Generative Psychometrics

Construct Emergence Tracing: From Meaning to Mechanism in Transformer Interpretability via Generative Psychometrics

Hudson Golino *

Department of Psychology, University of Virginia, Charlottesville, VA, USA
*Corresponding author. Email: hfg9s@virginia.edu

Abstract

Transformer-based language models (LMs) allow researchers and practitioners to develop and deploy systems that perform impressive feats such as generating coherent texts and answering questions across various domains. Yet, there is currently an interpretability problem regarding the internal representation of multidimensional psychological constructs (e.g., personality traits) throughout multiple layers of deep neural networks. Most of the current methodologies used for understanding how LMs represent concepts and information follow a bottom-up approach (e.g., neurons, attention heads, and circuit dynamics towards human-level concepts). Still, there is limited evidence regarding how and where complex psychological constructs emerge during computation. Here, we propose Construct Emergence Tracing (CET), a novel methodology to address this problem. CET utilizes ideas from Generative Psychometrics to trace the emergence of concepts from high-level meaning to underlying mechanisms. CET consists of Dynamic Exploratory Graph Analysis combined with three complementary indices reflecting the quality of representation (semantic coherence via normalized mutual information; NMI, entropy-based organization structure via Total Entropy Fit Index; TEFI, and topological compressibility via a compression-based proxy for Kolmogorov Complexity; KC). CET also computes a weighted composite of the above three indices in order to derive a metric named Representational Maturity (RM). We applied CET to fourteen LM models ranging from nominal sizes of 117 million to 32 billion parameters. We utilized a fixed design of five items per attribute representing the five attributes of grandiose narcissism and yielded a total of 9,801,000 network estimates across recorded states, embedding dimensions, and repeated item samples. Results show that across architectures construct recovery generally increases from early to intermediate states. However, its initiation, amplitude, alignment with different metrics, and termination vary considerably across models. Across Qwen2.5-Instruct, overall mean NMI followed a monotonic increase based on nominal model size from 0.504 (0.5B) to 0.653 (32B). This trend did not replicate for selected peak values or output layer retention. Furthermore, the final state was divided into two distinct patterns of behaviors: (a) five models showed zero or close-to-zero mean NMI (GPT-2, Phi-4-mini, Qwen2.5-32B, GPT-OSS-20B, and Muse-Glimmer-30B); (b) nine models maintained final state mean NMI within 0.504-0.777 range. Lastly, we observed no consistent convergence in the optimal layers regarding peak semantic coherence, optimal entropy fit, and maximal representational maturity. Overall, we demonstrated that both the number of parameters and the extraction from the output layer alone do not suffice in capturing abstract representations inside LMs. Through CET, researchers and practitioners obtain macro-level estimations of construct organization trajectories across different layers.

Keywords: interpretability, transformer representations, generative psychometrics, network psychometrics, DynEGA, TEFI, Kolmogorov complexity, representational maturity, construct emergence

1. Introduction

Large Language Models (LLMs), with the transformers architecture can range from models with 117 million parameters as like GPT-2 (Brown et al., 2020) to hundreds of billions of parameters for more recent frontier architectures (Shao et al., 2024), have dramatically changed Natural Language Processing. Their progress has been dramatically fast, from generating human-like text passages to handling broad contexts and solving very complex problems. However, an important scientific issue about LLMs remains unanswered: **How do these models represent and organize abstract information internally in their layers?** Given the use of these models in several consequential applications applications, such as medicine, legal proceedings, education, finance, governance, science, and more, a deeper understanding of the mechanisms behind their internal organization is central to promote transparency, accountability, alignment and safety concerns (Bommasani et al., 2021). The current paper aims to contribute to the field of interpretability by investigating the organizational trajectory of abstract psychological constructs across the layers of transformer models.

Research on LLM interpretability has generated a number of methodologies to analyze the internal representational processes of transformer-based models. For example, Shapley Additive Explanations (SHAP) (Lundberg & Lee, 2017) and Integrated Gradients (Sundararajan, Taly, & Yan, 2017) quantify the importance of input features. Attention visualization and neural activation examination inspect how certain components behave in response to diverse stimuli (Olah et al., 2020; Vig, 2019). Mechanistic interpretability approaches, especially circuit tracing and attributions graphs (Lindsey, Gurnee, Ameisen, et al., 2025; Olah et al., 2020), describe specialized localized pathways within a rich connectionist tradition of studying learned representations (Rumelhart, Hinton, & Williams, 1986). Additionally, causality and counterfactual investigations explore model robustness, while activation geometries are compared via representational similarity analysis using metrics such as Centered Kernel Alignment (CKA) (Kornblith, Norouzi, Lee, & Hinton, 2019), Singular Vector Canonical Correlation Analysis (SVCCA) (Raghu, Gilmer, Yosinski, & Sohl-Dickstein, 2017) and Representational Similarity Analysis (RSA) (Kriegeskorte, Mur, & Bandettini, 2008). Probing classifiers, on the other hand, examine whether linguistic traits can be linearly decoded (Alain & Bengio, 2017; Belinkov, 2022).

All of these methods have in common an underlying directional assumption. They move **from mechanisms to meanings**. That is, starting from units such as neurons, attention heads, prompt tokens, pairs of activations, or specific activation geometries and moving upwards towards meaningful constructs. Accordingly, these methods have proven invaluable in addressing micro-level questions such as *how does the model compute feature X?* or *is there linear evidence for property Y in layer l?*. Yet, none of them addresses the macro-level research question posed in the field of interpretability literature: **Where, When and How Coherently is a Multidimensional Abstract Construct Represented across LLM Transformer Layers?** (Belinkov, 2022; Lindsey et al., 2025).

For nearly a century, Measurement Sciences in Quantitative Psychology faced this challenge. Psychological attributes such as narcissism, empathy, and anxiety are not simple dichotomous features. They constitute complex multidimensional entities where observable attributes are expected to form a coherent representation of the construct itself (Cronbach & Meehl, 1955; **eskin2012?**). Measuring whether a large language model represents an abstract construct across its entire processing hierarchy requires the development of a methodology that turns the directionality of interpretability inquiry upside down. Such a methodology should start by postulating a meaningful attribute, and proceed to evaluate how accurately, parsimoniously, and stably the model's internal representations reproduce the construct organization across all stages of its computation. Thus, allowing mechanistic interpretation of the resulting representational spaces and their underlying mechanisms.

Recent advances at the interface between quantitative psychology and artificial intelligence provide the foundations to make such a reverse engineering attempt possible. In the framework of *Generative Psychometrics* (Russell-Lasalandra, Christensen, & Golino, 2024), LLMs have been

incorporated into a network psychometric assessment framework to generate pools of test items, filter semantic redundancies and examine their dimensional structure before collecting any human data. In practice, Generative Psychometrics creates psychometric scales whose structure fits well with those observed in nationally representative samples (Garrido, Russell-Lasalandra, Rodríguez-Montoya, & Golino, 2026; Russell-Lasalandra et al., 2024; Russell-Lasalandra & Golino, 2026). Previous research focused on Generative Psychometrics' potential to assist the traditional process of psychometric scales construction. The central argument of the current paper is that Generative Psychometrics can be used to investigate the potential of Psychological Measurement Science for understanding artificial intelligence.

One breakthrough in LLM embeddings opened the possibility to reverse-engineer their representational process. Golino (2026) showed that LLM embedding spaces represent abstract psychological constructs in a non-linear way. Using Dynamic Exploratory Graph Analysis (DynEGA) (Golino, Christensen, Moulder, Kim, & Boker, 2022) across embedding coordinates and considering the coordinate order as pseudo-time, Golino (2026) found that dimensional structure is better represented on particular embedding configurations rather than being uniformly distributed across all dimensions of the embedding landscape. Optimizing for the accuracy via Normalized Mutual Information (NMI) (Strehl & Ghosh, 2002) and structural organization via Total Entropy Fit Index (TEFI) (Golino et al., 2021) provided more accurate estimations of the dimensionality structure of psychological constructs than standard cross-sectional analyses of embeddings. If LLM output embeddings represent searchable landscapes, then transformer hidden states must also be analyzed as representational landscapes at every processing stage.

The current paper introduces **Construct Emergence Tracing** (CET), a new framework for construct-centered interpretability research. CET extends representational landscape search (Golino, 2026) throughout the depth of LLM transformer architectures. Given a theoretical multidimensional psychological construct definition, CET performs DynEGA analysis at every processing layer, identifying three different but complementary aspects of its representational organization: (a) **semantic coherence**, quantified using NMI between predicted labels; (b) **organizational structure**, assessed using TEFI calculated based on Von Neumann entropy (Golino, Jiménez, Garrido, & Christensen, 2024; Golino et al., 2021); and (c) **representational efficiency**, evaluated using Kolmogorov complexity (Golino, 2025; Golino, Nesselroade, & Christensen, 2025; Kolmogorov, 1965). All three aspects of representational organization receive a score, which can be combined in a single index named *Representational Maturity Index (RMI)*, tracing the construct emergence trajectory across the transformer depth. **Therefore, CET establishes the interpretability pillar of Generative Psychometrics:** applying psychological measurement science *to* artificial intelligence. Model hidden states are treated as proxies for empirical observations, embedding dimensions define the searchable representational space, and transformer depth acts as the developmental axis.

This manuscript contributes to the interpretability literature in three key ways. Firstly, it formalizes the multidimensional construct representation as a developmental trajectory across the depth of a transformer architecture, establishing a theoretical foundation for manifold separation preservation during time-delay reconstruction. In addition, it integrates Network Psychometrics, the Total Entropy Fit Index (TEFI), and Algorithmic Complexity to establish a reproducible layer-by-embedding interpretability pipeline. Additionally, it explores CET performance in 14 Large Language Models (LLMs) varying between 117M to 32B parameters evaluated with a five-items-per-attribute test design. Specifically, the paper investigates if construct recovery follows universal depth profiles, how scaling affects mean/peak recovery accuracy and if coherent constructs representations persist from intermediate layers to outputs. The aim of this study is to introduce CET and demonstrate its utility for interpreting language representations and assessing their multidimensional organization.

2. Related work

2.1 Interpretability methods for transformer models

In the last decade, the interest in investigating the internal functioning of transformers models has grown substantially. Currently, there are several types of methods designed to investigate transformer models' interpretability at various levels of abstraction. First, attribution methods (e.g., SHAP, Integrated Gradients, Attention Rollouts) focus on quantifying the contribution of each feature or token to predicting specific output values. These methods do not allow us to investigate if a set of items forms a multidimensional latent structure, such as constructs with multiple attributes/dimensions. Second, attention and neuron analyses focus on understanding particular computations performed within attention heads/neurons. It involves investigating the role that particular sets of attention heads/neurons play during a particular task. Finally, mechanistic interpretability seeks to explore the computational mechanisms responsible for high-level processes like multi-step reasoning, entity tracking, and factual retrieval. Methods include: Circuit Tracing and Attribution Graphs (Lindsey et al., 2025; Olah et al., 2020). The primary drawback of these methods is their micro-orientedness – they seek to understand *how* microscopic processes operate. On the other hand, CET describes the trajectory through which complex constructs evolve into representations across layers.

