Strategic Framework for Cognitive Emergence in Post-Transformer Architectures: A Comprehensive Roadmap for the Synaptix Frontier AI Hackathon

1. Introduction: The Thermodynamic Crisis of the Transformer Paradigm

The current trajectory of artificial intelligence research has arrived at a critical inflection point, characterized by the asymptotic limits of the Transformer architecture. Since the seminal introduction of the Attention mechanism, the field has been dominated by a paradigm of dense, matrix-multiplication-based computing that, while undeniably effective at pattern matching, suffers from fundamental structural deficiencies. As outlined in the problem statement for the Synaptix Frontier AI Hackathon, the Transformer is architecturally static; it is intelligence "frozen in amber," incapable of genuine adaptation once the training phase concludes.[1] Every interaction is a *tabula rasa* event, where the model wakes up with no memory of the previous interaction, relying solely on a finite, computationally expensive context window to simulate continuity. This "Groundhog Day" phenomenon represents not merely a functional limitation but a thermodynamic inefficiency, requiring immense energy to re-compute relationships that biological systems essentially cache in their physical structure.
The Dragon Hatchling (BDH) architecture represents a radical departure from this static orthodoxy. By replacing the dense, immutable weight matrices of the Transformer with a **scale-free graph of locally interacting neuron particles**, BDH aligns artificial intelligence with the principles of biological cognition.[1] It proposes a system where memory is not a buffer of vectors but a physical alteration of the topology itself—a strengthening of synaptic connections via Hebbian plasticity. This shift from **state-as-vector** to **state-as-structure** opens the door to infinite context reasoning, linear computational complexity ($O(T)$), and, most critically, true continuous learning.[1]
For participants in Track 2, Task A ("Dataset Experiments"), the challenge is not simply to train a model on new data. The prompt explicitly weights "Analytical Depth" at 40%, significantly higher than mere implementation or training success.[1] To secure victory in this track, one must treat the BDH model not as a software artifact to be optimized for a leaderboard metric, but as a complex adaptive system—a digital organism whose emergent behavior must be studied, quantified, and visualized. The winning submission will be the one that provides rigorous

empirical evidence of **cognitive emergence**: the spontaneous organization of random neural initialization into structured, domain-isomorphic knowledge graphs that evolve in real-time. This report articulates a comprehensive research strategy to achieve exactly that, leveraging the unique properties of BDH to dominate the competition.

2. Architectural Deconstruction: The Physics of The Dragon Hatchling

To formulate a winning strategy for Task A, it is imperative to first deconstruct the underlying physics of the BDH architecture. The documentation provided describes BDH as a "missing link" between Transformers and brain models, a claim that rests on specific mathematical and structural foundations.[1] Understanding these foundations allows us to design experiments that do not just utilize the architecture, but *stress-test* its defining features against the limitations of the Transformer.

2.1 The Scale-Free Graph Topology vs. Dense Matrix Multiplication

The defining characteristic of the Transformer is its reliance on dense linear algebra. In a standard Multi-Head Attention layer, every token attends to every other token, resulting in a quadratic $O(T^2)$ complexity explosion. This density is structurally homogeneous; the connectivity pattern between neurons is uniform and predefined. BDH, conversely, is defined as a "scale-free graph of neurons".[1] In network theory, a scale-free network is one where the degree distribution follows a power law, $P(k) \sim k^{-\gamma}$. This topology is ubiquitous in nature, characterizing everything from protein interaction networks to the internet and the human brain.

The implications of this topological difference are profound for Task A. In a dense matrix, information is diffuse; a concept is represented as a distributed vector, making it difficult to isolate specific causal mechanisms. In a scale-free graph, information flows through "hubs"—highly connected nodes that serve as aggregators and distributors of signal. This structure naturally supports **modularity**, where distinct sub-graphs specialize in distinct features of the data.[1] A winning experiment must therefore demonstrate that as BDH trains on a specialized dataset, it physically reorganizes itself from a random initialization into a structured topology that mirrors the semantic structure of that dataset. If the dataset contains hierarchical concepts (e.g., a codebase or a medical ontology), the BDH graph should evolve a hierarchical modularity that can be measured and visualized.

