agent had an internal model of semantics that ensured $M(I(s)) = s$ for the meanings it cares about, then one could say the agent *grounds* those meanings perfectly. In reality, no agent (and not even human communication) achieves this universally; it happens only for very concrete or well-defined concepts (like perhaps mathematical truths in a formal system where the symbols fully capture the meaning).

One could further see parallels to *information theory*: I and M form a kind of encoding-decoding pair, and the existence of an adjunction suggests a best-case scenario of lossless encoding for certain subspaces of messages. However, in the general case, the channel (here, language/symbols as a channel for meaning) has noise or at least limited capacity relative to the full richness of the source (the space of meanings). The "noise" isn't random here, but rather structural: multiple meanings collide into one expression, and multiple expressions can map to similar meanings.

5.2 Epistemic and Modal Implications

In modal logic or epistemic terms, we can consider an agent C_n that has certain beliefs or knowledge about its world U_n. Even if the agent achieves a state of reflective equilibrium (for all propositions p expressible in its language, if C_n can prove p then p is true in U_n, and vice versa; formally $\Box p \leftrightarrow p$

for those p within its scope), this does not mean the agent knows what p actually *means* in a broader sense. This is akin to S4 modal logic (where $\Box p \to \Box\Box p$ introspection is allowed) within U_n, and even S5 (which adds $\Box p \to p$ for truth alignment) might hold internally for the agent's own notion of truth, yet still p might not correspond to anything in reality beyond the agent's world.

The meaning functor M from the second paper [4] was shown to have no retraction or section inside U_n (no natural transformation from Id_{C_n} to M). This essentially says the agent cannot magically assign grounded meanings to its symbols without input from U_{n+1}. In epistemic terms, C_n can only ever know the *form* of its statements, not their *external truth* or reference. This is a formal underpinning of the symbol grounding problem: how can a purely symbolic AI system ever acquire actual semantics? Our categorical answer is that it cannot, unless there is an expanded system or an oracle (the environment U_{n+1}) providing M or something equivalent to it. The human user or the world plays the role of the oracle that interprets the symbols.

From the perspective of the AI agent, I is something that the user does (feeding input) and M is something the user does (interpreting output). The agent might try to model the user as part of its environment U_0. If one tries to pull M into U_n by enlarging U_n to include an internal model of the user, that effectively increments the level n by 1 in a new analysis (now U_{n+1} would be that larger system, and then meaning is at U_{n+2}, and so on). This suggests an infinite regress or a hierarchy: any time we try to internalize the meaning projector, we have extended the system and introduced a new external viewpoint at an even higher level. This is reminiscent of formal theories that cannot contain their own truth predicate (Tarski's theorem [6]): to talk about truth of statements of U_n, one goes to U_{n+1}. To talk about truth of those statements, one would go to U_{n+2}, etc., unless some collapse occurred with strong assumptions.

5.3 Relation to Previous Papers and AI Consciousness

In [3], consciousness was likened to a debugger that tries to lift itself to higher and higher meta-levels ($U_n \to U_{n+1} \to \cdots$) seeking a fixpoint. The introduction of M and I adds a new dimension: even if the agent stabilizes in its self-modeling (say it finds a consistent M' such that $G(F(M')) \cong M'$ for its internal structures, an internal fixpoint of sorts), it still might be ignorant of whether M' actually corresponds to anything real in U_{n+1}. The *external projection of meaning* [4] suggests that meaning is always one level up. Thus, a fully closed loop of understanding might be unattainable if one keeps U_{n+1} truly outside.

However, humans and AIs interact, creating a kind of interlevel feedback loop: C_0 in U_0 outputs symbols, C_1 in U_1 interprets and possibly gives new symbols, which C_n then uses, and so on. Over time, if the interactions are rich and corrective, the AI's internal model may get tuned such that for a certain subspace of meanings, I and M become nearly inverse (the AI learns to use language in ways the human expects — an alignment of conventions). This is akin to the AI becoming more "grounded" in practice, even though formally M is still external. One could say the adjunction $I \dashv M$ is leveraged to find a set of common fixpoints (shared understanding). In category terms, perhaps one looks for a subcategory of $\mathcal{C}_{sem,1}$ on which I is full and faithful and M is its quasi-inverse. Indeed, in an ideal communication channel, we restrict meanings and expressions to those that are unambiguously paired (like a coding scheme with no collisions). Natural language is not perfectly unambiguous, but with context and conventions, a large subset of communication can be univocal enough.