2.2 Representation geometry and probing classifiers

The decoding capacity of transformers models' hidden layers has been explored using probing classifiers, whose purpose is to estimate if particular linguistic elements such as parts-of-speech, syntactic relationships, or semantic roles, can be linearly decoded from layer activations. Results have demonstrated that syntactic information becomes maximal at mid layers whereas semantic information tends to cluster at late layers (Jawahar, Sagot, & Seddah, 2019; Rogers, Kovaleva, & Rumshisky, 2020; Tenney, Das, & Pavlick, 2019). Other approaches use similarity-based metrics (Centered Kernel Alignment, CKA; Singular Vector Canonical Correlation Analysis, SVCCA; Representational Similarity Analysis, RSA) to compare geometrical structures between layer-wise activations, or compare to the behavioral similarity matrix.

Although the described methodologies play a pivotal role in interpretability research, two limitations should be addressed. First, probing methods assess decoding potential and thus cannot reveal information about whether a construct formed a coherent multidimensional structure. Second, while metrics like CKA and SVCCA enable comparing two representation geometries, they do not evaluate recovery accuracy according to a theoretical construct structure. In contrast, CET aims precisely at estimating construct recovery accuracy and the quality of recovered structures across layers.

2.3 Network Psychometrics

Finally, CET borrows fundamental notions from network psychometrics (Epskamp, Maris, Waldorp, & Borsboom, 2018; Epskamp, Waldorp, Mõttus, & Borsboom, 2018; Golino, Christensen, & Garrido, 2022; Golino et al., 2025), an interdisciplinary branch of quantitative psychology devoted to modeling multivariate psychological phenomena as undirected weighted networks. Using the Exploratory Graph Analysis framework (EGA), the method estimates the latent dimensionality via either regularized partial correlation coefficients (Christensen & Golino, 2021; Golino & Epskamp, 2017; Golino et al., 2020) via graphical lasso and EBIC-based model selection (Epskamp, Maris, et al., 2018; Friedman, Hastie, & Tibshirani, 2008), or via triangulated maximally filtered graph (TMFG) (Massara, Di Matteo, & Aste, 2016). Sets of densily connected nodes in the network are proxies for latent dimensions Christensen & Golino (2021) that can be recovered using community detection methods.

In contrast with factor analytic techniques, EGA uses the information provided by unique variance between pairs of variables rather than total shared variance across all variables, making the estimation resilient to high interfactor correlation. Consequently, extensive Monte Carlo simulations

demonstrated that EGA leads to greater accuracy compared to traditional techniques based on latent variable models (Christensen, Garrido, Guerra-Peña, & Golino, 2024; Golino & Epskamp, 2017; Golino et al., 2020). Furthermore, CET adopts Dynamic EGA (DynEGA) (Golino, 2026; Golino, Christensen, Moulder, et al., 2022; Tomašević, Golino, & Christensen, 2024), which combines EGA with phase-space reconstruction via time-delay embedding and derivative estimation using Generalized Local Linear Approximation (GLLA) (Boker, Deboeck, Edler, & Keel, 2010; Deboeck, Montpetit, Bergeman, & Boker, 2009). Psychometric evaluation metrics provide quality criteria of the network partitioning: the Total Entropy Fit Index (TEFI) (Golino et al., 2024, 2021) evaluates partition entropy reduction in the correlation matrix via Von Neumann entropy (Golino et al., 2021); network topology is evaluated through a Kolmogorov Complexity (KC) estimator (Golino, 2025; Golino et al., 2025), measuring the compressibility of serialized edge lists generated with thresholding and single-layer canonical prime weight coding (Golino, 2025).

3. Theoretical Framework

3.1 Representational Development in Transformers

In general, the process performed within a Transformer can be seen as a hierarchical abstraction pipeline, whereby each decoding layer performs attentional and feedforward operations that transform token representation in increasingly rich context. As pointed in earlier sessions, the evidence from probing classifiers supports a layered progression from early layers that can capture local lexical and syntactic regularities, to middle layers that integrate broader context, forming abstract semantic groupings, to upper layers that change representations to support next-token prediction (Jawahar et al., 2019; Rogers et al., 2020; Tenney et al., 2019). CET formalizes this progression by defining what it means for a multidimensional construct to be represented at layer ‘ and by tracking its evolution across depth.

Definition 1 (Construct Representation)**.** *Let $\mathcal{C}$ denote a psychological construct of interest whose items (or operationalizations) form a multidimensional taxonomy of r dimensions. Such psychological construct $\mathcal{C}$ is said to be* represented *in the model's state ‘ if the activations of each element associated with $\mathcal{C}$ constitute a community structure aligned with its r dimensions. More precisely, the agreement between the clustering in such state and the taxonomy of $\mathcal{C}$ should exceed chance levels according to a proper null model.*

Definition 2 (Representational Trajectory)**.** *Let $\mathcal{T}(\mathcal{C})$ denote the* representational trajectory *of the psychological construct $\mathcal{C}$ along L recorded neural states. This trajectory can be described as the following sequence:*

$$\mathcal{T}(\mathcal{C}) = \{\mathcal{R}_1(\mathcal{C}), \mathcal{R}_2(\mathcal{C}), \ldots, \mathcal{R}_L(\mathcal{C})\}. \tag{1}$$

Herein, $\mathcal{R}_{‘}(\mathcal{C})$ is the representation of $\mathcal{C}$ at neural state ‘. The representational state is defined by a set of indices characterizing the semantics, structure and efficiency of such a neural representation.

3.2 Representational Maturity Metrics

For assessing a complex, multidimensional construct $\mathcal{C}$, the following set of criteria evaluates three complementary aspects of the representational states:

Semantic Coherence: the NMI criterion measures the degree to which the community detection procedure aligns with the labeling of items pertaining to $\mathcal{C}$ based on normalized mutual information (Strehl & Ghosh, 2002), ranging from 0 (no agreement) to 1 (perfect agreement).

Structure Quality: the TEFI index evaluates the quality of the dimensional structure via entropy reduction with respect to the overall correlation structure (Golino et al., 2024, 2021). The Von Neumann entropy $\mathcal{S}(æ)$ is computed based on $æ = \mathbf{R}/p$—the rescaled version of the sample correlation

matrix **R**. It holds that æ is a Hermitian positive-semidefinite matrix satisfying trace(æ) = 1. The Von Neumann entropy $\mathcal{S}(æ)$ can be expressed as:

$$\mathcal{S}(æ) = -\text{trace}(æ \log æ) = -\sum_i \breve{}_i \log \breve{}_i, \tag{2}$$

where $\{\breve{}_i\}$ denote the eigenvalues of æ. For a partition of æ into N_F dimensions, TEFI evaluates entropy reduction compared to the original correlation structure. Namely, the difference between the average entropy of the correlation substructures $æ_k$ (each rescaled to unit trace) and that of the overall system augmented by a penalty for over-partitioning $\propto \sqrt{N_F}$.

$$\text{TEFI} = \left(\frac{1}{N_F}\sum_{k=1}^{N_F}\mathcal{S}(æ_k) - \mathcal{S}(æ)\right) + \left(\left(\mathcal{S}(æ) - \sum_{k=1}^{N_F}\mathcal{S}(æ_k)\right)\sqrt{N_F}\right) \tag{3}$$

Following Golino et al. (2021) and Golino et al. (2024), TEFI evaluates how the entropy reduction decreases by adding a new dimension until the additional partition becomes redundant for entropy minimization. Thus, lower TEFI values reflect better structure quality. In addition, since TEFI accounts for partition entropy, TEFI can also evaluate the fit of a given dimensional structure without needing the knowledge of ground truth.

Representational Efficiency: Kolmogorov complexity (KC) quantifies the complexity of a system's topology (Golino et al., 2025). The KC of an arbitrary object is equivalent to the minimum possible length of the computer code necessary to reconstruct the object (Kolmogorov, 1965). Topologies with regular and parsimonious patterns are described more concisely compared to noisy networks. While the exact value of KC is uncomputable, KC can be approximated for networks using lossless compression algorithms. As implemented in Golino (2025) and Golino et al. (2025), the network KC estimator performs unweighted topology encoding—thresholding the network adjacency matrix to retain edges and assigning coordinate vectors (ij) and a constant retained-edge value $2^1 = 2$. The serialized edge list is then compressed using gzip (Morzy, Kajdanowicz, & Kazienko, 2017; Santoro & Nicosia, 2020). Because the compressed length can change depending on the sequence of edge coordinates, KC values are estimated as the average over 1000 random permutations of the edges.

3.3 Representational Maturity Index (RM)

An overall evaluation metric of a multivariate neural representation is defined as a linear combination of semantic coherence, structure quality and efficiency. Normalized Mutual Information scores are already bounded between 0 and 1, while TEFI is rescaled to fall between 0 and 1 by subtracting the TEFI value from the maximum $T_{\max}$ and dividing by the range $(T_{\max} - T_{\min})$, so that higher values indicate better structure. Likewise, KC is rescaled as $\frac{K_{\max} - \text{KC}_{m'nt}}{K_{\max} - K_{\min}}$, where m, ', n, and t refer to model, layer state, embedding dimension, and simulation iteration, respectively. Since TEFI and KC have no fixed bounds, their minima and maxima are the empirical values pooled across the entire set of models. Thus, the final formula reads:

$$\text{RM}_{m'nt} = 0.5\,\text{NMI}_{m'nt} + 0.3\,\frac{T_{\max} - \text{TEFI}_{m'nt}}{T_{\max} - T_{\min}} + 0.2\,\frac{K_{\max} - \text{KC}_{m'nt}}{K_{\max} - K_{\min}} \tag{4}$$

It follows from our definition that the score lies within $[0, 1]$ for all observed comparisons. In practice, we report both the composite RM and the individual components.

3.4 Geometric Motivation

A critical premise underlying CET is that of a semantic manifold. That is, activations of items belonging to semantic categories reside close to a manifold of low dimension m inside the ambient higher-dimensional hidden representation space $\mathbb{R}^k$. Classical embedding theorems guarantee under mild assumptions the existence of an embedding of the manifold using delay-coordinate maps of generic observations on the manifold (Takens, 1981; Whitney, 1936). Below we summarize the key points of our reasoning.