2.2 Hebbian Plasticity: The Mechanism of Inference-Time Learning

Perhaps the most revolutionary aspect of BDH is its memory mechanism. The industry standard for "memory" in Large Language Models is the Key-Value (KV) cache, a First-In-First-Out buffer that grows linearly with context length. This is not true memory; it is a temporary scratchpad. BDH replaces this with **Hebbian synapses**—state variables (σ) located on the edges between neurons that update dynamically during the forward pass.[1]

The governing equation, derived from the principle that "neurons that fire together, wire together," allows the model to encode correlations between events $X(i)$ and $Y(j)$ directly into the synaptic weight $\sigma(i,j)$.[1] This means that the model is effectively **training itself during inference**. As it processes a sequence of tokens, it is not just attending to them; it is rewriting its internal circuitry to better predict future tokens based on the immediate past. This "fast weight" mechanism allows for adaptation to non-stationary data distributions (concept drift) without the need for expensive gradient updates.

For Track 2 Task A, this suggests that the ideal dataset is one that exhibits strong temporal dependencies and evolving concepts. A dataset where the rules change over time—such as financial markets or unfolding news narratives—will expose the fragility of static Transformers and the adaptability of BDH. The experiment must be designed to quantify this "Inference-Time Learning," perhaps by measuring the rate at which the model adapts to new vocabulary or shifting correlations within a single long-context session.

2.3 Sparse Positive Activations: The Thermodynamics of Reasoning

The problem statement highlights "Reasoning you can see" as a core value proposition of BDH.[1] This visibility is achieved through **sparse positive activations**, where only approximately 5% of neurons are active at any given time. This contrasts sharply with the dense, often negative or complex-valued activations found in other architectures.

From a thermodynamic perspective, sparsity implies efficiency. The brain consumes roughly 20 watts of power because it does not fire every neuron for every stimulus; it recruits only the necessary circuits. BDH mimics this through a ReLU-lowrank mechanism that gates information flow, ensuring that signals propagate only through relevant pathways.[1] This sparsity is the precursor to **monosemanticity**—the property where a single neuron or synapse corresponds to a single, interpretable concept (e.g., a "finance" neuron or a "Java code" synapse).[2]

A winning submission must rigorously analyze this sparsity. It is insufficient to merely state that the model is sparse; one must analyze the *dynamics* of this sparsity. Does the sparsity increase as the model converges? Do rare, high-information tokens trigger broader activation patterns than common, low-information tokens? These second-order insights into the model's "metabolism" will demonstrate the required Analytical Depth.

2.4 Linear Attention: The Gateway to Infinite Context

Finally, BDH utilizes a form of linear attention that scales as $O(T)$, effectively decoupling memory cost from sequence length.[1] This theoretically allows for "infinite context reasoning." However, infinite context is useless if the model cannot distinguish signal from noise over long durations. The damping parameters $u(i,j)$ in the BDH update rule serve as a forgetting mechanism, ensuring that the synaptic state σ does not become saturated with obsolete information.[1]

In the context of Task A, this capability must be stress-tested. The experiment should involve sequence lengths that would choke a standard GPT-2 model (e.g., >100k tokens). By demonstrating that BDH maintains coherent reasoning and low perplexity over these extended horizons, the submission provides empirical validation of the "infinite context" claim.

---

3. Strategic Domain Selection: Engineering the Optimal Environment

The choice of dataset is the single most critical decision in Track 2 Path A. The prompt suggests domains like "financial text, medical notes, code, regional languages".[1] To maximize the chances of success, we must select a domain that is **topologically isomorphic** to the BDH architecture—one that is graph-structured, temporally dynamic, and logically dense.

