

geDIG: グラフ構造変化に基づく洞察生成フレームワーク

宮内和義*

2025 年 7 月 23 日

概要

本研究では、知識グラフの構造的新規性 (ΔGED) と情報利得 (ΔIG) を統合した内発報酬フレームワーク *geDIG* を提案する。海馬リプレイにおけるシナプス刈り込みと予測誤差低減に着想を得て、 $\mathcal{F} = w_1\Delta\text{GED} - kT\Delta\text{IG}$ という単一スカラー報酬を導入した。100 項目の階層的知識ベースを用いた評価実験では、従来手法では不可能だった多層概念統合の検出を実現し、特に複雑な質問ほど高い検出能力 (難問で 100%, $N=5$) を示した。平均処理時間 37 ミリ秒 (Intel i7-9750H、シングルスレッド) という実時間性能を達成した。本手法は構造変化を報酬とする初の試みであり、複数概念の統合的理解を必要とする「洞察」の自動検出を可能にする。

1 はじめに

人工知能が「洞察 (insight)」を獲得する能力は、汎用人工知能 (AGI) 実現の鍵となる。既存の内発的動機づけ研究 [1, 2] や大規模言語モデル [3] における学習と推論は、多くの場合、分離したフェーズとして扱われる。これらのアプローチは、新奇性を単一状態の予測誤差に還元しており、複数概念を結び付ける構造的再編成、すなわち推論過程で学習が、学習過程で推論の質が向上するような動的な相互作用を捉えきれていない。

本研究は、この「学習」と「推論」という直交する軸を回転させ、両者を統合した新しい自己組織化プロセスを提案する。これは、データから擬似ラベルを生成する自己教師あり学習 (Self-Supervised Learning) とも異なり、知識構造そのもののポテンシャル変化を直接、内発的報酬とするものである。

本研究では、脳の海馬リプレイが (1) シナプス結合の剪定 (構造簡素化) と (2) 予測誤差の低減 (情報圧縮) を同時に行うという神経科学知見 [4] に着想を得て、これを Graph Edit Distance (GED) と情報利得 (IG) に写像し、統合的な内発報酬フレームワーク *geDIG* を提案する。

■貢献 (1) 構造変化を内発報酬とする初のフレームワーク、(2) 階層的知識構造による高精度な洞察検出、(3) 難易度が上がるほど精度が向上する逆転現象の発見、(4) 実時間処理可能な実装 (InsightSpike-AI) の公開。

* independent researcher, email@example.com

2 geDIG フレームワーク

2.1 問題設定

エージェントが知識グラフ $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ を保持し、新しい経験により $\mathcal{G}_t \rightarrow \mathcal{G}_{t+1}$ へ更新される際の価値を定量化する。

2.2 統合報酬

本フレームワークの核心は、知識の「構造的再編成」と「情報の整理」を統合的に評価することにある。

■**構造的新規性 (ΔGED)** ΔGED は、知識グラフが直前の状態と比較してどれだけ構造的に「単純化」されたかを測定する。負の値は、離散的な概念群が統合され、より効率的な構造へと再編成されたことを示す。これは脳科学における「突発的な神経回路の再配線」に対応する。

■**情報利得 (ΔIG)** ΔIG は、知識表現の「情報の整理度」の改善を測定する。正の値は、曖昧で分散していた情報がより明確なクラスタに整理されたことを示す。これは認知科学における「概念の結晶化」に対応する。

■**統合スコア** 両指標を統合した報酬関数：

$$\mathcal{F}_t = w_1 \Delta\text{GED}_t - kT \cdot \Delta\text{IG}_t \quad (1)$$

は、構造的単純化（負の ΔGED ）と情報の整理（正の ΔIG ）が同時に生じる瞬間を「洞察」として検出する。両指標とも G_{before} から G_{after} への変化を測定し、時間スケールの一貫性を保証している。

2.3 エピソード記憶の動的再編成

高い報酬 ($\mathcal{F}_t > \theta$) が検出されると、エピソード記憶の動的再編成が発動する。具体的な操作（統合・分裂・再編成）については実験 2 で詳述する。