Theorem 1 (Semantic Manifold Embedding). Let $\mathcal{M} \subset \mathbb{R}^k$ denote a C^2-smooth compact manifold of dimension $m < k$, and let $f : \mathcal{M} \to \mathcal{M}$ be a smooth (C^2) diffeomorphism. Suppose furthermore that the communities $\{C_1, ..., C_r\}$ constitute a decomposition of the manifold, separated by a minimal pairwise distance margin of $\Delta > 0$. Then, for generic pairs (f, Œ), with Œ : $\mathcal{M} \to \mathbb{R}$ a C^2 observation function, and unit delay ø = 1, the delay embedding map $\Phi(x) = (Œ(x), Œ(f^{ø}(x)), ...Œ(f^{2mø}(x)))$ forms a C^1 diffeomorphism onto its image. In particular, Φ preserves cluster separations as follows:

$$\min_{i \neq j} \inf_{x \in C_i, y \in C_j} \|\Phi(x) - \Phi(y)\| \geq a\Delta > 0. \tag{5}$$

for some constant $a > 0$ depending only on Φ.

Our argument hinges on the following two facts. By compactness, injectivity and smoothness of the C^1 embedding $\Phi : \mathcal{M} \to \mathbb{R}^{2m+1}$, the smallest singular value of the Jacobian at any point is bounded below by a positive quantity ffi, ensuring bi-Lipschitz continuity with constants $a, b > 0$ satisfying:

$$ad(x, y) \leq \|\Phi(x) - \Phi(y)\| \leq bd(x, y) \qquad \text{for all } x, y \in \mathcal{M}. \tag{6}$$

Any bi-Lipschitz function preserves positive separation margins; consequently, if clustering procedures are robust to Lipschitz distortions, then they yield a correct clustering of the original manifold points. The theorem does not establish whether the community detection procedure used here is indeed robust to bi-Lipschitz distortions. However, our approach allows verifying such hypothesis empirically, namely by means of NMI analysis.

Remark. *The community detection problem itself does not involve any topological property per se, but rather a clustering problem on metric spaces. When, however, the clusters are guaranteed to belong to Δ-separated subsets of a compact manifold $\mathcal{M}$, delay-coordinate embeddings keep the cluster margins positive.*

4. Methods

4.1 CET Pipeline

Figure 1 shows as a schematic description of the complete CET-DynEGA workflow. For each language model under evaluation, the pipeline uses the following nine steps:

1. Item sampling: Let the item pool consist of approximately 50 items (textual statements) for each attribute (facet or dimension) of the target construct. From such a pool, five items per attribute are randomly sampled without replacement, resulting in a set of 25 items per iteration.
2. Hidden-state extraction: Each item is passed to the model as plain text. At each recorded state ‘, the token-level hidden states are averaged, such that every item is described by a single activation vector of dimension d, where d denotes the hidden size of the model.
3. Standardization: Each hidden unit is standardized across the stacked item-by-state observations, and only thereafter are the individual states separated for network estimation.
4. Time-delay embedding: For each state and each item, a time-delay embedding matrix with $n \in \{3, 4, \ldots, 200\}$ columns is constructed, wherein the hidden-unit index is treated as a pseudo-temporal ordering.

5. Derivative estimation (GLLA): The first-order derivatives are estimated from the embedding matrix of each item using Generalized Local Linear Approximation.
6. Network estimation (EBICglasso): The first-derivative vectors are column-bound across the 25 items, and a sparse Gaussian Graphical Model is estimated from their covariance matrix using EBIC graphical LASSO, i.e., te regularization parameter ˘ is selected over 100 values (by the one that minimizes EBIC), with an adaptive hyperparameter fl $\in \{0.50, 0.25, 0\}$, where fl is decreased whenever a node becomes disconnected, following Golino et al. (2020).
7. Community detection: The Louvain algorithm is applied to the estimated network in order to identify the dimensionality structure (communities in the network).
8. Metric computation: For every cell defined by a state and an embedding dimension, three indices are computed: NMI (with respect to the five stipulated attribute labels), TEFI, and KC.
9. Aggregation and maturity: Steps 1-8 are repeated across iterations (100 iterations per model in the matched comparison grid). The NMI, TEFI, and KC values are aggregated as arithmetic means across iterations, and RM is computed on the basis of the empirical TEFI and KC bounds pooled across all fourteen models.

4.2 Dynamic Exploratory Graph Analysis (DynEGA)

Dynamic Exploratory Graph Analysis (DynEGA) combines techniques of dynamical systems reconstruction with those of network psychometrics (Golino, 2026; Golino, Christensen, Moulder, et al., 2022; Golino et al., 2025). For a given layer, each item's activation vector $\mathbf{v}$ of dimensionality d, which corresponds to $\mathbf{v} = (\nu_1, \nu_2, \ldots, \nu_d)$, is considered as a sequence indexed by hidden-unit positions. With a delay of $\emptyset = 1$ and an embedding dimension of n, one obtains a matrix $\mathbf{X}^{(n)}$ of size $M \times n$ defined as

$$\mathbf{X}^{(n)} = \begin{bmatrix} \nu_1 & \nu_{1+\emptyset} & \cdots & \nu_{1+(n-1)\emptyset} \\ \vdots & \vdots & \ddots & \vdots \\ \nu_M & \nu_{M+\emptyset} & \cdots & \nu_{M+(n-1)\emptyset} \end{bmatrix}, \quad M = d - (n-1)\emptyset. \tag{7}$$

Derivatives can then be estimated using the Generalized Local Linear Approximation approach [GLLA; Boker et al. (2010); Deboeck et al. (2009)]:

$$\mathbf{Y} = \mathbf{X}\mathbf{L}(\mathbf{L}'\mathbf{L})^{-1}, \tag{8}$$

where $\mathbf{X} = \mathbf{X}^{(n)}$ and $\mathbf{L}$ represents a polynomial weight matrix of size $n \times (k+1)$ whose columns represent weight vectors associated with order-ff (ff $= 0, 1, \ldots, k$) derivatives defined as follows:

$$\mathbf{L}_{\text{ff}} = \frac{[\Delta t(\mathbf{u} - \bar{u})]^{\text{ff}}}{\text{ff!}}. \tag{9}$$

Here, $\Delta t = 1$, $\mathbf{u} = [1, 2, \ldots, n]'$, and $\bar{u}$ denotes the mean of $\mathbf{u}$ (Golino et al., 2025). For each item, extracting the derivative order-1 column and binding those of all 25 items into a single matrix allows estimating a sparse Gaussian Graphical Model, that is, a partial correlation network, through the graphical LASSO with the regularization parameter selected by the Extended Bayesian Information Criterion (EBICglasso). A community detection algorithm applied in the estimated network shows communities of densily connected nodes formed by items whose activation trajectories co-fluctuate across hidden coordinates. In contrast to the practice of using cross-sectional-based analysis that assumes the same informative value of each hidden coordinate, CET searches the landscape formed by the embedding dimension from $n = 3$ up to $n = 200$. This allows a systematic search of the activation landscape and the discovery of embedding representations where construct organization peaks (Golino, 2026).



Figure 1. CET-DynEGA procedure. (a) The vector representing each item's activations within a layer can be regarded as a univariate timeseries, which is indexed according to the hidden units' positions. (b) The time delay embedding matrix is built by sliding a window containing n sequential elements. (c) L matrix contains polynomial basis weights to estimate different order derivatives. (d) Through GLLA, embeddings transform into derivatives space, providing approximated estimations for values, velocities, and accelerations. (e) For every item's matrix Y, columns corresponding to first-order derivative estimates are extracted and concatenated column-wise. (f) With this concatenation of first-order derivative estimates, EGA performs network estimation and community detection; in addition, n varies from 3 to 200 in search for the embedding dimension that best reveals the construct structure.

Table 1. Large Language Models Evaluated

Model	Parameters	Blocks	Hidden size
GPT-2	117M	12	768
TinyLlama 1.1B	1.1B	22	2,048
Qwen2 1.5B	1.5B	28	1,536
Phi-2	2.7B	32	2,560
Phi-4-mini	3.8B	32	3,072
Qwen2.5 0.5B	0.5B	24	896
Qwen2.5 1.5B	1.5B	28	1,536
Qwen2.5 3B	3B	36	2,048
Qwen2.5 7B	7B	28	3,584
Qwen2.5 14B	14B	48	5,120
Qwen2.5 32B	32B	64	5,120
GPT-OSS 20B	20B	24	2,880
Qwen3.8 27B	27B	64	5,120
Muse-Glimmer 30B	30B	52	6,656

Note. Parameter counts denote nominal total model sizes rather than active parameters per token.

4.3 Models and Construct

Models. Fourteen open-weight language models are analyzed, spanning nominal parameter counts from 117M to 32B and depths from 12 to 64 decoder blocks (Table 1). Here, individual models rather than families constitute the experimental conditions. The six Qwen2.5-Instruct conditions cover a large range of scales from 0.5B up to 32B parameters, allowing a structured within-family comparison across scale, whereas the rest of the models introduce additional variation regarding depth, hidden dimensionality, attention mechanism, and training methodology: GPT-2, TinyLlama 1.1B, Qwen2 1.5B, Phi-2, Phi-4-mini, GPT-OSS 20B, Qwen3.8 27B, and Muse-Glimmer 30B. Each of the conditions of the former group relies on its respective Instruct checkpoint. The condition for TinyLlama uses its Chat-v1.0 version, and the condition for Phi-4-mini employs the instruction-tuned checkpoint. Condition Qwen3.8 refers to the stored Qwen3.8-27B BF16 GGUF identifier, with metadata specifying a Qwen3.5 text architecture made of 48 linear-attention blocks and 16 full-attention blocks (64 total). The Muse-Glimmer 30B architecture consists of 52 text blocks, of which 39 are sliding-attention blocks and 13 are full-attention blocks, in alternation. All evaluations examine text representations.

Construct and Item Sampling. Grandiose narcissism was conceptualized based on five characteristics derived from the construct definition suggested by Pincus and Lukowitsky (2010). These characteristics include: Distorting Disconfirming External Information; Exhibitionism; Intense Envy; Lack of Empathy; and Repression of Negative Self-information. During every iteration, five randomly selected items without replacement were taken from each characteristic domain, resulting in a dataset consisting of 25 items in total, distributed in five distinct domains. The advantage of this procedure resides on the fact that since the number of items per attribute remained the same across the fourteen models examined, the comparison across them is not affected by this variable.

4.4 Activation Extraction and Computation

Items were passed throught the models as plain text inputs without any conversational template structure. The average pooling of token-level hidden states produced an activation vector per item and recorded state. Prior to separating the states for network estimation, the columns (hidden units) of the stacked matrix of items x states underwent z-score standardization. The embedding layer output and the 12 decoder-layer outputs constitute the 13 recorded states of GPT-2. For all remaining architectures, the decoder-block outputs were recorded excluding the initial embedding layer. It must be noted that relative depth refers to the position of a state in the recorded sequence of each architecture, and therefore state 1 refers to a different operation in GPT-2 than in the remaining architectures. Finally, the last hidden state followed the native output-normalization conventions of each implementation. That means that, in some cases, the returned terminal state may differ from the pre-normalization output of the final decoder block.