We categorize the potential domains based on their alignment with BDH's strengths in Table 1 below.

Domain Category	**Topological Fit**	**Temporal Dynamics**	**Sparsity/Precision**	**Verdict**
Regional Languages	Moderate	Low	Low	**Weak.** While valuable for inclusivity, standard language modeling does not maximally stress the "reasoning" or "graph" aspects of BDH. The structure of language is implicit, not explicit.
Medical Notes (SOAP)	High	Moderate	High	**Strong.** Medical reasoning is highly sparse (ruling out diagnoses) and causal. However, privacy concerns and the static nature of medical knowledge (physiology doesn't change rapidly) limit the "continuous learning"

				demonstration.
Software Repositories	**Very High**	High	High	**Very Strong.** Code is explicitly a graph (ASTs, dependency trees). It requires long-range dependency resolution. However, evaluating "reasoning" in code generation can be binary and brittle.
Financial Transcripts	**Very High**	**Very High**	Moderate	**Optimal.** Financial data is strictly chronological, highly causal ("Inflation -> Rates -> Yields"), and subject to extreme concept drift. It allows for the clearest demonstration of Hebbian adaptability.

3.1 The Optimal Choice: Financial Earnings Call Transcripts

We advocate for the selection of **S&P 500 Earnings Call Transcripts** as the primary dataset for this experiment. This domain offers a unique convergence of properties that perfectly highlight BDH's advantages over Transformers.

3.1.1 Temporal Continuity and Concept Drift

Financial markets are non-stationary systems. A model trained on data from 2019 would be baffled by the economic conditions of 2024 (e.g., high inflation, supply chain shocks, AI boom). A static Transformer would suffer from **catastrophic forgetting** or irrelevance. BDH, with its Hebbian plasticity, should theoretically be able to adapt to these regime changes in real-time. By feeding the model a chronological stream of transcripts (e.g., 2020 through 2024), we can measure its ability to "learn the new rules" of the economy as they emerge, purely through synaptic updates.

3.1.2 Causal Density and Graph Structure

Earnings calls are argumentation structures. Executives present a thesis (e.g., "Our margins expanded..."), followed by supporting evidence ("...due to operational efficiencies"), and then defend this thesis against analyst skepticism in the Q&A. This is not just text; it is a causal graph. The relationship between "Revenue Growth" and "Stock Price" is a weighted edge in the market's knowledge graph. BDH's architecture is uniquely suited to mapping this graph directly onto its internal topology (G_x).

3.1.3 Interpretability via Monosemanticity

Financial analysts rely on specific indicators. Terms like "EBITDA," "Guidance," "Headwinds," and "AI" act as semantic attractors. We hypothesize that BDH will dedicate specific, monosemantic neurons to these concepts.[2] This allows for a compelling demonstration of the "Analytical Depth" criteria: we can identify the "Inflation Neuron" and watch it fire during the 2022 transcripts, proving the model is tracking macroeconomic trends at the neuronal level.

3.2 Dataset Specification and Preprocessing

To execute this strategy, we define the following data pipeline:

1. **Source:** The S&P 500 Earnings Call Transcripts dataset (e.g., kurry/sp500_earnings_transcripts on HuggingFace [3]). This dataset provides the necessary historical depth (2005–2025) and volume.
2. **Filtering:** We will restrict the scope to the **Technology Sector**. This sector exhibits the highest volatility and rapid vocabulary evolution (e.g., "Cloud" -> "Crypto" -> "Metaverse" -> "Generative AI"), providing the strongest signal for detecting adaptability.
3. **Chronological Sequencing:** Unlike standard training where data is shuffled, we will strictly enforce chronological ordering. The model will consume Q1 2015, then Q2 2015, and so on. This is essential to testing the "Continuous Learning" hypothesis.
4. **Structure Preservation:** We will retain the distinction between the "Presentation" (prepared remarks) and the "Q&A" (spontaneous reasoning). This allows us to test if the model builds a context during the Presentation that improves its perplexity on the Q&A—a proxy for "listening" and "understanding."

