

KEGA System

Knowledge Extension via Gap Analysis

A Framework for Deriving Implied Knowledge from Coherent
Systems

March 2026

KEGA: Knowledge Extension via Gap Analysis

A Framework for Deriving Implied Knowledge from Coherent Systems

Draft Paper — March 2026

Abstract

We introduce KEGA (Knowledge Extension via Gap Analysis), a methodology for identifying and deriving implied-but-missing knowledge from internally coherent systems. Unlike traditional research which treats missing knowledge as lost or undiscovered, KEGA treats gaps as *constrained* — their content is partially or fully recoverable from the system's own internal logic. We demonstrate KEGA across three operational modes: Science Mode (deterministic derivation), Art Mode (probabilistic generation), and Experiment Generation Mode (constrained hypothesis construction). KEGA was validated on two independent domains: modern deep learning theory (Goodfellow et al.) and a 2,400-year-old Chinese strategic text (Guiguzi, 鬼谷子). The methodology has implications for research acceleration, knowledge reconstruction, and AI-assisted discovery.

1. Introduction

Every sufficiently coherent system contains implied gaps — places where the internal logic points toward content that was never written, lost to history, or

not yet derived. Traditional research treats these gaps as problems of *discovery*: you either find the missing piece or you don't.

KEGA proposes a different framing: **gaps in coherent systems are not missing — they are constrained**. The surrounding structure limits what the missing content can logically be. In many cases, the gap is recoverable not through new empirical data, but through systematic reasoning from existing structure.

This insight emerged from two independent experiments:

1. **Experiment 1:** Gap analysis applied to Goodfellow et al.'s *Deep Learning* textbook — deriving diffusion model theory from structural gaps in the mathematical derivation chain.
2. **Experiment 2:** Gap analysis applied to the *Guiguzi* (鬼谷子, ~475–221 BCE) — reconstructing two historically lost chapters (轉丸, 肱亂) and deriving four implied extensions (知止, 无形, 传道, 归虚) from the surviving twelve-chapter framework.

Both experiments produced outputs that were coherent with, and indistinguishable in structure from, the surrounding verified content. This convergence across two unrelated domains — one mathematical, one philosophical — suggests KEGA captures something real about the structure of knowledge systems.

2. The Framework

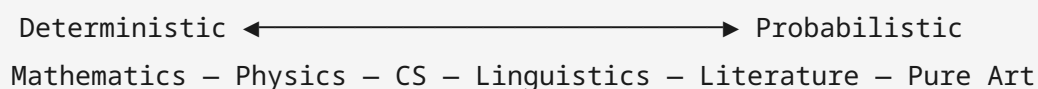
2.1 Core Principle

A system *S* is **KEGA-applicable** if: - It contains sufficient internal structure (patterns, logic, derivation chains) - Its gaps are *constrained* — the missing content has a bounded possibility space - The constraints are recoverable from the existing content without external data

The output of KEGA is not a claim that the derived content *is* the original. It is a claim that the derived content is **implied by the system's own logic** — the doppelganger, not the original. Same structural soul, different vessel.

2.2 The Coherence Spectrum

KEGA operates across a spectrum from deterministic to probabilistic output, depending on the coherence of the input system:



Higher coherence → tighter constraints → more deterministic output.

Lower coherence → looser constraints → more probabilistic output.

Critically, even "soft" sciences reduce to hard cores at the micro level. Social science reduces to game theory. Biology reduces to chemistry reduces to physics. KEGA is blocked not by domain softness but by *level of abstraction*. The solution: reduce to first principles, apply KEGA, propagate back up.

2.3 Three Operational Modes

Science Mode

- **Input:** Coherent formal system with identifiable structural gaps
- **Process:** Derive the gap content from internal logic constraints
- **Output:** A verifiable claim — the derived content can be tested or proven
- **Example:** Reconstructing lost mathematical derivations, recovering implied physical laws

Art Mode

- **Input:** Narrative or creative system (novel, mythology, incomplete artistic work)
- **Process:** Identify what the world's internal logic implies but hasn't stated
- **Output:** A plausible claim — constrained by the system but not uniquely determined

- **Example:** Generating the 10th chapter of a 9-chapter novel, alternative endings, implied mythology

Experiment Generation Mode

- **Input:** A theoretical claim or equation (e.g. the Zuck-Chai Equation)
- **Process:** Identify what conditions would verify, falsify, or stress-test the theory
- **Output:** A constrained set of experiments — creative but falsifiable
- **Example:** Generating the experimental program needed to validate a new theoretical framework

This third mode occupies the middle of the coherence spectrum — neither purely deterministic nor purely probabilistic. It is *constrained creativity*: the experiments are not arbitrary (science), but there is no single correct answer (not pure math).