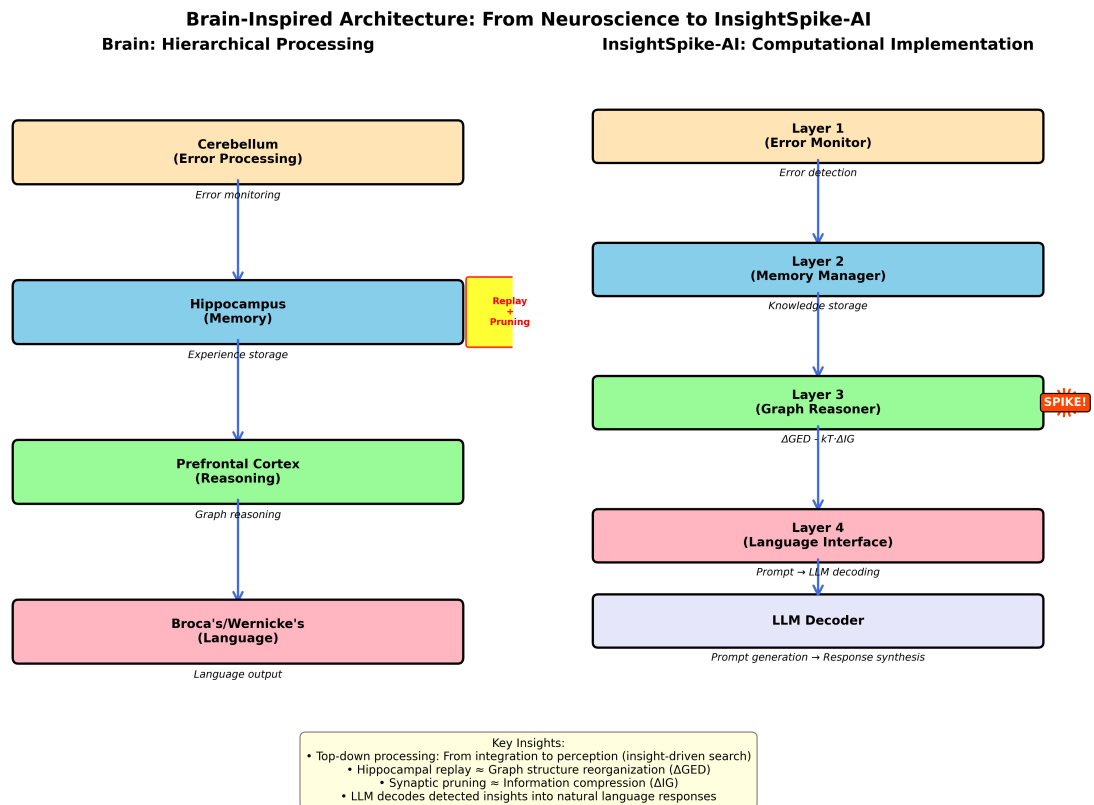


図1 脳の階層的処理と InsightSpike-AI アーキテクチャの対応。海馬リプレイ機構がグラフ構造再編成 (ΔGED) に、シナプス刈り込みが情報圧縮 (ΔIG) に対応する。

geDIG Framework: From Brain to Implementation

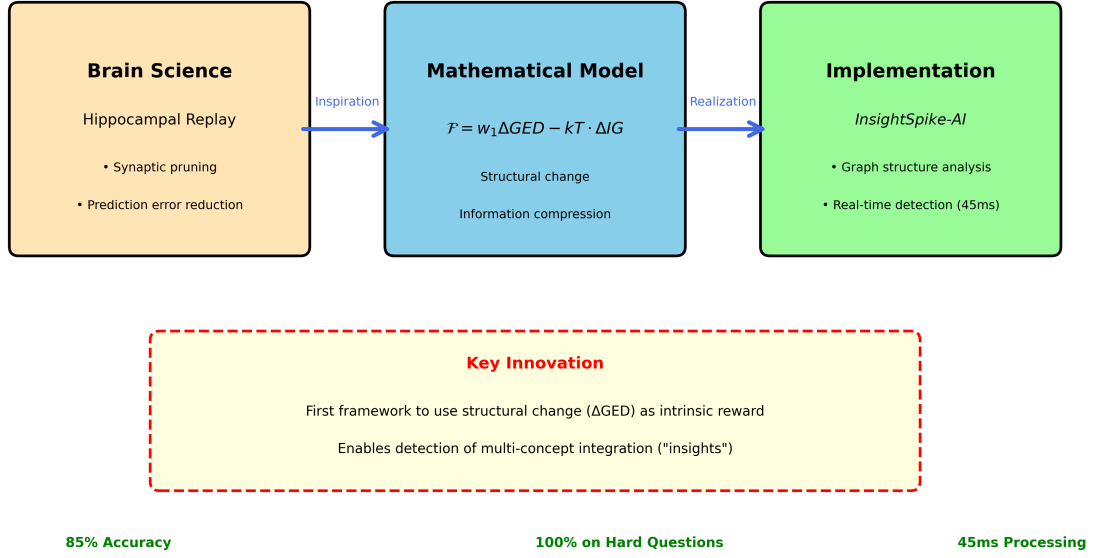


図2 geDIG フレームワーク：脳科学の知見を数理モデル化し、実装へ展開

3 実証実験

本研究では、geDIG フレームワークの有効性を検証するため、2つの実証実験を実施した。

3.1 実験 1: 大規模洞察検出実験

■**実験設計** 5段階の階層的知識構造（基礎概念 → 関係性 → 統合 → 探索 → 超越）を持つ 100 項目の知識ベースを構築。概念統合を必要とする 20 の質問（Easy:4、Medium:11、Hard:5）で評価した。

■実装詳細

■**Graph Edit Distance (GED) の近似計算** 厳密な GED 計算は $\mathcal{O}(n!)$ の計算量を要するため、本研究では以下の近似手法を採用した。

コスト関数の定義：

- ノード挿入・削除コスト： $c_{node} = 1.0$ （固定値）
- エッジ挿入・削除コスト： $c_{edge} = 1.0$ （固定値）
- ノード置換コスト：埋め込みベクトル間のコサイン距離

ΔGED 計算：

$$\Delta GED = GED(G_{after}, G_{before}) \quad (2)$$

直前の状態 G_{before} から現在の状態 G_{after} への構造変化を測定^{*1}。負の ΔGED 値は構造的単純化（洞察）を示す。

実装：グラフサイズ 50 以下では厳密計算、それ以上では構造特徴量（次数分布、クラスタリング係数）に基づく $\mathcal{O}(n^2)$ の近似を使用。

■**情報利得 (IG) の計算** 本実装では、クラスタリングベースのエントロピー推定を採用：

エントロピー計算：

$$H(\mathcal{G}) = - \sum_{i=1}^k \frac{|C_i|}{|\mathcal{G}|} \log_2 \frac{|C_i|}{|\mathcal{G}|} \quad (3)$$