---

4. Experimental Methodology: Quantifying Cognitive Emergence

The problem statement explicitly warns against "Surface-level 'I ran the demo' submissions".[1] To satisfy the 40% weight on "Analytical Depth," we must define and calculate rigorous metrics that go beyond standard loss curves. We propose a suite of **Frontier Metrics** designed to measure the specific architectural claims of BDH.

4.1 Frontier Metric 1: The Topological Modularity Index (TMI)

Objective: To verify the claim that BDH evolves a "scale-free graph of neurons" with "high modularity".[1]
Theoretical Basis: In biological brains, modularity minimizes wiring cost and allows for specialized processing. If BDH is learning correctly, its effective weight matrix should not be random; it should cluster into densely connected communities.
Methodology:

1. **Graph Extraction:** We interpret the product of the decoder and encoder matrices $W_{eff} = D_x E$ as the weighted adjacency matrix of a directed graph $G = (V, E)$, where V is the set of neurons.
2. **Sparsification:** Since W_{eff} is dense, we apply a threshold τ to retain only the top $k\%$ of edges (e.g., $k=5\%$) based on magnitude, simulating the active synaptic pathways.
3. **Community Detection:** We apply the **Louvain Algorithm** or **Leiden Algorithm** to detect communities within this graph.[4]
4. Metric Calculation: We compute the Newman Modularity (Q) score:

   $$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

   Where A_{ij} is the edge weight, k_i is the degree, m is total edge weight, and $\delta(c_i, c_j)$ indicates if neurons i and j are in the same community.
5. **Comparative Analysis:** We compare the Q value of the trained BDH model against (a) an untrained initialized BDH model and (b) the FFN weights of a GPT-2 model of comparable size.

Hypothesis: The trained BDH model will exhibit a significantly higher Q score ($Q_{trained} \gg Q_{init}$), indicating the emergence of functional modules (e.g., a "Revenue" module, a "Risk" module).

4.2 Frontier Metric 2: The Synaptic Persistence Score (SPS)

Objective: To measure the durability of Hebbian memory and verify the "True Continuous Learning" capability.[1]
Theoretical Basis: In a Hebbian system, memories are encoded as synaptic strengths. If the model "learns" a new fact (e.g., "NVIDIA acquires ARM"), the synapses connecting the representations of "NVIDIA," "Acquire," and "ARM" should show a persistent increase in weight that decays slowly over time (power law forgetting), rather than vanishing instantly (exponential forgetting).
Methodology:

1. **State Snapshotting:** During the inference pass on the chronological test set, we save snapshots of the synaptic state matrix σ_t at regular intervals (e.g., every 100 tokens).
2. Auto-Correlation: We calculate the cosine similarity between the state at time t and

time $t+k$:

$$\text{SPS}(k) = \frac{\text{vec}(\sigma_t) \cdot \text{vec}(\sigma_{t+k})}{\| \text{vec}(\sigma_t) \| \| \text{vec}(\sigma_{t+k}) \|}$$

3. **Decay Analysis:** We plot $\text{SPS}(k)$ against log-time.

Hypothesis: BDH will exhibit **Long-Range Dependence** (Hurst exponent $H > 0.5$), meaning that memory traces persist over long durations ($O(T)$ or greater). In contrast, a Transformer's effective memory (attention) cuts off abruptly at the context window limit. This proves that BDH carries "wisdom" from the beginning of the earnings call to the end.

4.3 Frontier Metric 3: The Sparsity-Entropy Correlation (SEC)

Objective: To validate the "Reasoning you can see" claim and the efficiency of sparse activations.[1]

Theoretical Basis: The **Free Energy Principle** in neuroscience suggests that brains minimize energy expenditure. They should be sparse (low activity) when the environment is predictable (low entropy) and dense (high activity) when the environment is surprising (high entropy/surprisal).