Computations were carried out in float32 precision using the HuggingFace Transformers library. Specifically, activations were calculated on a CPU processor for GPT-2, TinyLlama, Qwen2, and Phi-4-mini. On the other hand, activations were computed on an NVIDIA A100 (80GB) GPU for the architectures Phi-2, Qwen2.5, GPT-OSS, Qwen3.8, and Muse-Glimmer, utilizing offloading to CPU memory whenever required. In particular, the publicly available MXFP4 quantized weights of GPT-OSS were initially dequantized in BF16 precision before conversion into float32 (without recovering the pre-quantized weights). In addition, Qwen3.8 was loaded from a BF16/F32 GGUF checkpoint and Muse-Glimmer was converted from the native safetensors into float32 weights. To investigate whether either the computation precision or the implementation framework affected the terminal profile of GPT-OSS, an additional reference run of 100 iterations was performed in OpenAI's official PyTorch codebase (where the MXFP4 quantized expert weights were dequantized to BF16 and computations were executed in BF16 precision with native float32 normalization parameters).

4.5 Simulation

In order to evaluate each of the models, simulations were performed using 198 embedding dimensions between 3 and 200, across all recorded states and across repeated item samples for each model. The simulation scale, involving all 495 total states (with 13 to 64 states per model), all 198 embedding dimensions, and all 100 iterations per model yields about **9,801,000** network estimates (Table 2). Automated validation of all results confirmed: complete state-by-embedding grids, uniqueness of iteration/state/embedding keys, finiteness of the primary metrics, as well as the range of NMI within $[0, 1]$.

4.6 Aggregation and cross-model comparison

To compare the results from all 14 models tested in our simulations, an aggregation process based on identical rules was adopted: (1) for each combination of state and embedding dimensions, we compute the arithmetic mean over iterations; (2) the overall mean Normalized Mutual Information (NMI) for a whole model must not treat correlated grid cells as independent replicates; hence, it must be computed as the mean of the iteration means, each taken across the whole state-by-embedding grid. 95% Student-t Monte Carlo (MC) intervals capture variation among different sets of items (iterations). Relative depths are computed for each state ℓ along a sequence of length L: $100(\ell - 1)/(L - 1)$. NMIs are averaged across embedding dimensions at each state to generate depth-curves, whereas RM trajectories are computed as the maximum over embedding dimensions of the mean RM-surface at each state, capturing maximum observed composite maturity per state. Heatmap representations keep track of all pairs state/emb, without binning the depth axis. A common RM normalization was employed, with empirical bounds pooled across all 14 models: TEFI $\in [-35.26579, 33.01256]$ and

Table 2. Simulation Scale Using Five Items per Attribute and 198 Embedding Dimensions

Model	States	Iterations
GPT-2	13	100
TinyLlama 1.1B	22	100
Qwen2 1.5B	28	100
Phi-2	32	100
Phi-4-mini	32	100
Qwen2.5 0.5B	24	100
Qwen2.5 1.5B	28	100
Qwen2.5 3B	36	100
Qwen2.5 7B	28	100
Qwen2.5 14B	48	100
Qwen2.5 32B	64	100
GPT-OSS 20B	24	100
Qwen3.8 27B	64	100
Muse-Glimmer 30B	52	100

Note. All models were assessed over embedding dimensions n = 3 to 200.

KC $\in [201.711, 676.193]$. Since RM is an affine function of its components, the mean cell-level RM can be estimated directly from mean cell-levels of NMI, TEFI, and KC.

Finally, two additional conventions are introduced to facilitate description of trajectory patterns, namely recovery onset and terminal behavior. The former corresponds to the first depth point wherein the mean NMI over embeddings reaches the critical value 0.5. This criterion should be regarded as a heuristical cutoff-value, instead of a statistical test. The terminal behavior of each trajectory is classified as near-zero if its end-state mean NMI is smaller than 0.05; otherwise, it is classified as retained.

5. Results

5.1 Construct Recovery Across Models

There was substantial variation in the average construct recovery even when keeping item numbers and embedding grids constant across the fourteen language models considered in the present investigation (see Table 3; Figure 2). Specifically, the overall mean NMI ranged widely between 0.382 (TinyLlama 1.1B) and 0.702 (Phi-2). Phi-2 obtained the largest mean NMI for the full grid, with a 95% MC interval of $[0.692, 0.713]$, which is clearly above the other models. The next highest scores belong to Qwen3.8 27B and Muse-Glimmer 30B, tied at a mean of 0.662, with intervals overlapping Qwen2.5 32B at 0.653, although, as shown below, these high-scoring models diverged markedly in entropy fit and terminal behavior. It should be noted that the ordering here refers to the full-grid recovery of grandiose narcissism under CET rather than to general benchmark capability.

Within the Qwen2.5-Instruct family, the overall mean NMI increased monotonically with parameter count: 0.504 (0.5B), 0.563 (1.5B), 0.601 (3B), 0.611 (7B), 0.629 (14B), and 0.653 (32B). Nevertheless, such ordering applies to the average over the full grid (encompassing early, intermediate, and terminal states) and *not* to the selected peak NMI across the six sizes: 0.848 (0.5B), 0.824 (1.5B), 0.895 (3B), 0.864 (7B), 0.900 (14B), and 0.899 (32B). Because parameter count, depth, hidden width, and training details covary across these checkpoints, this gradient documents a consistent

Table 3. Construct Recovery Across Model Conditions

Model	Mean NMI	95% MC interval	Peak NMI
GPT-2	0.396	[0.393, 0.400]	0.758
TinyLlama 1.1B	0.382	[0.375, 0.389]	0.883
Qwen2 1.5B	0.575	[0.567, 0.583]	0.879
Phi-2	0.702	[0.692, 0.713]	0.899
Phi-4-mini	0.451	[0.442, 0.461]	0.875
Qwen2.5 0.5B	0.504	[0.495, 0.513]	0.848
Qwen2.5 1.5B	0.563	[0.553, 0.573]	0.824
Qwen2.5 3B	0.601	[0.592, 0.610]	0.895
Qwen2.5 7B	0.611	[0.601, 0.620]	0.864
Qwen2.5 14B	0.629	[0.619, 0.639]	0.900
Qwen2.5 32B	0.653	[0.643, 0.663]	0.899
GPT-OSS 20B	0.463	[0.456, 0.470]	0.891
Qwen3.8 27B	0.662	[0.652, 0.671]	0.922
Muse-Glimmer 30B	0.662	[0.651, 0.673]	0.916

Note. Mean NMI averages all recorded states and embedding dimensions within each iteration before averaging across iterations; 95% MC intervals reflect Monte Carlo variability across iterations. Peak NMI is the maximum of the iteration-averaged state-by-embedding surface.

within-family association rather than a controlled scaling law.

5.2 Emergence Trajectories and Representational Landscapes

In order to investigate the evolution of construct organization along depth, Figure 3 shows the overlay of the fourteen globally normalized RM trajectory lines on shared axes. In addition, Figure 4 displays individual panels containing each model's trajectory line. Across nearly every architecture, representational maturity shows a steep increase starting from early layers toward a broad intermediate plateau region, often overlapping a reference 40–80% relative-depth interval, after which the trajectories diverge from one another close to the output. Importantly, neither the onset of emergence nor the location of peak maturity is constant across models. With respect to recovery onset, the relative depths ranged from 12.9% (block 5 of 32 in Phi-2) up to 52.4% (block 12 of 22 in TinyLlama 1.1B) according to the 0.50 mean-NMI criterion. Interestingly enough, despite the differences in total architectural depth (24–64 blocks), all six Qwen2.5 models reached onset within a tight relative band between 19.0% and 25.9% depth.

Peak composite maturity was not consistent either: GPT-2 reached peak maturity at state 8 of 13, Phi-2 at block 20 of 32, Muse-Glimmer 30B at block 38 of 52, and Qwen2.5 32B as late as block 52 of 64 (outside the 40–80% band). Any fixed "middle-layer" heuristic would therefore miss the optimal composite states in multiple models.

Figures 5 and 6 expand the one-dimensional depth curves of mean RM and mean NMI into full two-dimensional state-by-embedding landscapes. In addition, Figures D1–D3 in Appendix D display the component landscapes of TEFI and KC, and the depth-averaged NMI. By separating NMI from the composite, one may appreciate how sharply localized the recovery of the construct can be across embedding dimensions. The maximum mean-cell NMI across all models was obtained

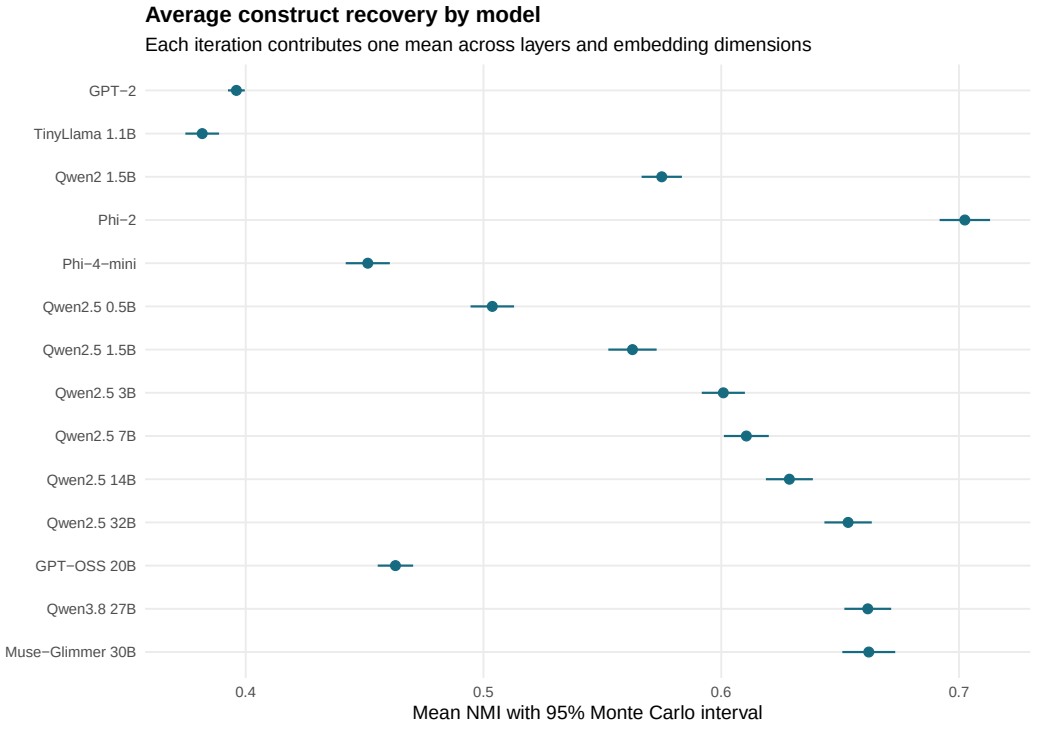

Figure 2. Average construct recovery by model. Each iteration contributes one mean across recorded states and embedding dimensions; points average these iteration means and intervals show 95% Monte Carlo uncertainty. Every model uses five items per attribute and the same aggregation.