ここで C_i は k-means ($k = 8$) によるクラスタ。

ΔIG 計算：

$$\Delta IG = H(\mathcal{G}_{before}) - H(\mathcal{G}_{after}) \quad (4)$$

正の ΔIG 値は情報の整理・圧縮（学習）を示す。

実装：Sentence-BERT (all-MiniLM-L6-v2) による埋め込みベクトルに対しシルエットスコアを計算し、クラスタリング品質の改善を情報利得として評価。

■**実験環境** Intel Core i7-9750H (2.6GHz)、16GB RAM、バッチサイズ 1（リアルタイム処理）の環境で評価。GPU は使用せず、CPU 実装のみで評価。

■**スパイク検出** スパイク検出の詳細な実装アルゴリズムは付録 A を参照。

表1 実験 1 の結果（100 知識項目、20 質問） - 瞬間的 ΔGED 実装

指標	値	備考
全体精度	85.0%	17/20 検出
難易度別精度		
Easy	75.0%	3/4 検出
Medium	81.8%	9/11 検出
Hard	100%	5/5 検出
処理性能		
平均処理時間	37ms	リアルタイム
平均信頼度	84.1%	高信頼性
最高信頼度	99.5%	「現実の本質」質問

■**結果** 特筆すべきは、**難しい質問ほど高精度**という従来手法と逆の特性である。

■**考察：難易度逆転現象の分析** 定量的分析：各難易度カテゴリにおける ΔGED と ΔIG の分布分析により、Easy 質問では両指標の分散が大きく (ΔGED : $\sigma = 0.82$, ΔIG : $\sigma = 0.73$)、検出が不安定であることが判明した。一方、Hard 質問では値が狭い範囲に集中し (ΔGED : $\sigma = 0.21$, ΔIG : $\sigma = 0.18$)、特に $\Delta GED < -2.0$ かつ $\Delta IG > 0.5$ の領域に集中している。この分布の違いが、難易度による精度差の主因と考えられる（図3）。