Methodology:

1. **Input Entropy:** For each input sequence x_t, we calculate the empirical entropy $H(x)$ or use the model's own perplexity as a proxy for "surprisal."
2. **Activation Density:** We measure ρ_t, the fraction of neurons in the layer with non-zero activations after the ReLU-lowrank operation.
3. **Correlation:** We compute the Pearson correlation coefficient r between Perplexity and ρ_t.

Hypothesis: We expect a strong positive correlation ($r > 0$). When the CEO reads a boilerplate disclaimer (low perplexity), the model should "sleep" (low sparsity). When the analyst asks a hard question about missing revenue targets (high perplexity), the model should "wake up" (higher activation density). Demonstrating this **dynamic metabolism** would be a powerful indicator of the model's sophistication.

4.4 Frontier Metric 4: Monosemanticity and Concept Localization (MCL)

Objective: To empirically demonstrate that individual synapses encode specific concepts.[1]

Theoretical Basis: The paper claims BDH exhibits monosemanticity even at small scales. We need to find the "concept neurons."

Methodology:

1. **Concept Probing:** We define a set of high-frequency financial concepts: $C = \{ \text{"Inflation"}, \text{"AI"}, \text{"Dividend"}, \text{"Layoffs"} \}$.
2. **Activation Mapping:** We run a subset of the data tagged with these concepts through the model. For each concept $c \in C$, we identify the set of neurons N_c that activate maximally when c is present in the input.
3. Selectivity Score: For each identified neuron n, we calculate a selectivity score:

$$S(n, c) = \frac{P(\text{active} | c)}{P(\text{active} | \neg c)}$$

A high score implies the neuron is a dedicated detector for that concept.

Visualization: We will generate "Activation Atlases" [6]—visual maps where neurons are colored by their semantic role. Showing a cluster of neurons that only fire for "Risk Factors" would be visual proof of the model's interpretability.

---

5. Implementation Roadmap: The "Dragon Tamer" Pipeline

This section translates the theoretical strategy into a concrete engineering plan. We assume the use of the official pathwaycom/bdh repository, which is built on PyTorch.[7]

5.1 Phase 1: Infrastructure and Data Engineering

Repository Setup:

The base repository provides bdh.py (model architecture) and train.py (training loop).7 We must fork this and create a new branch track2-finance.

- **Dependencies:** We will need networkx for graph analysis [5], scikit-learn for clustering, and matplotlib/seaborn for visualization.
- **Hyperparameters:**
  - **Model Size:** We will target a "Small" configuration to fit within standard hackathon compute limits (e.g., T4 or A100 GPU).
  - N (Neurons): 4096 (Scale-free graph size).
  - D (Low-rank dimension): 256.
  - L (Layers): 6.
  - **Why these numbers?** Keeping N high relative to D ($N \gg D$) is crucial for the "bottleneck" effect that drives sparsity and modularity.

Data Ingestion:

We will write a custom data loader finance_loader.py.

- **Parsing:** Use regex to split transcripts into (Speaker, Text) tuples.
- **Tokenization:** Use the standard GPT-2 BPE tokenizer for compatibility.
- **Streaming:** Implement a IterableDataset in PyTorch to stream the data chronologically. This mimics the "Live Data" aspect of Pathway's engine, even if we are processing offline for Track 2.

5.2 Phase 2: The Training Regime

We will implement a **Two-Stage Training Protocol** to isolate the effects of structural learning vs. Hebbian adaptation.

Stage A: Structural Pre-training (2005–2018 Data)

- **Goal:** Learn the static weights (E, D_x, D_y) that define the "grammar" of the financial

domain and the base topology of the neuronal graph.
- **Method:** Standard Autoregressive Language Modeling (Next Token Prediction).
- **Optimization:** AdamW optimizer, Cosine Learning Rate Schedule.
- **Constraint:** We apply L1 regularization on the activations during this phase to *encourage* the emergence of sparsity, reinforcing the architecture's natural tendency.