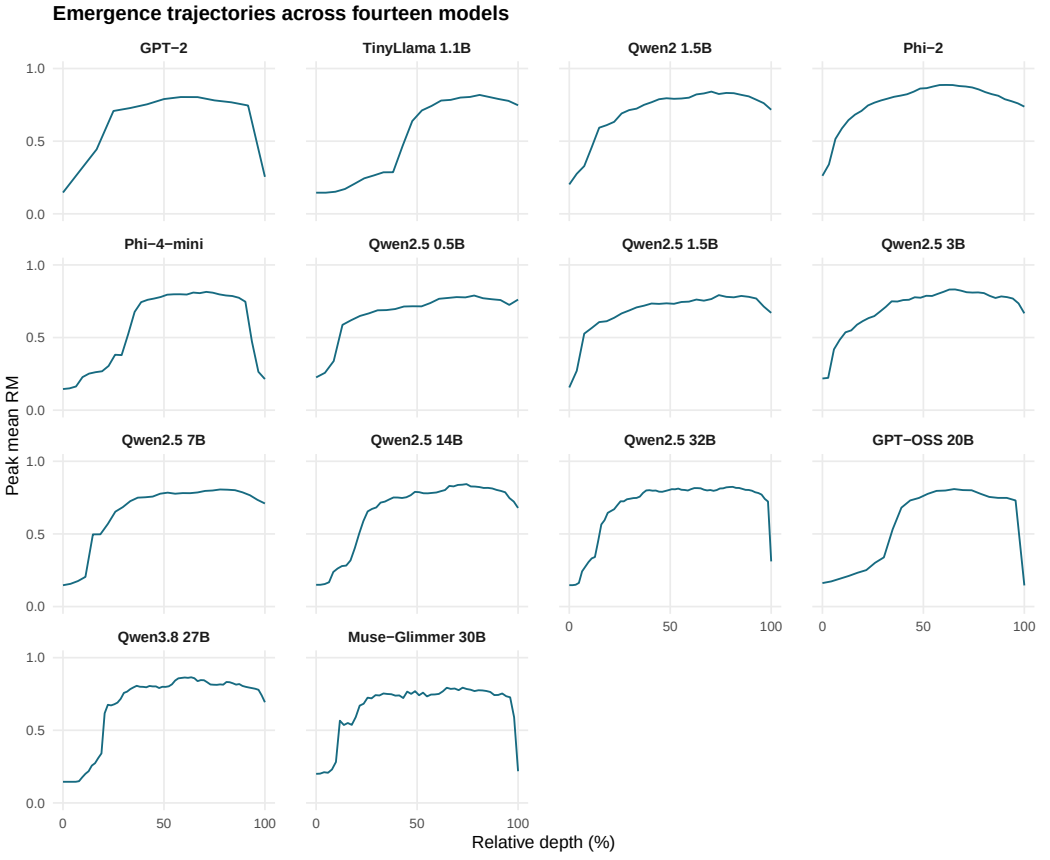

Figure 3. Construct emergence trajectories across fourteen models on common axes. Each point represents the maximum, over embedding dimensions, of mean RM at one recorded state. All curves use five items per attribute and the same TEFI and KC normalization pooled across the fourteen models. The shaded 40–80% depth interval marks the reference Optimal Abstraction Zone; individual model optima may lie outside this interval. Relative depth includes the embedding output for GPT-2.

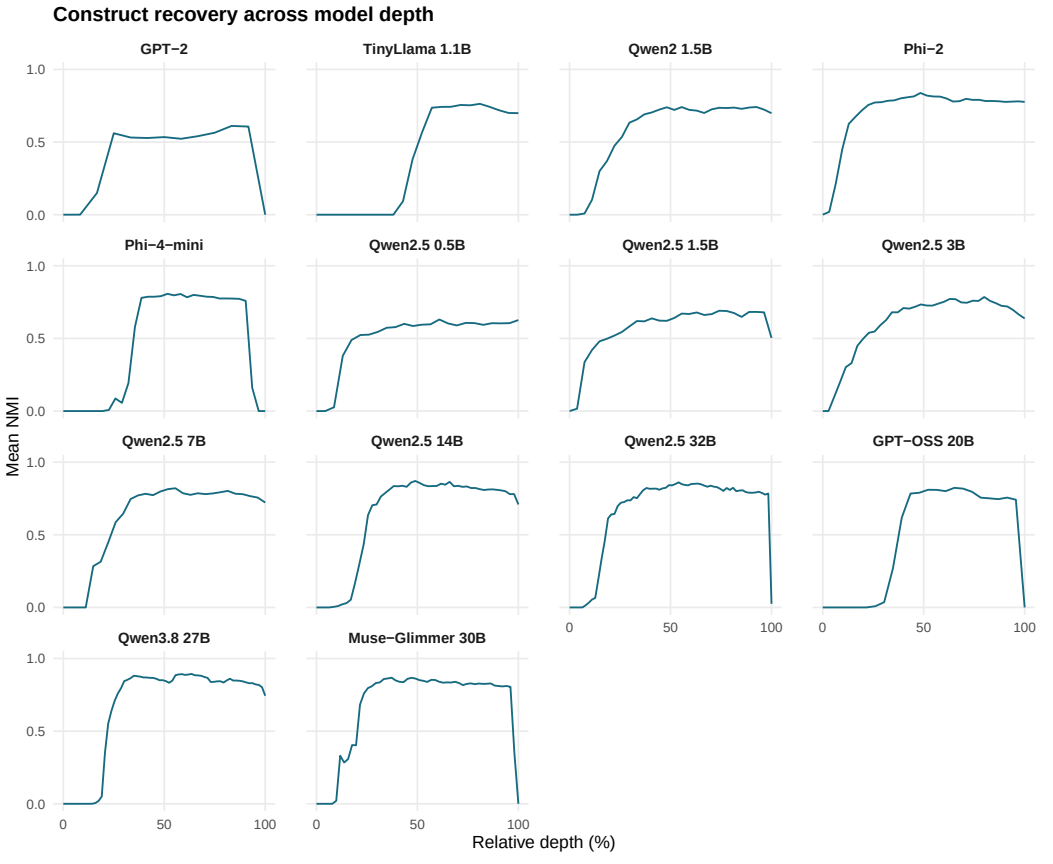

Figure 4. Emergence trajectories across fourteen models. Each curve shows the maximum, over embedding dimensions, of mean RM at each recorded state using pooled TEFI and KC normalization. Panels share depth and maturity axes.

for Qwen3.8 27B (0.922, block 37; n = 13), closely followed by Muse-Glimmer 30B (0.916, block 26; n = 9). However, those two top-scoring models illustrate the central gap between intermediate coherence and preservation at the output state. While Qwen3.8 retained a high final-state mean NMI of 0.744, the NMI of Muse-Glimmer dropped to exactly zero at its last block.

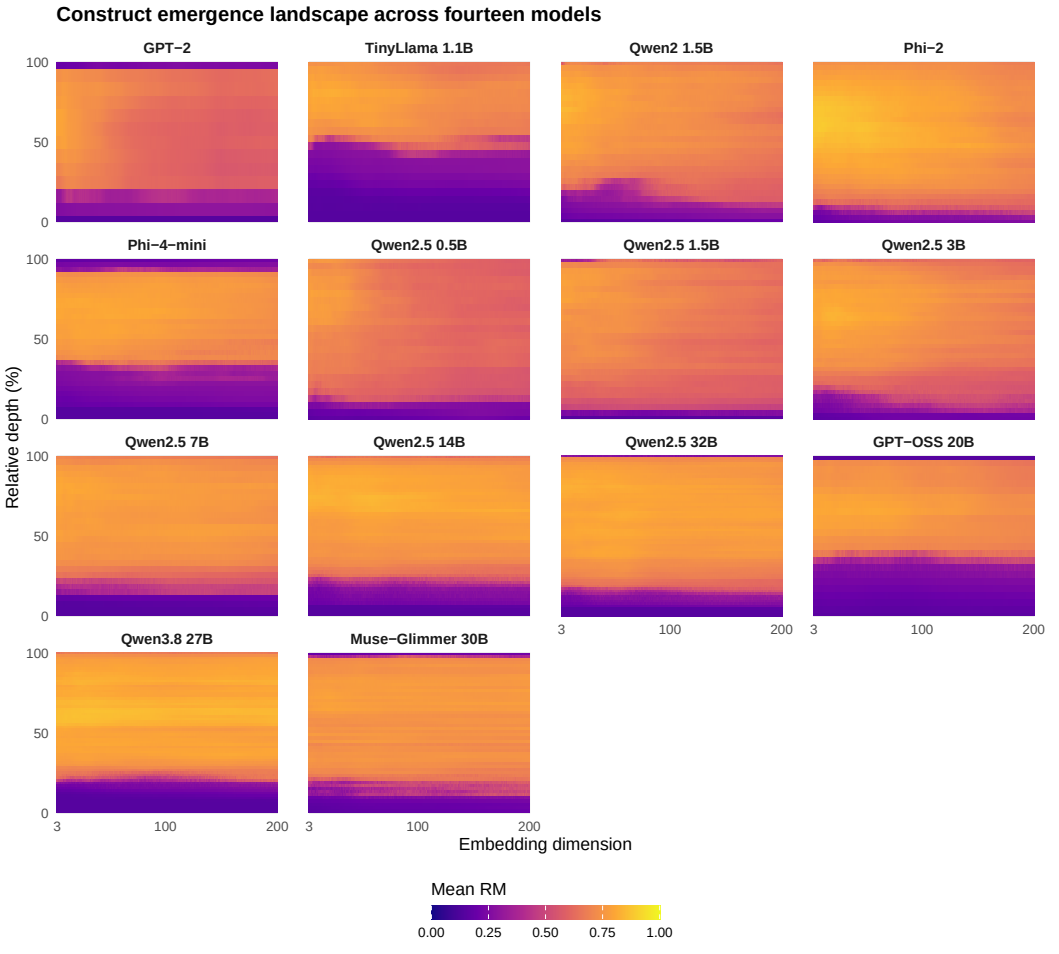

Figure 5. Construct emergence landscape across fourteen models. Each cell shows mean RM at a recorded state and embedding dimension using the same pooled normalization as Figures 3 and 4. States are placed on a relative-depth axis without binning, preserving narrow terminal bands.