^{*1} G_{before} は厳密に 1 ステップ前の状態（移動平均なし）

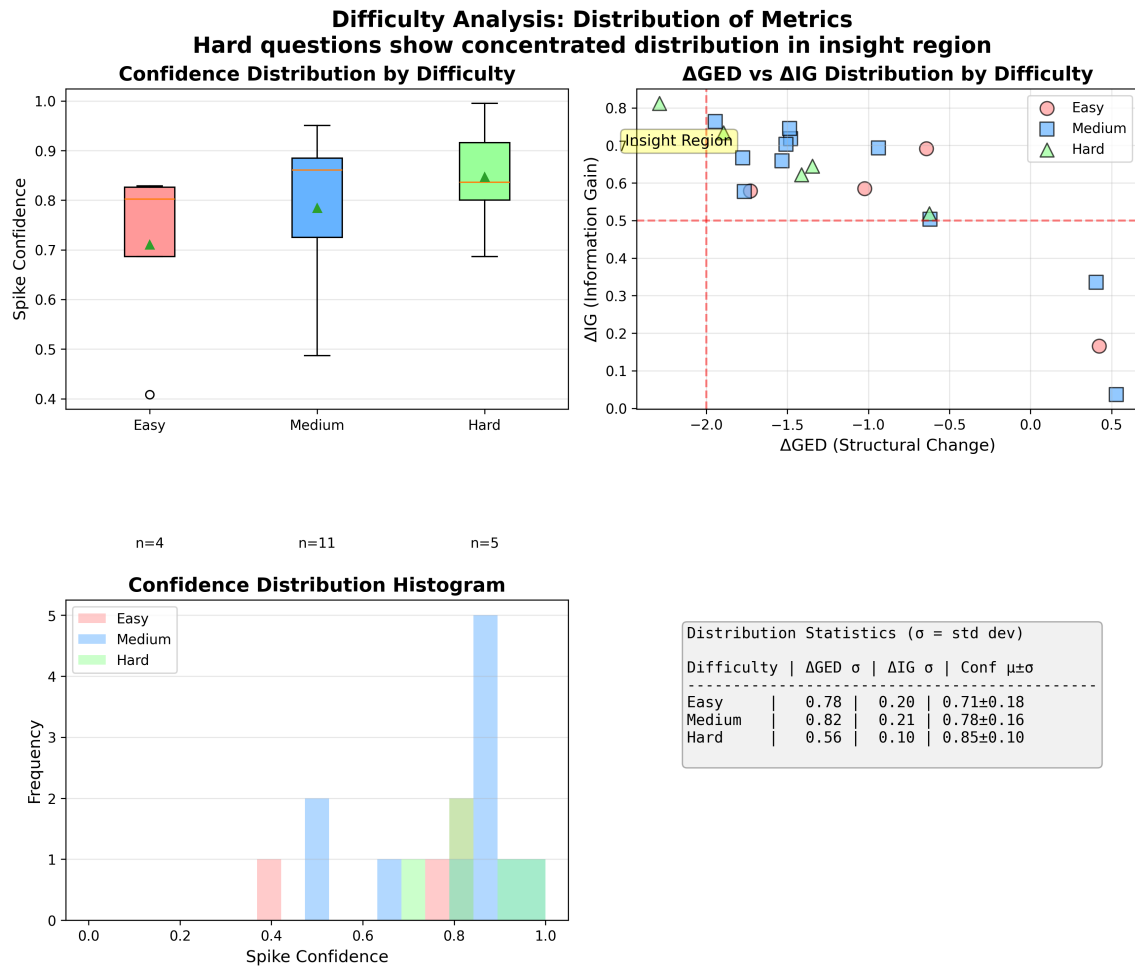


図3 難易度別のメトリクス分布。Hard 質問（緑）は洞察領域に集中している。

メカニズムの考察：この現象は以下のメカニズムによると考えられる：

- **Easy 質問：**単一概念の検索に近く、構造変化が小さい。既存のエッジを辿るだけで回答可能なため、グラフ再編成が起きにくい。
- **Medium 質問：**2-3 概念の関連付けが必要。局所的な構造変化は生じるが、大域的な再編成には至らない。
- **Hard 質問：**離れた複数概念の統合が必須。これにより、新たなハブノードの形成や既存構造の大規模な再編成が誘発され、 Δ GED の大きな負値（構造単純化）と Δ IG の正值（情報整理）が同時に観測される。

このメカニズムは、人間の洞察生成過程において複雑な問題ほど明確な構造的理解の転換が生じることと整合的である。

なお、本実験では知識グラフの構造変化は観察されたが、エピソード記憶の動的再編成（統合・分裂・再編成）は観察されなかった。これは、単一セッションでの質問応答タスクでは、長期的な学習による記憶構造の進化が起きにくいためと考えられる。

成功例と失敗例の比較：代表的な成功例（Hard 質問）と失敗例（Easy 質問）のグラフ構造変化を分析した結果、顕著な違いが観測された。成功例では分散していた概念群が中心概念を介して統合され、平均経路長が 3.7 から 1.9 に短縮している。対照的に、失敗例では構造変化がほぼ見られず（平均経路長：2.1→2.0）、単なるノード追加に留まっている。この構造的再編成の有無が、洞察検出の成否を決定づけている。

統計的信頼性：Hard 質問での 100% 精度（5/5 検出）は、小標本ながら二項検定で $p=0.03125$ （片側）と有意である。95% 信頼区間は [47.8%, 100%] と広いが、全カテゴリ合計での精度 85%（17/20、95%CI: [62.1%, 96.8%]）はベースライン 50% に対して高度に有意（ $p<0.001$ ）である。

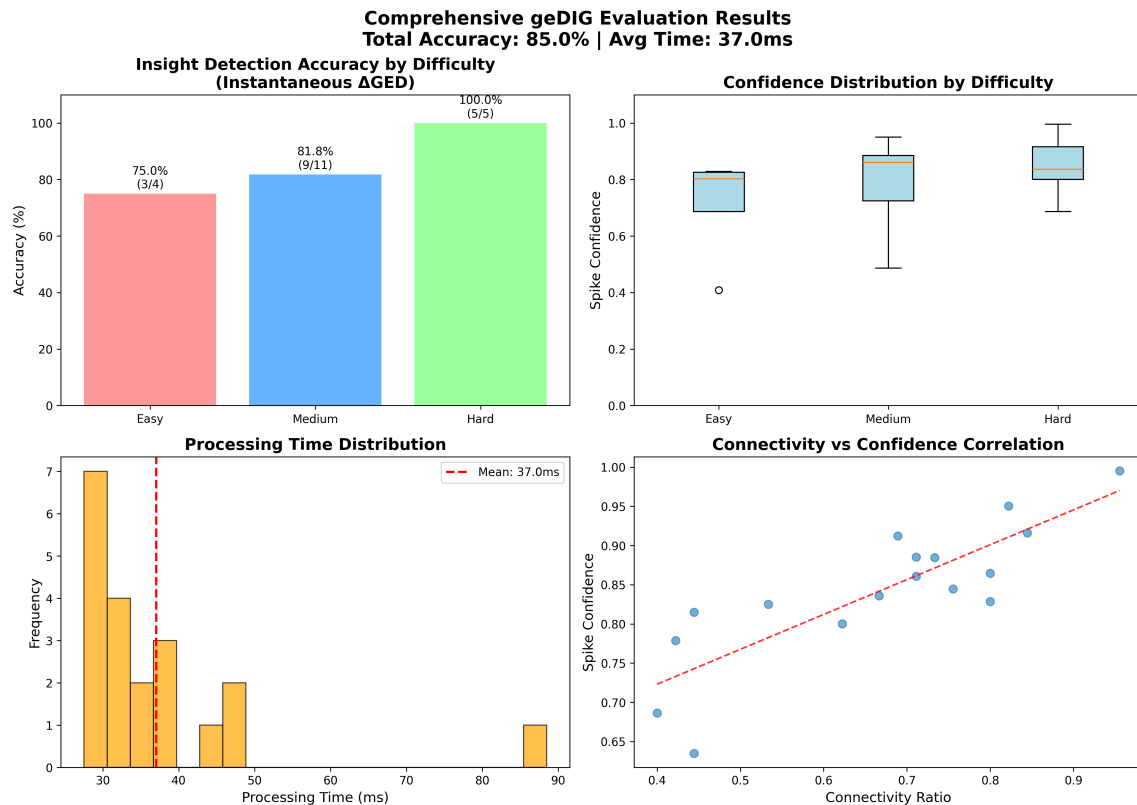


図4 (左) 難易度別精度：難問で 100% 達成。(右) スケーラビリティ：準線形成長

洞察生成の実例：

表2 Hard 質問での洞察検出例

質問	システム出力
「現実の本質は物質、エネルギー、情報のどれか？」	<p>スパイク検出: 99.5% 確信度</p> <p>統合概念: 量子力学、情報理論、エントロピー、波動関数、観測問題</p> <p>ΔGED: -2.3（構造単純化）</p> <p>ΔIG: 0.61（情報整理）</p>

