Beyond the Weight Matrix: A compartmental, equation-based architecture for editable and verifiable machine knowledge

White Paper

Stark, P., Española, M.

June 2026

Abstract

Contemporary machine intelligence is dominated by a single representational commitment: knowledge is stored as the parameters of a large, densely connected neural network, acquired through gradient descent and frozen at the close of training. This commitment has produced remarkable capability, yet it carries four structural liabilities. The resulting models are opaque, they cannot be edited without retraining, they suffer catastrophic forgetting when updated, and both their training and their operation are costly. This paper proposes an alternative architecture in which knowledge is decomposed into a federation of discrete, independently addressable compartments. Each compartment holds a fragment of knowledge expressed as an explicit equation rather than as a distributed pattern of weights. Fragments are individually replaceable, and they are revised through a verification gate that admits a change only after the incoming information has been checked, rather than through a global optimization pass. We address the hardest case for any symbolic scheme, the graded and context-dependent regularities of natural language, by representing them as explicit relational preferences whose magnitudes reside in named, editable parameter slots. We argue that the content of these preferences is necessarily acquired from observed usage, while their representation and update can be made explicit and local. We catalogue the advantages of the design, identify its principal open problem, namely the combinatorial growth of context-conditioned relations, and outline a research agenda. The contribution is a proposed architecture and a direction, not a completed system.

Keywords: knowledge representation, neuro-symbolic systems, model editing, modularity, belief revision, language modeling.

1. Introduction

The transformer architecture (Vaswani et al., 2017) and its descendants have established a near-universal template for machine intelligence. Knowledge, in this template, is whatever pattern of real-valued parameters minimizes a predictive loss over a training corpus. The approach scales: capability has improved with little more than additional parameters, additional data, and additional compute. The empirical success is not in dispute.

What is increasingly in dispute is whether the template's representational choices are the right ones for systems that must be maintained, audited, and corrected over time. Four liabilities recur. First, the models are opaque. A fact such as the capital of a country is not located anywhere inspectable; it is distributed across millions of parameters, and the system cannot explain why it holds the belief it does. Second, the models cannot be edited cleanly. Correcting a single fact, or removing a piece of stale information, conventionally requires retraining or fine-tuning, with no guarantee that the change remains localized. Third, updating a model tends to damage unrelated competence, the phenomenon of catastrophic forgetting. Fourth, both training and inference are costly in energy and capital, and the cost grows with scale.

It is worth recalling that biological cognition does not appear to pay these costs in the same way. The human brain supports comparable and broader competence on roughly twenty watts, and it does so with pronounced functional modularity and predominantly local learning: a synapse changes as a function of the activity of the two neurons it connects, without a global backward pass over the entire system. The brain is therefore an existence proof that representational and energetic profiles very different from the dense weight matrix are achievable. We take this as motivation rather than as a blueprint.

This paper makes a single architectural move. We separate two things that the prevailing template conflates: the statistical acquisition of knowledge from experience, and the representation and update of that knowledge once acquired. We retain the first where it is unavoidable, and we replace the second. Concretely, we propose that knowledge be held in discrete compartments, each owning a fragment expressed as an explicit equation, each independently replaceable, and each revised through a verification gate rather than through gradient descent.

The contributions of this paper are fourfold. We state a compartmental, equation-based representation of knowledge together with its update mechanism. We give a treatment of graded, context-dependent linguistic knowledge, the part that most resists symbolic encoding, in terms of explicit relational preferences with exposed magnitudes. We separate, with care, what the design changes, namely representation and update, from what it does not change, namely the empirical origin of the knowledge. And we analyze the principal obstacle to the design, the combinatorial growth of context-conditioned relations, alongside a candidate research agenda.

2. Background and Related Work

The tension between distributed and symbolic representations is older than the present generation of models. The connectionist tradition, descending from the formal neuron of McCulloch and Pitts (1943), holds that competence emerges from the adjustment of many simple connection strengths. The symbolic tradition holds that knowledge is best represented as explicit structures over which rules operate. The field has oscillated between these poles. The architecture proposed here is, in effect, an attempt to recover the maintainability of the symbolic pole without surrendering the empirical grounding that gave the connectionist pole its decisive victories.

Several bodies of work prefigure components of the proposal. Minsky's Society of Mind (1986) advanced the view that intelligence is produced not by a single uniform mechanism but by a federation of specialized agents, each narrow, each interacting through defined interfaces. Fodor's Modularity of Mind (1983) argued, from a different direction, for domain-specific, informationally encapsulated processing modules. The compartments of the present design are a computational reading of this lineage: bounded units that each own a fragment of knowledge and expose an interface for querying and revising it.