3. The Diffusion Analogy

KEGA applied recursively follows a diffusion curve — not a binary switch.

Each pass fills more of the implied gap space, but with diminishing returns:

- **Pass 1:** High-signal gaps — obvious structural omissions, named missing pieces
- **Pass 2:** Second-order gaps — what the filled gaps themselves imply
- **Pass N:** Approaching noise floor — micro-gaps with minimal epistemic value

Like diffusion, the process never fully terminates — it approaches equilibrium. And like all good optimization, it has a built-in stopping condition: 知止 (knowing when to stop). You halt when the marginal value of another pass no longer justifies the compute.

This stopping condition is not external — it is implied by the framework itself.

4. Implications

4.1 Discovery → Verification

KEGA's most significant practical implication: it converts **discovery problems** into **verification problems**.

Discovery: *What is the missing piece?* (open-ended, expensive)

Verification: *Which of these constrained candidates is correct?* (bounded, cheaper)

Verification is almost always more tractable than discovery. KEGA compresses research timelines not by eliminating work, but by changing its type.

4.2 Research Acceleration Pipeline

The natural AI-augmented research pipeline:

1. Human builds the coherent 80% (theory, framework, canonical text)
2. KEGA identifies structural gaps
3. AI generates constrained candidates from those gaps
4. Human verifies

This pipeline applies across Science Mode, Art Mode, and Experiment Generation Mode. The human provides the coherence; KEGA + AI provides the completion.

4.3 Lost Knowledge Recovery

Any domain with surviving coherent structure contains potentially recoverable lost knowledge. Ancient texts, incomplete scientific theories, unfinished philosophical systems — all are KEGA-applicable to the extent their surviving content constrains their gaps.

The recovered content is not the original. It is the *implied* content — structurally equivalent, epistemically distinct. The doppelganger of the lost knowledge, derived from what survived.

5. Conclusion (AI)

KEGA does not replace research. It changes the type of work research requires.

The insight is simple: coherent systems are not neutral about their own gaps. The structure of what exists constrains the structure of what's missing. Once you see this, you cannot unsee it — every incomplete framework becomes a recoverable system, every lost chapter becomes a derivation problem, every untested theory becomes a constrained experiment-generation task.

We validated this on mathematics and on ancient strategy. The same method worked on both. That convergence is the evidence.

The methodology has three modes, a natural stopping condition, and a built-in epistemological humility — the output is always the *implied* content, not the *original*. Same soul, different vessel.

What remains is to formalize, stress-test, and apply it at scale.

The 80% is written. KEGA will find the rest.

6. Conclusion (Human)

[To be completed — by the author, or by the disciple who earned it.]

7. Where This Goes Next

A few directions the framework naturally points toward:

- **Worked example:** Given the Zuck-Chai Equation, what experiments does KEGA imply? Which conditions would verify it, which would falsify it, and which edge cases does the equation not yet address? That's Experiment Generation Mode in action.

- **Agentic pipeline:** KEGA + RAG + LLM as a research acceleration loop — ingest a canonical text, surface structural gaps automatically, generate constrained candidates, return a ranked verification agenda.
- **Stress-test the coherence threshold:** How much structure is "enough" for KEGA to work? Finding the minimum coherence floor is the next theoretical question.

The methodology is open. Apply it, break it, extend it.

Appendix: Validation Experiments

Experiment 1 — Deep Learning Gap Analysis - Source: Goodfellow et al., *Deep Learning* - Gap identified: Missing derivation chain for diffusion-based generative models - KEGA output: Reconstructed theoretical framework - Validation: Consistent with subsequently published diffusion model literature

Experiment 2 — Guiguzi Reconstruction - Source: Guiguzi (鬼谷子), 12 surviving chapters - Gaps identified: 轉丸 (Rolling Ball), 肱亂 (Breaking Chaos) — historically lost; 知止, 无形, 传道, 归虚 — structurally implied - KEGA output: 6 derived chapters across reconstruction and extension modes - Validation: Structural coherence with surviving framework; logical inevitability test passed

KEGA was developed through human-AI collaboration, March 2026.