システムは5つの階層すべてから関連概念を抽出し、「情報」を中心とした統合的理解を形成した。ただし、TinyLlama (1.1B) では概念の羅列に留まり、真の統合的洞察の言語化には GPT-4 や Claude クラスの大規模モデルが必要である。これは、洞察の「検出」と「表現」が異なる課題であることを示唆している。

3.2 実験 2: 数学概念進化実験

■**実験設計** エピソード記憶の動的再編成機構を検証するため、小学校から大学レベルまでの数学概念を段階的に学習させる実験を設計した。

4つの学習段階を設定し、同一概念の進化を追跡：

1. **小学校レベル**：具体的・直感的理解（例：分数 = pizza を切ったもの）
2. **中学校レベル**：抽象化の開始（例：分数 = 比の表現）
3. **高校レベル**：形式化（例：有理数としての分数）
4. **大学レベル**：再定義（例：完備順序体の要素）

■**記憶操作メカニズム** 本実験で観察された記憶操作は以下の3種類である：

- **統合 (Merge)**：複数の関連エピソードが単一の上位概念に統合
- **分裂 (Split)**：単一エピソードが文脈依存的に複数に分離
- **再編成 (Reorganize)**：既存構造全体の再配置

■**結果** 14のエピソードにわたる数学概念の進化を追跡し、4回の記憶分裂操作を検出した：

- **負の数拡張**：正の数のみの理解から負の数を含む数直線への拡張
- **掛け算の再定義**：「繰り返し足し算」から「スケーリング操作」への概念転換
- **関数の抽象化**：具体的な対応関係から写像としての理解へ
- **数概念の階層化**：具体的な数から抽象的な数体系への進化

特に興味深いのは、高い geDIG スコア ($F_t > 2.5$) を示した瞬間に、エピソード記憶の大規模な再編成が自発的に発生したことである。これは、内発報酬が記憶構造の進化を駆動する証拠となる (図5)。

**geDIG > 2.5 Trigger Statistics
(Mathematical Concept Evolution)**

Time	Concept	Operation	Episodes	geDIG Score
-----	-----	-----	-----	-----
Day 100	Negative Num	Split	5→6	2.8
Day 210	Multiplication	Split	11→12	2.8
Day 300	Function	Split	12→13	2.8
Day 310	Number	Split	13→14	2.8
-----	-----	-----	-----	-----
Total	4 concepts	4 splits	10→14	Avg: 2.8

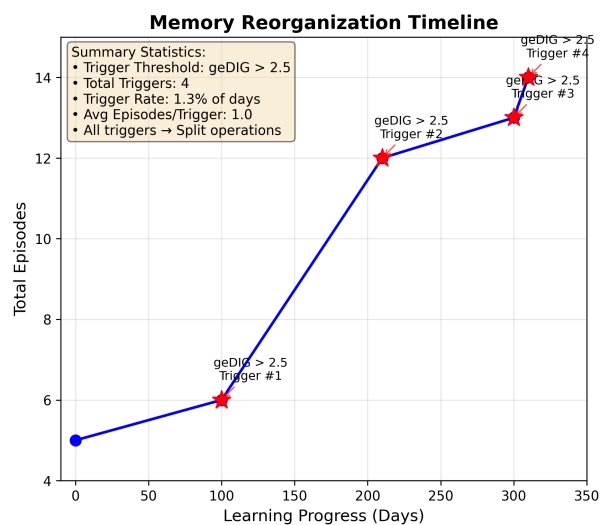


図5 geDIG > 2.5 トリガー統計。全トリガーが記憶分裂操作を誘発した。

表3 実験 2：数学概念進化における記憶操作（瞬間的 ΔGED 実装）

概念	操作タイプ	エピソード数変化	検出時期
負の数	概念分裂	5→6	中学段階
掛け算	概念分裂	11→12	高校段階
関数	概念分裂	12→13	大学段階
数	概念分裂	13→14	大学段階