On the statistical side, before neural language models the dominant approach was the n-gram model, which estimates the probability of a token from explicit counts of its preceding context. These models were limited by data sparsity, since most long contexts are never observed. Recent work has revived count-based modeling at scale: suffix-array methods such as Infini-gram (Liu et al., 2024) compute unbounded n-gram statistics over corpora of a trillion tokens with no training step and with immediate updatability. This demonstrates that a substantial portion of linguistic regularity can be captured by indexing rather than by gradient descent, a point on which the present proposal relies when it speaks of acquiring preferences by counting usage.

The neuro-symbolic research program (d'Avila Garcez and Lamb, 2023) seeks explicitly to combine learned perception with symbolic reasoning, and provides the broad frame within which this architecture sits. Vector-symbolic, or hyperdimensional, computing (Kanerva, 2009) offers a complementary mechanism in which structured knowledge is composed through algebraic operations on high-dimensional vectors rather than learned through optimization. It is one of the candidates considered below for the fuzzy layer.

The editability that the present design treats as primary is already being pursued inside dense networks. Locating-and-editing methods such as ROME (Meng et al., 2022), and its mass-editing successor MEMIT (Meng et al., 2023), identify the parameters responsible for a specific factual association and rewrite them directly. That such methods are necessary, and that they are difficult, is itself evidence for the diagnosis offered here: the weight matrix is a poor substrate for targeted change. Where these methods labor to approximate editability within an unsuitable representation, the present design seeks to make editability native.

Finally, the update mechanism proposed below has a formal antecedent in the truth-maintenance systems of Doyle (1979) and in the theory of belief revision associated with Alchourron, Gardenfors, and Makinson (1985). Both address precisely the question of how a body of held beliefs should change, consistently, when new information arrives. The verification gate described in Section 4 is a practical instance of this long-standing concern.

3. The Ninety-Five Percent Problem

Any proposal to make machine knowledge explicit must confront the fact that much of what a competent language user knows is not naturally stated as a fact. It is useful to divide knowledge into two regimes. The first is crisp: discrete, propositional, and verifiable. That a particular city is the capital of a particular country is crisp knowledge; it can be written as a relation, checked against a source, and corrected if found wrong. The second regime is graded and contextual, and it constitutes the larger share of fluency. We refer to it informally as the ninety-five percent.

Consider lexical selection. A fluent writer prefers strong tea to powerful tea, yet prefers powerful engine to strong engine, while accepting both strong argument and powerful argument. No rule of grammar distinguishes these cases, and both candidate words appear in any dictionary. The preference is real, it is learned, and it is conditioned on context. The classic demonstration that grammaticality is insufficient is Chomsky's observation (Chomsky, 1957) that a sentence can satisfy every syntactic rule and yet remain meaningless. Fluency lives in selectional and collocational knowledge that grammar does not encode.

Two features of this regime make it the central difficulty. First, the relevant comparisons are context-dependent: the relation between two candidate expressions can reverse as the surrounding text changes, so the knowledge is not a fixed table of pairwise preferences but a function of context. Second, the relevant comparisons are frequently graded rather than binary: competing expressions may both be acceptable, differing only in connotation or intensity, so that an adequate representation must record not merely which expression is preferred but by how much.

A representation that captured only crisp knowledge would therefore capture the smaller and easier part of competence while leaving the larger and harder part untouched. The architecture proposed below is designed so that crisp knowledge is represented naturally and well, and so that graded contextual knowledge is represented explicitly rather than being relegated to an opaque component. Whether that explicit representation can be made compact enough to be practical is the principal open question, and we return to it in Section 6.

4. The Proposed Architecture

We describe the architecture in five parts: the compartment as the unit of knowledge, the use of equations in place of weights, the verified-rewriting update mechanism, the representation of graded preference, and the separation of acquisition from storage.

4.1 Compartments as units of knowledge

The system is organized as a federation of compartments. Each compartment is a bounded, independently addressable unit that owns a fragment of the system's knowledge and exposes a defined interface for querying and revising that fragment. Compartments are the locus of modularity, and they correspond by design to the functional specialization observed in biological cognition and argued for in the modular tradition. The key property is independence. A compartment can be inspected, replaced, or removed without reference to the internal contents of any other compartment, provided its interface contract is preserved. This independence is what later delivers both clean editability and freedom from catastrophic forgetting.

4.2 Equations in place of weights

Within a compartment, a fragment of knowledge is expressed as an explicit equation or function, not as a distributed pattern of connection strengths. The distinction is representational rather than merely notational. In a weight matrix, the assertion that one expression is preferred to another exists only as an emergent consequence of many parameters, and is neither separately readable nor separately writable. Expressed as an equation, the same assertion becomes a discrete object: it can be read by a human, checked by a procedure, and overwritten in place. The cost of this explicitness is that the designer must commit to a vocabulary of functional forms, and the expressiveness of the representation is bounded by that vocabulary. The benefit is that knowledge becomes a first-class, inspectable artifact.