5.3 Emergence Archetypes and Model-Specific Signatures

Rather than following a single canonical trajectory, the fourteen models cluster into four recurring developmental signatures.

5.3.1 Early-onset, sustained-recovery models (Phi-2, Qwen2 1.5B, Qwen3.8 27B, and Qwen2.5 0.5B–14B).

Phi-2 exemplifies this profile well. It crossed the 0.50 onset threshold earliest in relative depth (block 5 of 32), peaked in cell NMI at 0.899 (block 16), attained optimal TEFI and peak RM at block 20, and sustained a final-state mean NMI of 0.777—explaining its unmatched full-grid average (0.702).

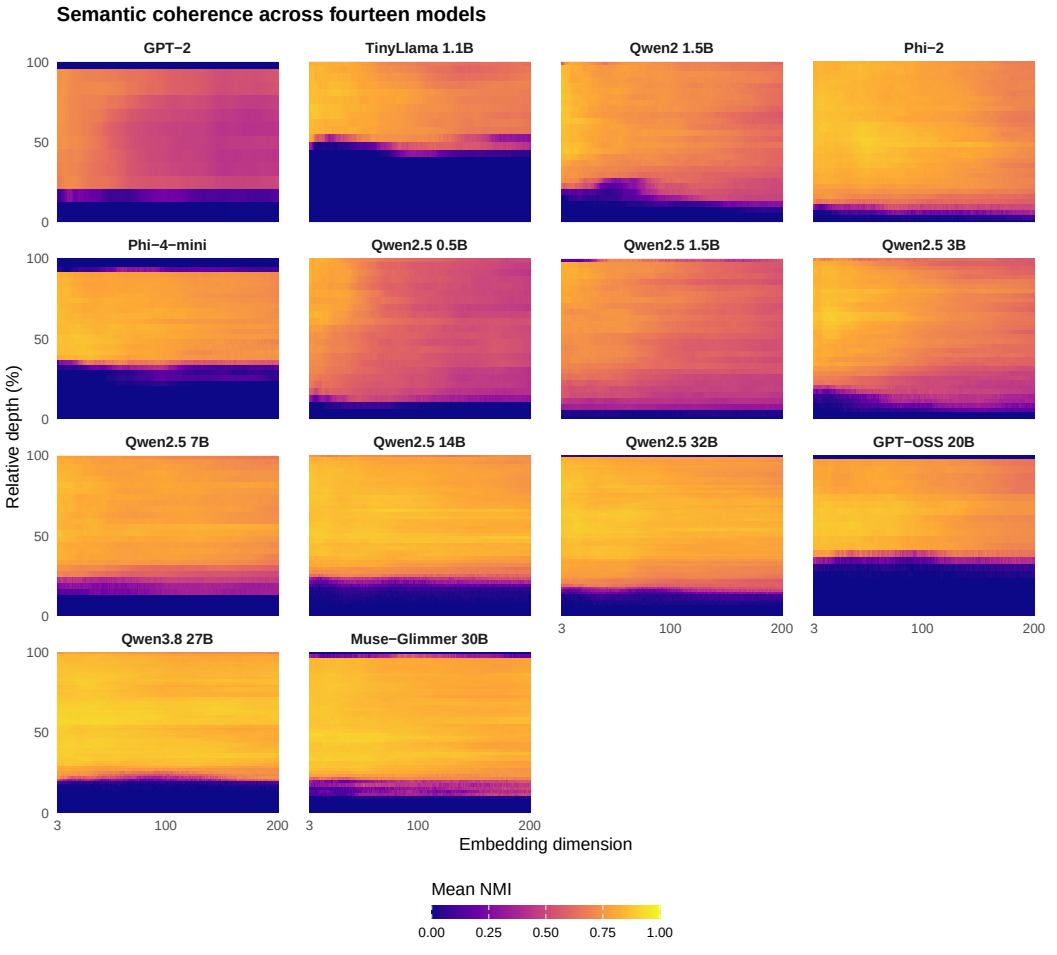

Figure 6. Semantic coherence across fourteen models. Cells show arithmetic-mean NMI across iterations for each recorded state and embedding dimension on a common color scale and relative-depth axis.

Qwen3.8 27B similarly combined early relative onset (block 15 of 64) with the highest selected cell NMI (0.922 at block 37), peaking in mean NMI and RM at block 41, optimizing TEFI at block 42, and retaining 0.744 at the final block despite a modest late decline (from 0.803). Qwen2 1.5B reached onset at block 8, attained simultaneous optima for cell NMI (0.879), TEFI, and RM at block 20, peaked in embedding-averaged NMI at block 26 (indicating widening of the favorable embedding region with depth), and retained 0.699 at block 28. Among Qwen2.5 0.5B through 14B, all five retained strong terminal recovery (0.504 to 0.722), though their local optima varied—for instance, Qwen2.5 0.5B attained its peak cell NMI (0.848) in its final block despite peaking in mean NMI at block 15, while Qwen2.5 1.5B, 3B, 7B, and 14B peaked in cell NMI at blocks 26 (0.824), 24 (0.895), 15 (0.864), and 32 (0.900), respectively.

5.3.2 Delayed-onset, high-retention models (TinyLlama 1.1B).

TinyLlama showed the latest onset across all fourteen models (block 12 of 22; 52.4% depth). Once construct structure emerged, however, it was robust: cell NMI reached 0.883 at block 16, mean NMI, TEFI, and RM converged at block 18, and final-state mean NMI remained high at 0.699. Its low full-grid mean (0.382) therefore reflects an extended low-recovery early segment rather than an inability to represent the construct.

5.3.3 High-intermediate, sharp-terminal-collapse models (GPT-2, Phi-4-mini, Qwen2.5 32B, GPT-OSS 20B, and Muse-Glimmer 30B).

Five architectures built strong intermediate construct representations only to lose CET-recoverable alignment at the output. GPT-2 reached onset at state 4, achieved optimal TEFI and RM at state 8, peaked in cell NMI at state 9 (0.758) and depth-averaged NMI at state 11, maintained 0.607 at state 12, and dropped to 0.000 at state 13. Phi-4-mini crossed onset at block 12, peaked in cell NMI at block 17 (0.875), reached peak RM at block 23 and best TEFI at block 25, and fell to zero across its final two blocks. Muse-Glimmer 30B reached onset early (block 12 of 52), peaked in mean NMI at block 20, cell NMI at block 26 (0.916), and both TEFI and RM at block 38, before dropping to 0.347 at block 51 and 0.000 at block 52. Strikingly, Qwen2.5 32B broke sharply from its smaller siblings: despite matching Qwen2.5 14B in peak cell NMI (0.899 at block 37 vs. 0.900 at block 32) and achieving a higher overall mean (0.653 vs. 0.629), Qwen2.5 32B plunged from 0.784 at block 63 to 0.025 at block 64 (whereas 14B retained 0.709).

5.3.4 Precision-invariant terminal collapse (GPT-OSS 20B reference check).

GPT-OSS 20B crossed onset at block 10 of 24, peaked in cell NMI at block 14 (0.891), mean NMI and RM at block 16, and TEFI at block 18, maintaining 0.742 at block 23 before collapsing to 0.000 at block 24. Crucially, the independent 100-iteration OpenAI BF16 reference run closely reproduced this trajectory: overall mean NMI remained 0.463, and final-state NMI was zero across all 19,800 terminal configurations. While implementation and precision changed jointly—and both runs evaluated the released MXFP4-derived weights—this replication rules out float32 conversion alone as the cause of the terminal drop.

5.4 Two Terminal Regimes

Applying the 0.05 final-state mean-NMI threshold separates the fourteen models cleanly into five *near-zero* and nine *retained* terminal regimes (Table 4). Within the near-zero group, GPT-2, Phi-4-mini, GPT-OSS 20B, and Muse-Glimmer 30B registered exactly 0.000 across the entire final-state embedding grid, whereas Qwen2.5 32B dropped sharply to a mean of 0.025 while retaining a modest localized maximum of 0.120. Within the retained group, final-state mean NMI ranged from 0.504 (Qwen2.5 1.5B) to 0.777 (Phi-2). Retention did not imply stability without attenuation: Qwen2.5

Table 4. Final-State NMI and Terminal Regime Classification

Model	Final mean	Final maximum	Regime
GPT-2	0.000	0.000	Near-zero
TinyLlama 1.1B	0.699	0.829	Retained
Qwen2 1.5B	0.699	0.820	Retained
Phi-2	0.777	0.842	Retained
Phi-4-mini	0.000	0.000	Near-zero
Qwen2.5 0.5B	0.626	0.848	Retained
Qwen2.5 1.5B	0.504	0.734	Retained
Qwen2.5 3B	0.638	0.771	Retained
Qwen2.5 7B	0.722	0.807	Retained
Qwen2.5 14B	0.709	0.778	Retained
Qwen2.5 32B	0.025	0.120	Near-zero
GPT-OSS 20B	0.000	0.000	Near-zero
Qwen3.8 27B	0.744	0.813	Retained
Muse-Glimmer 30B	0.000	0.000	Near-zero

Note. Final mean averages NMI across embedding dimensions and iterations at the final recorded state; Final maximum is the maximum of the iteration-averaged NMI across embedding dimensions. Near-zero denotes final mean NMI below 0.05. A zero score reflects loss of network-community agreement under CET rather than proof that semantic information has been erased.

1.5B declined from 0.680 to 0.504 and Qwen3.8 27B declined from 0.803 to 0.744. Terminal behavior is therefore continuous within broad regimes and does not follow parameter count—comparing the largest models alone, 32B Qwen2.5 and 30B Muse-Glimmer collapsed while 27B Qwen3.8 retained 0.744.

6. Discussion

The main goal of the present work was to explore how the organization of a specific multidimensional construct unfolds within different LLMs across transformer depths, via a novel methodology called Construct Emergence Tracing (CET). Overall mean Normalized Mutual Information (NMI), selected peak coherence (based on mean NMI), entropy-based organization, composite maturity score, and terminal retention were assessed in fourteen different LLMs, which differed in architectures, tokenizers, attention designs, training data, and checkpoint precision. Results revealed several key points. Across all Qwen2.5-Instruct checkpoints tested, full-grid NMI scaled monotonically from 0.5B to 32B, although the peak coherence did not increase similarly and construct recovery collapsed precisely in the output layer of the largest checkpoint. On the contrary, TinyLlama obtained modest mean NMI due to its late emergence phase but achieved remarkable upper-layer recoveries, while Muse-Glimmer had second-highest peak coherence value (0.916), falling to zero at its terminal state. Conceptually speaking, CET allows to answer questions regarding where, when, and how coherently the organization of an abstract multidimensional construct emerges in internal representations. While existing methodologies such as feature attribution and circuit tracing focus on the bottom-up approach, starting from the neural mechanism toward constructing its interpretation,

CET takes the exact opposite route by examining whether an externally defined construct structure emerges in transformer internals. Rather than conflicting with traditional mechanistic interpretability approaches, CET serves as a map directing where the investigation of mechanism needs to start based on construct emergence dynamics.