Stage B: Hebbian Inference Evaluation (2019–2024 Data)
- **Goal:** Test "Continuous Learning" and "Generalization over Time."
- **Method:** We **freeze** the static weights (E, D_x, D_y). We run the model on the unseen future data (2019–2024).
- **Mechanism:** The model is allowed to update its synaptic state σ via the Hebbian update rule as it processes the stream. It is *not* allowed to update its static weights via backpropagation.
- **Comparison:** We compare the perplexity of this frozen BDH model against a frozen GPT-2 model.
    - *Success Criterion:* If BDH maintains lower perplexity, it proves that the **synaptic state dynamics alone** are sufficient to adapt to new market conditions (e.g., learning that "COVID-19" implies "lockdown" purely from context accumulation), whereas the Transformer fails to adapt.

5.3 Phase 3: Analytical Instrumentation

We will develop a library dragon_metrics.py to calculate the metrics defined in Section 4.
- **Hook System:** We will use model.register_forward_hook() in PyTorch to intercept the activation vectors $y_{t,l}$ and the state matrices $\sigma_{t,l}$ without disrupting the forward pass.
- **Graph Exporter:** A function export_topology(model, epoch) will extract the effective adjacency matrix and save it as a .gexf file for analysis in Gephi or NetworkX.
- **Real-time Dashboard:** Using wandb (Weights & Biases), we will log the "Sparsity" and "Modularity" metrics continuously during training, allowing us to visualize the *emergence* of structure in real-time.

---

6. Discussion: Interpreting the Emergence

This section anticipates the "Insight Quality" components of the final submission, synthesizing the expected results into a coherent narrative about the nature of the BDH architecture.

6.1 The Isomorphism of Markets and Minds

We anticipate that our topological analysis (TMI) will reveal a striking isomorphism between the BDH neural graph and the financial knowledge graph. Just as the financial system has "hub" entities (Central Banks, Tech Giants) that drive the market, the BDH model will evolve "hub" neurons that represent these central concepts. This confirms that BDH does not just "learn patterns" in the statistical sense; it **internalizes the causal structure** of its environment. This is a profound argument for the "Biological Plausibility" of the model.[1]

6.2 The End of "Training vs. Inference"

Our experiment in Stage B (Hebbian Inference) challenges the rigid dichotomy between "Training" and "Inference" in Deep Learning. In the Transformer era, a model is trained once and then deployed as a static artifact. In the BDH era, **inference is learning**. Every token processed leaves a synaptic trace that alters how the model processes the next token. This "Continuous Adaptation" capability suggests that BDH models could be deployed in "always-on" environments (e.g., algorithmic trading, network security) where they naturally evolve with the data, rendering the concept of "model drift" obsolete—drift is now just "learning."

6.3 Monosemanticity as a Safety Feature

The demonstration of monosemantic neurons (MCL metric) has immense implications for AI Safety. If we can isolate the "Fraud" neuron or the "Hallucination" synapse, we can utilize **Graph Pruning** to surgically remove these behaviors without retraining the model. This offers a level of control and auditability that is impossible with "black box" Transformers, making BDH uniquely suited for high-stakes industries like Finance and Healthcare.

---

7. Advanced Tactics: "Model Merging" and Beyond

To secure the final "Novelty" and "Creativity" points, we propose an extension experiment based on Section 7.1 of the research paper: **Model Merging**.[1]

7.1 The "Conglomerate" Hypothesis

The paper suggests that BDH models can be concatenated. We can demonstrate this by training two separate "Specialist" models:

1. **Tech-BDH:** Trained only on Technology sector earnings calls.
2. **Pharma-BDH:** Trained only on Healthcare sector earnings calls.

We then create a **Merged Model** by concatenating their neuron graphs (expanding $N_{total} = N_{tech} + N_{pharma}$) and their static weights.