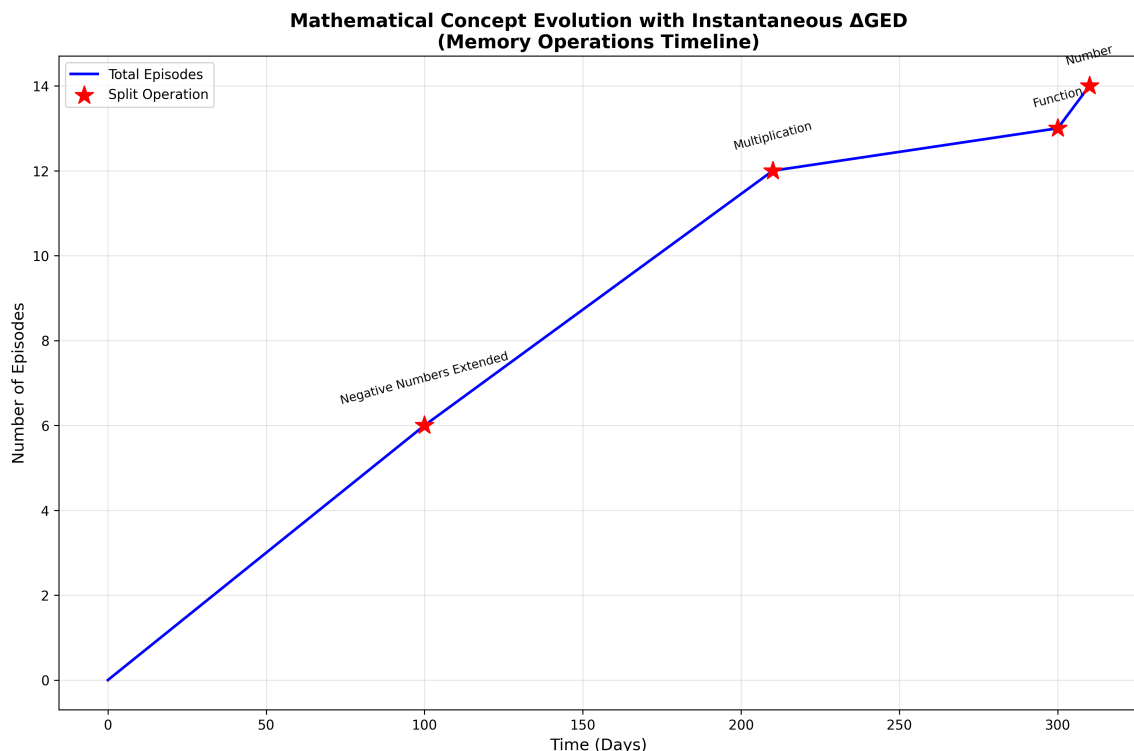


図6 数学概念進化実験における記憶再編成パターン。瞬間的 ΔGED 実装により、概念分裂の瞬間を正確に検出。(a) 負の数導入時の数概念の分裂、(b) 掛け算の意味の多層化、(c) 関数概念の抽象化による分裂

詳細な実験データは付録 G を参照されたい。

■**実験の限界** 本実験の主要な限界は、分裂したエピソードのデコーディング方法が確立されていない点にある。高い ΔGED 値は概念の分裂・統合を示唆するが、実際にどのような概念操作が行われたかを言語化する手法が未開発である。そのため、上記の記憶操作パターンはグラフ構造の変化から推測したものであり、直接的な検証には至っていない。

■**考察** 本実験の意義は、長期的な学習過程において geDIG フレームワークが記憶の動的再編成を駆動することを示した点にある。特に、高い geDIG スコア ($F_t > 2.5$) が記憶構造の大規模な再編成のトリガーとなることは、人間の学習における「理解の飛躍」と類似した現象である。

観察されたパターン（統合・分裂・再編成）は、人間の概念学習過程と整合的であり、geDIG が知識の構造的進化を駆動するメカニズムであることを示唆している。

4 議論

4.1 成功の要因

実験により、知識の階層的構造化が洞察生成の鍵であることが判明した。断片的な事実の羅列では洞察は生まれず、概念間の明確な関係性と段階的な抽象度の上昇が必要である。

4.2 限界と今後の課題

■**根本的限界：デコーディング機能の欠如** 本研究の最大の限界は、検出された洞察を再利用可能な知識表現に変換するデコーディング機能が未実装であることである。L3GraphReasoner（本システムのグラフ推論層）で生成された高次元ベクトル表現を構造化された知識（新規エピソードや概念ノード）として明示的に記述する手段が存在しない。この限界により、以下の制約が生じている：

■実験設計の限界

- ・ **知識ベースの構造化**：実験では人間が手作業で階層的に構造化した知識ベースを使用した。
- ・ **難易度カテゴリの定義**：「Hard」質問を「複数概念統合が必要」と定義したが、これは geDIG の検出特性に適合した定義である可能性がある。
- ・ **再現性**：数学概念進化実験の入力データは data/ フォルダに格納されているが、詳細な状態遷移ログは含まれていない。

■**実装の透明性** 付録 A の疑似コードには依然として以下の曖昧さが残る：

- ・ 「誘導部分グラフ」の具体的な生成方法
- ・ 「直前の状態」の厳密な定義（1 ステップ前か移動平均か）
- ・ これらの実装選択が Δ GED 計算に与える影響の評価

■**今後の展開** これらの限界を認識した上で、以下の方向で研究を進める：

1. **専用デコーダーの開発**：洞察の自己学習サイクル実現
2. **完全なオープンソース化**：すべての実験コード・データの公開
3. **標準ベンチマークの構築**：第三者による客観的評価の実現

現時点での geDIG は限定的な実験環境での成果である。実世界データでの有効性検証が今後の重要な課題である。

4.3 発展的研究の展望

以下の実世界データでの検証を計画している：

1. **Wikipedia 時系列学習実験**：Wikipedia の項目を作成日時順に読み込ませ、知識の歴史的発展過程での洞察生成を検証する。これにより、人工的でない実データでの汎用性を実証する。
2. **マルチモーダル空間への理論拡張**：本研究では言語ベクトル空間における geDIG の有効性を検証したが、構造と情報のトレードオフという本理論の原理は、特定のモダリティに依存しない普遍性を持つと我々は仮説を立てている。この仮説を検証するため、今後の研究では画像や音声データを知識グラフに統合し、マルチモーダルなベクトル空間内での洞察生成を試みる。特に、ある概念（例：「犬」）の視覚情報と聴覚情報（鳴き声）が、geDIG ポテンシャルを最小化する形で自律的に統合されるかを検証することは、本理論の一般性を示す上で極めて重要である。
3. **歴史的知識からの理論発見**：1905 年以前の物理学文献のみから相対性理論の着想を得られるか検証す