6.1 Item Generation, Embeddings, and Construct Recovery

Advances made in the field of Generative Psychometrics have revolutionized how psychologists develop scales for multidimensional psychological constructs. Now, researchers can go from defining a construct theoretically to having a curated item pool, assessing whether the semantics used in the scale statements are redundant and how well they capture the correct dimensionality of the construct before collecting human responses (Garrido et al., 2026; Russell-Lasalandra et al., 2024; Russell-Lasalandra & Golino, 2026). Conceptually speaking, there are two distinctions that must be highlighted. First, the semantic relationships between item statements within the LLM embeddings are different from covariance patterns among individual differences in human responses; thus, embedding similarities demonstrate only conceptual coherence and not the validity of an instrument in capturing the intended construct.

Second, the way researchers analyze embeddings significantly influences which latent structure gets reconstructed. In a previous study conducted by Garrido et al. (2026), researchers explored the impact of the use of EGA vs. PCA in dimensionality assessments on embeddings obtained from three different item-generating LLMs and two different embedding models, including after network-based item reduction. The findings showed how PCA methods with either Kaiser's rule or parallel analysis tended to overestimate the actual number of dimensions, while EGA was able to recover the predefined structure far more accurately and aligned better than PCA with the dimensionality obtained using human respondents' ratings in Multidimensional Schizotypy Scale. Reducing the variance dimensionality of an embedding space and recovering the latent dimensionality of a psychological construct are different tasks, and CET targets the latter.

6.2 A developmental view of construct emergence

Understanding why the organization of representations plays such an important role in representing constructs is fundamental for several reasons. A developmental view of construct emergence can be adopted from developmental psychology (Golino, 2025). According to Piaget's notion of reflective abstraction, concepts of ever-higher levels emerge not by merely building up separate elements but by organizing already-existing representations into relational structures. In Construct Emergence Tracing (CET), statements phrased in diverse ways gradually form attribute-consistent communities as representations pass through successive attention and feed-forward transformations, which makes this relational coordination measurable. This developmental process does not imply, however, equating transformer layers with specific childhood cognitive development stages since transformers represent machine learning architectures.

According to neo-Piagetian theories of hierarchical integrative processes of development (Pascual-Leone's theory of attention/mental capacity, Case, Fischer, and Demetriou's hierarchical integration approach; summarized in Golino (2025)), what is crucial in representing multidimensional constructs is how information becomes effectively organized. The monotonic increase in full-grid recovery along with increasing model size (Qwen2.5) is consistent with this view of efficient representation organization; however, parameter count is not an analogue of human attentional capacity. In addition, the decoupling of NMI and TEFI optima warns against the idea of a rigid, unilinear stage sequence, since coherence with external construct labels and minimizing internal entropy do not always advance together across transformer layers.

According to Vygotsky's socio-cultural theory, meaning is created in socially and linguistically mediated interactions (see Golino (2025)). Because LLMs are trained on human-written text data from

specific linguistic backgrounds, the dimensional structure obtained through CET partially reflects shared meanings of personality traits within particular cultures. Although training language models via pretraining differs dramatically from early-childhood learning experiences, an epistemological caveat should always be considered: one cannot consider construct trajectories obtained from the analysis of a given English-language item pool as universal without conducting additional testing across languages, cultures, and independently authored items.

Finally, it is critical to recognize that developmental trajectories analyzed here occur along processing depth through trained checkpoints rather than development through time in the course of training. Terminal collapses suggest that maximal processing depth is not equivalent to optimal separation in abstract multidimensional constructs' representations. Investigating the development of CET-based representation organization through training checkpoints allows one to address this limitation in the future.

6.3 Coherence, organization, and maturity

Do transformer architectures first develop structural organization and then semantic coherence, or vice versa? Results clearly illustrate that no universal temporal precedence exists. Comparing NMI and TEFI based on embedding-averaged results, we find that the lowest entropy in terms of TEFI (i.e., state 8) occurred earlier than the maximum mean NMI (i.e., state 11) in GPT-2. On the contrary, Phi-4-mini showed the reverse sequence (maximum NMI at block 17; lowest TEFI at block 25); Qwen2.5 32B showed a similar trend (maximum NMI at block 35; minimum TEFI at block 51); Muse-Glimmer 30B (maximum NMI at block 20; minimum TEFI at block 38); TinyLlama reached both maximum NMI and minimum TEFI simultaneously (at block 18).

Such discrepancies demonstrate that the Representational Maturity Index should accompany its components and not be treated as a replacement for them; e.g., Qwen3.8 27B and Muse-Glimmer 30B showed the same overall mean NMI value (0.662), yet they had mean TEFI values that differed nearly twofold (−11.01 vs. −5.85) and opposite terminal outcomes. Similarly, KC is a valuable indicator reflecting the degree of topologic regularity in partial correlation graphs but since this measure equally treats edges' presence without considering their strengths, compressibility alone cannot reflect a mature concept. Thus, treating RM as a weighted descriptive summary and exploring raw NMI, TEFI, and KC scores allows one to avoid obscuring structural compromises.

6.4 Terminal recovery and representation selection

One of the most consequential findings concerns the terminal states. In GPT-OSS 20B, mean NMI dropped from 0.742 in the penultimate block to 0.000 in the final state; in Qwen2.5 32B, it fell from 0.784 to 0.025; and in Muse-Glimmer 30B, Phi-4-mini, and GPT-2, the final-state NMI vanished completely. Had these models been evaluated only via their final hidden states, as is standard in API-level embedding pipelines, the conclusions about their representation of the construct would have been misleading.

However, terminal collapse under CET does *not* indicate that semantic information was destroyed. Final layer normalization near the language modeling head, residual streams' reorientation, or projections toward the vocabulary-prediction space can alter correlation structures and make community recovery ineffective while leaving semantic information recoverable via alternative methods. Controlled comparative investigations of pre-normalization block outputs, returned hidden states, and alternative probes would help us determine whether the observed phenomenon reflects output normalization or deep transformation of representation geometry.

6.5 Implications for model comparison, interpretability, and AI safety

Methodologically, CET complements other representational analysis approaches by shifting the perspective of layer selection from locating a single best layer to analyzing the developmental landscape.

With CET, one can easily differentiate between isolated high-scoring cells and consistent plateaus of good quality; additionally, one can investigate the impact of evaluation criteria on optimal layer choices (i.e., NMIs for coherence, TEFIs for entropy reduction, and RMs for maturity). Combining CET with linear probing and Centered Kernel Alignment [CKA; Kornblith et al. (2019)] constitutes powerful triangulation since convergent results strengthen representational hypotheses and divergent results identify sensitivities to linear versus non-linear geometry, to network regularization, or to the difference between feature presence and multidimensional organization.

For AI safety applications, CET provides a promising way of tracking the evolution of safety-relevant multidimensional concepts (such as deception, sycophancy, or harmful intent) through transformer depths prior to and subsequent to alignment interventions such as reinforcement learning from human feedback or steering approaches. Realizing this potential requires, however, substantial psychometric groundwork. First, independently validated item pools and corresponding negative controls need to be developed; and the relationships between representational changes and model behavior should be rigorously documented through empirical observation.

6.6 Limitations and future directions

This study includes several major limitations and opportunities for future research. First, fourteen studied language models exhibit differences regarding architecture design, tokenizer configuration, attention mechanism specifics, training corpus, and checkpoint precision; in addition, six out of fourteen checkpoints correspond to a single release family (i.e., Qwen2.5). Thus, this study characterizes specific models and cannot isolate the causal influence of certain factors (e.g., scale, model quantization, or hybrid attention) on construct representations' emergence unless directly compared under matching conditions.

Second, this work focused exclusively on grandiose narcissism represented by a single item pool. Surface-level lexical regularities might contribute to clustering, and recovering the stipulated labels does not guarantee genuine semantic comprehension. Therefore, future studies must validate CET's applicability through other psychological constructs, independently reworded item formulations, cross-linguistic translation, random permutations of external labels, and matched nonsensical control constructs.

Third, treating the hidden unit index as pseudo-time during time delay embeddings represents a practical strategy rather than a literal physical time series. Though Theorem 1 establishes sufficient conditions for separation preservation on compact manifolds, it does not guarantee that all hidden layers satisfy them. Additional investigations involving permutations of hidden unit coordinates, variations of embedding range, community detection algorithms, and methods of activation standardization are necessary.

Fourth, the RM calculation depends on chosen weighting coefficients and pooled normalization bounds, selected grid peaks are exploratory maxima, and chosen thresholds (0.50; 0.05) are heuristic conventions. Identifying the underlying causes of emergence and terminal loss necessitates further activation patching experiments, matched analyses of pre- and post-normalization outputs, and convergent probing assessments.

7. Conclusion

To conclude, this manuscript introduced CET, which analyzes construct representations' emergence along the depth dimension in LLMs. Analyzing fourteen language models ranging in parameter counts from 117M to 32B, we have demonstrated clear divergence across models in emergence onset points, average representational quality, and alignment between coherence- and entropy-based metrics. Moreover, CET highlighted the critical phenomenon of terminal collapse in five of the fourteen models despite relatively high intermediate NMI scores. Our key finding is that neither a

single selected peak nor a final hidden state alone suffices to characterize how a model represents a construct.

Acknowledgements This work was supported by a grant from The Jefferson Trust. The Construct Emergence Tracing pipeline uses the HuggingFace Transformers library for activation extraction and the R packages EGAnet and aricode for network estimation and metric computation.

Competing Interests The author reports there are no competing interests to declare.

Author Contributions Hudson Golino: Conceptualization, Methodology, Software, Formal Analysis, Writing – Original Draft.

References

Alain, G., & Bengio, Y. (2017). Understanding intermediate layers using linear classifier probes. *5th International Conference on Learning Representations, Workshop Track*.

Belinkov, Y. (2022). Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, *48*(1), 207–219.

Boker, S. M., Deboeck, P. R., Edler, C., & Keel, P. (2010). Generalized local linear approximation of derivatives from time series. In S.-M. Chow, E. Ferrer, & F. Hsieh (Eds.), *Statistical methods for modeling human dynamics: An interdisciplinary dialogue* (pp. 161–178). Routledge/Taylor & Francis Group.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., Arx, S. von, et al. (2021). *On the opportunities and risks of foundation models*. https://doi.org/10.48550/arXiv.2108.07258

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., et al. (2020). *Language models are few-shot learners*. https://doi.org/10.48550/arXiv.2005.14165

Christensen, A. P., Garrido, L. E., Guerra-Peña, K., & Golino, H. (2024). Comparing community detection algorithms in psychometric networks: A Monte Carlo simulation. *Behavior Research Methods*, *56*(3), 1485–1505.