- **Prediction:** The merged model should immediately be competent in *both* domains without any fine-tuning.
- **Significance:** This suggests a future where we do not train massive monolithic models (like GPT-4) from scratch. Instead, we train small, specialized "Hatchlings" and fuse them into a "Dragon" tailored to specific needs. This **Modular AI** paradigm is far more sustainable and scalable than the current "scaling laws" approach.

7.2 Visualizing the Brain of the Market

For the visual component of the submission, we will create a dynamic animation.

- **Left Panel:** The scrolling text of an earnings call.

- **Right Panel:** The BDH neural graph.
- **Action:** As keywords appear in the text (e.g., "Revenue beat estimates"), specific clusters of neurons in the graph will "light up" (change color/size). This visualization will make the abstract concept of "sparse activations" visceral and immediate for the judges, satisfying the "Reasoning you can see" criterion.[1]

---

8. Conclusion: The Path to Victory

The Synaptix Frontier AI Hackathon is not merely a coding competition; it is a search for the next paradigm in artificial intelligence. The Transformer has brought us far, but its static, dense nature is a dead end for true continuous learning. The Dragon Hatchling (BDH) offers a path forward—a path paved with scale-free graphs, Hebbian plasticity, and sparse, interpretable reasoning.

This report has outlined a comprehensive strategy to win Track 2 Task A by:

1. **Selecting a Domain (Finance)** that maximizes the model's structural advantages (Time, Causality, Drift).
2. **Defining Frontier Metrics (TMI, SPS, SEC, MCL)** that rigorously quantify "Analytical Depth" beyond simple loss functions.
3. **Designing a Two-Stage Experiment** that proves the reality of "Inference-Time Learning" via Hebbian dynamics.
4. **Proposing Novel Applications** like Model Merging and Surgical Safety via Graph Pruning.

By executing this plan, we do not just demonstrate that BDH works; we demonstrate *why* it matters. We move the conversation from "optimizing weights" to "growing minds." This is the frontier. And this is how we win.

Appendix: Deliverable Checklist for Track 2 Path A

Component	Description	Judging Criterion Targeted
train_finance.py	Custom training script with chronological streaming and L1 regularization hooks.	Technical Correctness (30%)
dragon_metrics.py	Library calculating Modularity (Q), Sparsity (ρ), and Persistence (SPS).	Insight Quality (30%)
Dataset Card	Documentation of the S&P 500 subset, highlighting concept drift and temporal structure.	Rigor (5%)
Topology Plots	Static SVG visualizations of the learned graph structure	Presentation (20%)

	(W_{eff}) showing community clusters.	
Sparsity Heatmap	Dynamic visualization of activation density over time overlaid on transcript text.	Presentation (20%)
Analysis Report	A PDF synthesizing the metrics into the "Cognitive Emergence" narrative.	Analytical Depth (40% - implied)
Bonus: Merging Demo	Script demonstrating the concatenation of Tech and Pharma models.	Creativity (20%) / Novelty (8%)

Works cited

1. 694cb6aaa753e_Synaptix_ps__1_.pdf
2. The Dragon Hatchling: The Missing Link between the Transformer and Models of the Brain - arXiv, accessed on December 26, 2025, https://arxiv.org/html/2509.26507v1
3. kurry/sp500_earnings_transcripts · Datasets at Hugging Face, accessed on December 26, 2025, https://huggingface.co/datasets/kurry/sp500_earnings_transcripts
4. Communities — NetworkX 3.6.1 documentation, accessed on December 26, 2025, https://networkx.org/documentation/stable/reference/algorithms/community.html
5. greedy_modularity_communities — NetworkX 3.6.1 documentation, accessed on December 26, 2025, https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.modularity_max.greedy_modularity_communities.html
6. Activation Atlas - Distill.pub, accessed on December 26, 2025, https://distill.pub/2019/activation-atlas/
7. pathwaycom/bdh: Baby Dragon Hatchling (BDH) – Architecture and Code - GitHub, accessed on December 26, 2025, https://github.com/pathwaycom/bdh