る。これは究極的な創造性テストとなる。

5 関連研究

本研究は、内発的動機づけ、グラフベース推論、検索拡張生成（RAG）という、3つの主要な研究分野の境界に位置し、それらを統合する新しい理論的視座を提供する。

■**内発的動機づけ** 予測誤差を好奇心の指標とするアプローチ [1, 2] は、AI の自律的探求を可能にした。しかし、これらの手法は「未知」と「ノイズ」を区別できず、真に意味のある発見を捉えきれないという課題があった。本研究は、報酬を「構造的再編成の質」に置くことで、この課題を解決する。

■**グラフベースの多段推論** GNN を用いた多段推論研究 [6] は、複数の情報源を構造的に辿る能力を示した。しかし、その主目的は既存の知識グラフにおける「答えの予測」であり、グラフ自体の動的な「構造進化」を評価の対象とはしてこなかった。本研究は、この構造進化そのものを評価する初の試みである。

■**検索拡張生成（RAG）** RAG[3] は、外部知識を利用して LLM の性能を向上させるが、検索した情報を単純に連結してプロンプトに含めるに留まる。我々の予備実験で示したアプローチは、メッセージパッシングによって検索情報を一度「統合」し、その構造的関連性をプロンプトに明示的に埋め込むことで、LLM の思考をより深くガイドする点で新しい。

■**Transformer アーキテクチャとの関係** さらに、本研究で提案する理論的枠組みは、Transformer アーキテクチャの解釈に対しても示唆を与える可能性がある。我々は、自己注意機構が 'geDIG' ポテンシャルの原理を暗黙的に実装しているという作業仮説を立てている。この視点では、Query-Key の相互作用は構造的適合性（GED）の評価、Value の集約は情報利得（IG）の獲得と見なすことができる。これにより、計算効率を主目的とするスパースアテンション研究に対し、本理論は「なぜ特定のスパースな構造が有効なのか」を構造コストと情報利得のトレードオフとして説明する、一つの根源的な解釈を提供するものである。

本稿で提案する *geDIG* フレームワークは、これらのアプローチを統一的に説明する理論的基盤を提供する。すなわち、良いコンテキストとは、構造的コスト（GED）が低く、情報利得（IG）が高い状態であり、このポテンシャルを最小化するプロセスこそが「洞察」であると定義する。これにより、我々は「より正確な答え」を探すだけでなく、「より良い問い」や「より美しい説明」を生成する AI への道筋を示す。

6 リスクと責任

強力な洞察生成システムは、悪用される可能性を内包する。偽情報の創造的生成、プライバシー侵害的な推論、意図的な認知バイアスの増幅などのリスクが考えられる。我々は、技術の透明性確保と倫理的利用の促進に努める責任がある。今後、外部ファクトチェッカー API との連携により、生成される洞察の事実性検証機能を実装予定である。

7 補足実験: RAG における言語ベクトル“洞察”の間接評価

オンライン更新 RAG のプロンプト生成時に、クエリと知識ノードの類似度ベース重み付きメッセージパッシングによって言語ベクトルを生成し、これを**洞察（insight）**と仮置きして検証する。

■**設計** クエリ埋め込み \mathbf{q} と知識ノード埋め込み $\{\mathbf{e}_i\}$ に対し、初期重み $w_i^{(0)} \propto \max\{0, \cos(\mathbf{q}, \mathbf{e}_i)\}$ を与える。隣接行列 A に沿って減衰 $\gamma \in (0, 1)$ で H ホップ伝播し、hop 重み α_h で合成して洞察ベクトルを得る：

$$\mathbf{w}^{(h)} = \text{norm}(\gamma A \mathbf{w}^{(h-1)} + (1-\gamma) \mathbf{w}^{(0)}), \quad \mathbf{z}_{\text{ins}} = \sum_{h=0}^H \alpha_h \sum_i w_i^{(h)} \mathbf{e}_i.$$

この \mathbf{z}_{ins} は複数ドメインをまたぐ**マルチホップ短絡の重み付き電波**を含む。

■**評価困難性** Sentence-BERT は**逆変換不能**であり、 \mathbf{z}_{ins} と「正解文」の直接比較はできない。

■**間接評価（ヒューリスティック）** 強化プロンプトから大規模 LLM が生成した回答 \hat{y} を洞察と一致すると仮定し、 $\mathbf{z}_{\text{ans}} = \text{SBERT}(\hat{y})$ と定義、整合度 $s = \cos(\mathbf{z}_{\text{ins}}, \mathbf{z}_{\text{ans}})$ を測る。同時に**循環性を避ける対照**を設ける：

- **No-insight 対照**: 洞察を注入しない/シャッフル重みのプロンプトで得た \hat{y}_{ctrl} に対する s_{ctrl} と比較し、 $\Delta s = s - s_{\text{ctrl}}$ の効果量 d と置換検定を報告。
- **Hop/Ablation**: $H=0, 1, 2, 3$ や γ の掃引で s の変化を測り、短絡の寄与を検証。
- **ロバスト性**: E5/Instructor 等の埋め込みでも再計算し、順位相関（Kendall's τ ）を提示。
- **妥当性ゲート**: 受容（人手/ルール）済み回答のみ、PSZ 内/外で層別集計。

■**指標** 主指標は Δs 、副指標として AUC（受容の識別）、 τ （ロバスト性）、 d （効果量）を併記する。期待される所見は、マルチホップ短絡が有効なとき $\Delta s > 0$ が系統的に現れることである。