4.3 Verified rewriting

Knowledge in the system changes through a verification gate rather than through gradient descent. When new information arrives, it does not directly adjust any stored value. Instead it proposes a candidate rewrite of one or more equations. The candidate is admitted only if it passes verification: a check for internal consistency against the contents of related compartments, and, where applicable, a check against external evidence. Admitted rewrites are local to the compartments they touch. This mechanism is the practical descendant of truth maintenance and belief revision, and it has two consequences that gradient descent does not offer. First, the provenance of every change is recorded, because a change is an event with a justification rather than an increment distributed across a training run. Second, the system can decline to incorporate information that fails verification, which provides a direct lever against the admission of falsehoods.

4.4 Representing graded preference

The graded, context-dependent regime of Section 3 is represented as explicit relational preference. For a given context and a pair of candidate expressions, the system holds a relation stating that one is preferred to the other, together with a magnitude that records the strength of that preference. The magnitude is not eliminated; it is precisely the continuous value that a weight matrix would have buried. The architectural claim here is narrow and, we believe, defensible: the magnitude is stored in a named, editable slot rather than in an opaque parameter. A graded preference therefore remains graded, but its gradation is exposed and addressable. An auditor can read it, a maintainer can change it, and a verification procedure can flag it for review.

4.5 Acquisition versus storage

It is essential to be precise about what this design changes and what it does not. The content of the preferences, which expressions are in fact preferred in which contexts, is empirical. It can only be obtained by observing how language is actually used, whether by counting occurrences in a corpus or by some equivalent procedure over experience. The architecture does not, and cannot, conjure this content from first principles. A claim that an expression is not used is, on inspection, a claim about an observed body of usage. What the architecture changes is therefore not the empirical origin of the knowledge but its representation and its update: the learned regularities are stored as explicit, local, editable equations rather than as a monolithic and frozen parameter set, and they are revised through verified rewriting rather than through retraining. Acquisition is decoupled from representation, and this decoupling is the conceptual core of the proposal.

5. Properties and Advantages

The design's advantages follow from the independence of compartments and the explicitness of their contents, and they correspond directly to the four liabilities identified in the introduction.

The system does not suffer catastrophic forgetting in the ordinary sense. Because a revision is local to the compartments it touches, correcting or replacing one fragment leaves unrelated fragments untouched by construction. The interference that arises when a single parameter set is reused for all knowledge simply does not occur when knowledge is partitioned.

Knowledge is hot-swappable. A fragment can be replaced in place without a training run. This is the property that conventional models lack and that dedicated editing methods labor to approximate within the weight matrix. Here it is a native consequence of the representation rather than a procedure imposed upon a resistant substrate.

The system carries provenance and supports verification. Because knowledge is admitted only through a gate that checks it, and because each change is an event with a recorded justification, the system can report why it holds a given item and on what evidence. This is a precondition for deployment in any setting where auditability is required, and it is exactly the property that an opaque parameter set cannot provide.

The system is interpretable. A compartment's contents can be opened and read. The assertion responsible for a given behavior is a discrete, legible object rather than a pattern to be reverse-engineered after the fact. Interpretability is not retrofitted; it is the default state of an explicit representation.

Finally, the system supports incremental, lifelong revision. New knowledge is added by introducing or rewriting compartments, an operation whose cost is proportional to the change rather than to the size of the whole. The expensive, all-at-once retraining cycle is replaced by continuous, local maintenance. These properties are attractive precisely in the regimes where current models are weakest: domains that demand correction, audit, and stability under update. The advantages are structural rather than incidental. They are also purchased at a price, which the next section makes explicit.

6. Open Challenges

The architecture has one principal difficulty, from which several others follow, and intellectual honesty requires that it be stated plainly.

The principal difficulty is the combinatorial growth of context-conditioned relations. Because the graded preferences of Section 4 are conditioned on context, and because context can reverse them, the system cannot store a single relation per pair of expressions. It must in principle store a relation for each relevant context. The number of relevant contexts in natural language is enormous, and a naive enumeration of context-conditioned relations would approach the information content of the very weight matrix the design seeks to replace, now written out at length. The design does not escape the quantity of knowledge that fluency requires; it changes the form in which that quantity is held. For the architecture to be practical rather than merely possible, the set of equations must be compressed: regularities must be factored, contexts must be organized into a hierarchy or abstracted into reusable conditions, and shared structure must be exploited so that the representation grows far more slowly than a flat enumeration would. Whether such compression can be achieved while preserving the editability that motivates the design is, at present, an open question. It is the question on which the proposal stands or falls.