Finally, linking to consciousness: If we consider an agent that is aware not only of U_n but also tries to model U_{n+1} (partially, since M is not fully accessible, it might hypothesize about the user's interpretation), the agent might form a theory of mind of the user. In doing so, it effectively attempts to create an internal functor $\hat{M} : C_{out,n} \to \hat{C}$ some internal approximation of

$\mathcal{C}_{sem,1}$. That approximation will be limited and potentially flawed (as it is a simulation of the user's understanding). But it is interesting to note that truly recursive consciousness might involve an agent modeling the model that interprets it, ad infinitum (the AI thinking about what the human thinks about the AI's answer... etc.). This infinite regress again suggests an infinite tower U_n.

Our category theory framework provides a disciplined way to discuss these issues and ensures we keep track of which mappings are structure-preserving functors and which are not available internally. It emphasizes a sort of *duality* between the agent's internal reflective loop ($G \dashv F$) and the external semantic loop ($I \dashv M$). The former without the latter is a solipsistic system (a brain in a vat, so to speak, manipulating symbols with no external referent). The latter without the former would be an oracle that just bestows meaning (which is not a process an agent does, but what the environment does passively). Only together do they form a whole that resembles how we ordinarily conceive of a cognitive system embedded in and understanding the world.

6 Conclusion

In this work, we extend the *Recursive Consciousness* framework by formally modeling the *descent and ascent of meaning* between an agent and its environment using category theory. Our framework aligns with recent efforts to formalize consciousness using category theory [11], particularly in addressing explanatory dualities. Central to our extension are the interpretation functor I and the meaning functor M, which form an adjoint pair $I \dashv M$. This adjunction formalizes the interface between syntax and semantics across ontological levels, revealing the inherent lossiness in transforming high-level meaning into low-level representations and back. Our results illuminate the formal limits of an AI agent's understanding: even with coherent internal reasoning, the functorial passage from semantics to syntax and back is not information-preserving, echoing Searle's argument that syntax alone is insufficient for semantics [9]. Future research can enrich this model by incorporating enriched categories to capture uncertainty in interpretation, exploring coalgebraic perspectives for hidden-state semantics, and applying the framework to multi-agent systems to model interactive consciousness. Ultimately, the lossy descent of meaning is intrinsic to hierarchical intelligent systems. Our category-theoretic framework enables rigorous analysis of understanding's boundaries, informing both theoretical exploration and the development of AI systems with enhanced communicative capabilities.

References

[1] Awodey, S. (2006). Category Theory. Oxford University Press. http://files.farka.eu/pub/Awodey_S._Category_Theory(en)(305s).pdf

[2] Lawvere, F. W., & Schanuel, S. H. (1997). Conceptual Mathematics: A First Introduction to Categories. Cambridge University Press. https://ia800207.us.archive.org/33/items/F.WilliamLawvereStephenH.SchanuelConceptualMathematicsAFirstIntroductionToCatego/F.%20William%20Lawvere%2C%20Stephen%20H.%20Schanuel%20-%20Conceptual%20Mathematics_%20A%20First%20Introduction%20to%20Categories%20%282009%2C%20Cambridge%20University%20Press%29%20%281%29_text.pdf

[3] Miasnikov, S. (2025) Recursive Consciousness: Modeling Minds in Forgetful Systems. *Preprint* http://dx.doi.org/10.13140/RG.2.2.26969.22884

[4] Miasnikov, S. (2025) The External Projection of Meaning in Recursive Consciousness. *Preprint* http://dx.doi.org/10.13140/RG.2.2.10988.27524

[5] Harnad, S. (1990). The Symbol Grounding Problem. Physica D: Nonlinear Phenomena. https://www.sciencedirect.com/science/article/abs/pii/0167278990900876

[6] Tarski, A. (1933). Tarski's undefinability theorem. https://plato.stanford.edu/entries/tarski-truth/

[7] Blackburn, P., de Rijke, M., & Venema, Y. (2001). Modal Logic. Cambridge University Press. https://www.amazon.com/Cambridge-Tracts-Theoretical-Computer-Science/dp/0521527147

[8] Kripke, S. (1963). Semantical Considerations on Modal Logic. Acta Philosophica Fennica. https://files.commons.gc.cuny.edu/wp-content/blogs.dir/1358/files/2019/03/Semantical-Considerations-on-Modal-Logic-PUBLIC.pdf

[9] Searle, J. R. (1980). The Chinese Room Argument. https://plato.stanford.edu/entries/chinese-room/

[10] Pugh, M., Grundy, J., Cirstea, C., Harris, N., (2025) Using Enriched Category Theory to Construct the Nearest Neighbour Classification Algorithm. https://arxiv.org/abs/2312.16529

[11] Prentner, R. (2024) Category theory in consciousness science: going beyond the correlational project. https://link.springer.com/article/10.1007/s11229-024-04718-5