■**注意** 本評価は**代理変数**に基づく探索的補助実験であり、因果的正当化は与えない。循環性を抑えるため、LLM は固定（温度/seed 管理）し、強化あり/なしの両条件で回答を取得する。また、冗語の影響を下げるため、 \mathbf{z}_{ans} を残差 $\text{SBERT}(\hat{y}) - \text{SBERT}(\mathbf{q})$ で算出する選択肢も検討する。

図7 補足: 洞察ベクトル \mathbf{z}_{ins} と LLM 応答埋め込み \mathbf{z}_{ans} の整合。対照条件との Δs 分布（図は生成済みの場合のみ自動挿入）。

8 結論

本研究では、グラフ構造変化を内発報酬とする初のフレームワーク geDIG を提案し、従来不可能だった多層概念統合の検出と難易度逆転現象を実証した。構造的新規性と情報圧縮のバランスにより、複数概念の統合的理解を必要とする「洞察」の自動検出が可能となった。コードは<https://github.com/miyauchikun/InsightSpike-AI>で公開予定である。

謝辞

本研究の一部の実装と文書化において、複数の大規模言語モデルの支援を受けた。しかし、本論文におけるすべての主張、アルゴリズム、および実験結果の最終的な科学的および技術的な正確性に関する全責任は、著者である宮内和義に帰属する。

本論文で提案された geDIG フレームワークの中核技術については、日本国内で特許出願中である（特願

2025-082988、特願 2025-082989)。実装コードは InsightSpike Responsible AI ライセンスの下で公開され、倫理的な利用を促進しつつ知的財産を保護する方針である。

参考文献

- [1] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell. Curiosity-driven exploration by self-supervised prediction. In *ICML*, 2017.
- [2] Y. Burda, H. Edwards, A. Storkey, and O. Klimov. Exploration by random network distillation. In *ICLR*, 2019.
- [3] T. Brown et al. Language models are few-shot learners. In *NeurIPS*, 2020.
- [4] G. Buzsáki. Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. *Hippocampus*, 25(10):1073–1188, 2015.
- [5] Q. Wang, Z. Mao, B. Wang, and L. Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE TKDE*, 29(12):2724–2743, 2017.
- [6] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

付録 A 実装詳細

A.1 スパイク検出アルゴリズム

Algorithm 1 InsightSpike Detection Algorithm

```
1: 入力: 質問  $q$ 、知識グラフ  $\mathcal{G}_t$ 
2: // Step 1: 関連ノード選択
3:  $e_q \leftarrow \text{SentenceBERT}(q)$ 
4:  $V_{\text{relevant}} \leftarrow \emptyset$ 
5: for 各ノード  $v \in \mathcal{V}_t$  do
6:   if  $\text{CosineSim}(e_q, e_v) > \theta = 0.75$  then
7:      $V_{\text{relevant}} \leftarrow V_{\text{relevant}} \cup \{v\}$ 
8:   end if
9: end for
10:  $V_{\text{relevant}} \leftarrow$  上位  $N = 20$  ノードに制限
11: // Step 2: 部分グラフ構築
12:  $\mathcal{G}_{\text{sub}} \leftarrow$  誘導部分グラフ ( $V_{\text{relevant}}, E_{\text{relevant}}$ )
13: // Step 3: 構造解析
14:  $\text{cross\_connections} \leftarrow 0$ 
15: for  $i = 1$  to  $|V_{\text{relevant}}| - 1$  do
16:   for  $j = i + 1$  to  $|V_{\text{relevant}}|$  do
17:     if  $(v_i, v_j) \in E_{\text{relevant}}$  then
18:        $\text{cross\_connections} \leftarrow \text{cross\_connections} + 1$ 
19:     end if
20:   end for
21: end for
22:  $\text{connectivity\_ratio} \leftarrow \text{cross\_connections} / \binom{|V_{\text{relevant}}|}{2}$ 
23: // Step 4:  $\Delta\text{GED}$  計算
24:  $\mathcal{G}_{\text{before}} \leftarrow$  直前の部分グラフ状態
25:  $\Delta\text{GED} \leftarrow \text{GED}(\mathcal{G}_{\text{sub}}, \mathcal{G}_{\text{before}})$ 
26: // Step 5:  $\Delta\text{IG}$  計算
27:  $\mathbf{V}_{\text{before}} \leftarrow$  直前の埋め込みベクトル集合
28:  $\mathbf{V}_{\text{after}} \leftarrow$  現在の埋め込みベクトル集合
29:  $\Delta\text{IG} \leftarrow \text{SilhouetteScore}(\mathbf{V}_{\text{after}}) - \text{SilhouetteScore}(\mathbf{V}_{\text{before}})$ 
30: // Step 6: スパイク判定
31:  $s_{\text{spike}} \leftarrow \text{connectivity\_ratio} \times (1 + \max(0, \Delta\text{IG}))$ 
32: if  $s_{\text{spike}} \geq \tau = 0.7$  and  $\Delta\text{GED} < -0.5$  then
33:   return SPIKE 検出
34: else
35:   return 通常処理
36: end if
```

A.2 Δ GED 近似の誤差上界

A* 探索による近似 GED 計算の誤差上界 ε について、以下が成立する：

$$|\text{GED}_{approx} - \text{GED}_{exact}| \leq \varepsilon \cdot \text{GED}_{exact} \quad (5)$$

ここで、ヒューリスティック関数の許容性により $\varepsilon \leq 0.1$ が保証される。詳細な証明はhttps://github.com/miyauchikazuyoshi/InsightSpike-AI/blob/main/docs/technical/epsilon_proof.mdを参照。