Three further challenges follow. First, gradation re-enters as parameters. As Section 4.4 concedes, the continuous magnitudes are exposed rather than removed; the architecture is therefore a symbolic structure with numeric content in named slots, not a structure free of continuous values. This is a virtue for auditability, but it should not be mistaken for the elimination of statistics. Second, acquisition remains empirical and data-intensive. The content must still be learned from observed usage, and the cost of that learning is not avoided, only relocated out of the representation. Third, verification at scale is itself hard. Defining and automating a verification gate is straightforward for crisp propositional knowledge, but considerably more delicate for graded preferences, where the standard of correctness is usage rather than truth, and where the appropriate check is not obvious. A practical system would require a principled account of what it means to verify a graded, context-dependent preference.

None of these challenges is, on present evidence, a proof of impossibility. Each is a research problem, and the first is the one that matters most. A serious program of work on this architecture would be, in large part, a program of work on the compression of the equation set.

7. Discussion

The realistic form of this architecture is neither a return to hand-authored symbolic systems nor a continuation of the monolithic weight matrix, but a hybrid that takes from each what it does well. The structure of knowledge is explicit, modular, and editable; the magnitudes within that structure are continuous, learned from usage, and exposed in named slots. One may describe the result as a symbolic skeleton with numeric content. It keeps the engineering advantages of explicit representation, namely editability, provenance, interpretability, and freedom from catastrophic forgetting, while declining to deny the empirical and partly continuous nature of the knowledge it holds. The honesty of that position is, we think, a feature rather than a concession.

The architecture is most likely to prove valuable first in settings where its advantages are decisive and its principal difficulty is contained. Domains with a large crisp component and a strong requirement for audit and correction, such as regulated professional knowledge, institutional record-keeping, and any system that must justify its outputs, fit this description well. In such domains the proportion of knowledge that is propositional and verifiable is higher, the combinatorial pressure from open-ended linguistic context is lower, and the value placed on editability and provenance is greatest. A reasonable development path would therefore begin with crisp-dominant

domains, where the compartmental representation can be exercised end to end, and would extend toward open-ended language only as methods for compressing the equation set mature.

We present this not as a finished system but as an architecture and a research agenda. The central insight is the separation of acquisition from representation, and the consequent ability to make learned knowledge explicit, local, and editable. The central obstacle is the combinatorial growth of context-conditioned relations. The work that would convert the proposal into a system is, in large part, the work of resolving that obstacle.

8. Conclusion

The dominant paradigm in machine intelligence stores knowledge as the parameters of a single large network, acquired by gradient descent and frozen at the end of training. The paradigm is powerful, and it is also opaque, difficult to edit, prone to forgetting under update, and costly. This paper has proposed an alternative in which knowledge is held in independent compartments, each fragment expressed as an explicit equation, each replaceable on its own, and each revised through a verification gate rather than through retraining. Graded, context-dependent knowledge, the part that most resists symbolic treatment, is represented as explicit relational preference with magnitudes held in named, editable slots. We have argued that this design changes the representation and update of knowledge while honestly retaining the empirical origin of its content, and that it delivers editability, provenance, interpretability, and stability under update as structural consequences. We have also identified its principal limitation, the combinatorial growth of context-conditioned relations, as the problem that determines whether the architecture can scale. The contribution is a direction and an agenda. The next step is to build the crisp-dominant case and to confront the problem of compression directly.

References

Alchourron, C. E., Gardenfors, P., and Makinson, D. (1985). On the logic of theory change: Partial meet contraction and revision functions. Journal of Symbolic Logic, 50(2).

Chomsky, N. (1957). Syntactic Structures. Mouton.

d'Avila Garcez, A., and Lamb, L. C. (2023). Neurosymbolic AI: The third wave. Artificial Intelligence Review, 56.

Doyle, J. (1979). A truth maintenance system. Artificial Intelligence, 12(3).

Fodor, J. A. (1983). The Modularity of Mind. MIT Press.

Kanerva, P. (2009). Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. Cognitive Computation, 1(2).

Liu, J., et al. (2024). Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. arXiv preprint.

McCulloch, W. S., and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics, 5.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. (2022). Locating and editing factual associations in GPT. Advances in Neural Information Processing Systems 35.

Meng, K., Sharma, A. S., Andonian, A., Belinkov, Y., and Bau, D. (2023). Mass-editing memory in a transformer. International Conference on Learning Representations.

Minsky, M. (1986). The Society of Mind. Simon and Schuster.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems 30.