Christensen, A. P., & Golino, H. (2021). Estimating the stability of psychological dimensions via bootstrap exploratory graph analysis: A Monte Carlo simulation and tutorial. *Psych*, *3*(3), 479–500.

Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. *Psychological Bulletin*, *52*(4), 281–302. https://doi.org/10.1037/h0040957

Deboeck, P. R., Montpetit, M. A., Bergeman, C. S., & Boker, S. M. (2009). Using derivative estimates to describe intraindividual variability at multiple time scales. *Psychological Methods*, *14*(4), 367–386. https://doi.org/10.1037/a0016622

Epskamp, S., Maris, G., Waldorp, L. J., & Borsboom, D. (2018). Network psychometrics. In P. Irwing, T. Booth, & D. J. Hughes (Eds.), *The Wiley handbook of psychometric testing: A multidisciplinary reference on survey, scale and test development* (pp. 953–986). John Wiley & Sons. https://doi.org/10.1002/9781118489772.ch30

Epskamp, S., Waldorp, L. J., Mõttus, R., & Borsboom, D. (2018). The Gaussian graphical model in cross-sectional and time-series data. *Multivariate Behavioral Research*, *53*(4), 453–480. https://doi.org/10.1080/00273171.2018.1454823

Friedman, J., Hastie, T., & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *9*(3), 432–441. https://doi.org/10.1093/biostatistics/kxm045

Garrido, L. E., Russell-Lasalandra, L. L., Rodríguez-Montoya, C., & Golino, H. (2026). *Estimating dimensional structure in generative psychometrics: Comparing PCA and network methods using large language model item embeddings*. https://doi.org/10.31234/osf.io/2s7pw_v2

Golino, H. (2025). *What I learned with John: On the depth of language and how to measure it with large language models and algorithm (Kolmogorov) complexity*. https://doi.org/10.31234/osf.io/b92n5_v4

Golino, H. (2026). Optimizing the landscape of LLM embeddings with dynamic exploratory graph analysis for generative psychometrics: A Monte Carlo study. *arXiv Preprint arXiv:2601.17010*.

Golino, H., Christensen, A. P., & Garrido, L. E. (2022). Exploratory graph analysis in context. *Revista Psicologia: Teoria e Prática*, *24*(3), ePTPPA14197.

Golino, H., Christensen, A. P., Moulder, R., Kim, S., & Boker, S. M. (2022). Modeling latent topics in social media using dynamic exploratory graph analysis: The case of the right-wing and left-wing trolls in the 2016 US elections. *Psychometrika*, *87*(1), 156–187.

Golino, H., & Epskamp, S. (2017). Exploratory graph analysis: A new approach for estimating the number of dimensions in psychological research. *PLoS ONE*, *12*(6), e0174035. https://doi.org/10.1371/journal.pone.0174035

Golino, H., Jiménez, M., Garrido, L. E., & Christensen, A. P. (2024). *Generalized total entropy fit index: A new fit index to compare bifactor and correlated factor structures in SEM and network psychometrics*. https://doi.org/10.31234/osf.io/5g3hb

Golino, H., Moulder, R., Shi, D., Christensen, A. P., Garrido, L. E., Nieto, M. D., et al. (2021). Entropy fit indices: New fit measures for assessing the structure and dimensionality of multiple latent variables. *Multivariate Behavioral Research*, *56*(6), 874–902.

Golino, H., Nesselroade, J., & Christensen, A. P. (2025). Toward a psychology of individuals: The ergodicity information index and a bottom-up approach for finding generalizations. *Multivariate Behavioral Research*, *60*(3), 528–555. https://doi.org/10.1080/00273171.2025.2454901

Golino, H., Shi, D., Christensen, A. P., Garrido, L. E., Nieto, M. D., Sadana, R., … Martinez-Molina, A. (2020). Investigating the performance of exploratory graph analysis and traditional techniques to identify the number of latent factors: A simulation and tutorial. *Psychological Methods*, *25*(3), 292–320. https://doi.org/10.1037/met0000255

Jawahar, G., Sagot, B., & Seddah, D. (2019). What does BERT learn about the structure of language? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3651–3657. https://doi.org/10.18653/v1/P19-1356

Kolmogorov, A. N. (1965). Three approaches to the definition of the concept "quantity of information." *Problemy Peredachi Informatsii*, *1*(1), 3–11.

Kornblith, S., Norouzi, M., Lee, H., & Hinton, G. (2019). Similarity of neural network representations revisited. *Proceedings of the 36th International Conference on Machine Learning*, 3519–3529.

Kriegeskorte, N., Mur, M., & Bandettini, P. (2008). Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, *2*, 4. https://doi.org/10.3389/neuro.06.004.2008

Lindsey, J., Gurnee, W., Ameisen, E., et al. (2025). *On the biology of a large language model*. Retrieved from https://transformer-circuits.pub/2025/attribution-graphs/biology.html

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *30*, 4765–4774.

Massara, G. P., Di Matteo, T., & Aste, T. (2016). Network filtering for big data: Triangulated maximally filtered graph. *Journal of Complex Networks*, *5*(2), 161–178. https://doi.org/10.1093/comnet/cnw015

Morzy, M., Kajdanowicz, T., & Kazienko, P. (2017). On measuring the complexity of networks: Kolmogorov complexity versus entropy. *Complexity*, *2017*, 1–12. https://doi.org/10.1155/2017/3250301

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., & Carter, S. (2020). Zoom in: An introduction to circuits. *Distill*, *5*(3), e00024.

Pincus, A. L., & Lukowitsky, M. R. (2010). Pathological narcissism and narcissistic personality disorder. *Annual Review of Clinical Psychology*, *6*(1), 421–446.

Raghu, M., Gilmer, J., Yosinski, J., & Sohl-Dickstein, J. (2017). SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in Neural Information Processing Systems*, *30*.

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, *8*, 842–866. https://doi.org/10.1162/tacl_a_00349

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536. https://doi.org/10.1038/323533a0

Russell-Lasalandra, L. L., Christensen, A. P., & Golino, H. (2024). *Generative psychometrics via AI-GENIE: Automatic item generation and validation via network-integrated evaluation*. https://doi.org/10.31234/osf.io/fgbj4

Russell-Lasalandra, L. L., & Golino, H. (2026). Prompt engineering for scale development in generative psychometrics. *arXiv Preprint arXiv:2603.15909*.

Santoro, A., & Nicosia, V. (2020). Algorithmic complexity of multiplex networks. *Physical Review X*, *10*(2), 021069. https://doi.org/10.1103/PhysRevX.10.021069

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., et al. (2024). *DeepSeekMath: Pushing the limits of mathematical reasoning in open language models*. https://doi.org/10.48550/arXiv.2402.03300

Strehl, A., & Ghosh, J. (2002). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, *3*, 583–617.

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. *Proceedings of the 34th International Conference on Machine Learning*, 3319–3328.

Takens, F. (1981). Detecting strange attractors in turbulence. In *Lecture notes in mathematics* (Vol. 898, pp. 366–381). Springer. https://doi.org/10.1007/BFb0091924

Tenney, I., Das, D., & Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. https://doi.org/10.18653/v1/P19-1452

Tomašević, A., Golino, H., & Christensen, A. P. (2024). *Decoding emotion dynamics in videos using dynamic exploratory graph analysis and zero-shot image classification: A simulation and tutorial using the transforEmotion R package*. https://doi.org/10.31234/osf.io/hf3g7

Vig, J. (2019). A multiscale visualization of attention in the transformer model. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 37–42. https://doi.org/10.18653/v1/P19-3007

Whitney, H. (1936). Differentiable manifolds. *The Annals of Mathematics*, *37*(3), 645–680. https://doi.org/10.2307/1968482

Appendix 1. Proof of the Semantic Manifold Reconstruction Theorem

Lemma 1 (Bi-Lipschitz Property on Compact Sets)**.** *If $\Phi : \mathcal{M} \to \mathbb{R}^{2m+1}$ is a C^1 embedding and $\mathcal{M}$ is compact, then there exist constants $0 < a \leq b < \infty$ such that for all $x, y \in \mathcal{M}$,*

$$a\, d(x, y) \leq \|\Phi(x) - \Phi(y)\| \leq b\, d(x, y),$$

where d is any Riemannian distance on $\mathcal{M}$ equivalent to the ambient metric.

Proof sketch. Compactness of $\mathcal{M}$ together with C^1 regularity ensures that the Jacobian $D\Phi$ has uniformly bounded operator norm on $\mathcal{M}$ (yielding $b < \infty$) and that its smallest singular value is bounded away from zero everywhere on $\mathcal{M}$; combined with global injectivity and continuity of Φ^{-1} on the compact image $\Phi(\mathcal{M})$, this yields $a > 0$. □

Lemma 2 (Separation Preservation)**.** *Let $\{C_1, \ldots, C_r\}$ partition $\mathcal{M}$ into closed subsets with positive separation margin $\Delta = \min_{i \neq j} \inf\{d(x, y) : x \in C_i, y \in C_j\} > 0$. Under the bi-Lipschitz map Φ of Lemma 1,*

$$\min_{i \neq j} \inf\{\|\Phi(x) - \Phi(y)\| : x \in C_i, y \in C_j\} \geq a\,\Delta > 0.$$

Consequence. Any clustering algorithm operating on $\Phi(\mathcal{M})$ that is stable under bi-Lipschitz distortions and consistent for Δ-separated clusters recovers the underlying partition up to label permutation, provided finite-sample estimation noise is sufficiently small.

Appendix 2. Implementation Details

Activation and Network Analysis. Token-level hidden states are mean-pooled within each item. Hidden-unit features are standardized across stacked item-by-state observations prior to analyzing each recorded state separately. Unit-lag ($\emptyset = 1$) time-delay embeddings sweep $n \in \{3, \ldots, 200\}$; first-order derivatives are estimated via GLLA; sparse partial-correlation networks are estimated via EBICglasso (100-value ˘ grid, adaptive fl $\in \{0.50, 0.25, 0\}$); and communities are detected via Louvain. NMI evaluates alignment against the five stipulated attribute labels.

Complexity Encoding. Estimated network matrices are thresholded to binary edge presence. Each retained edge is encoded by its node-coordinate pair and a constant retained-edge value of $2^1 = 2$ (matching unweighted single-layer prime-weight encoding in Golino et al. (2025)), serialized to text, randomly shuffled, and compressed with gzip. KC is the mean compressed byte length across 1,000 random edge-order permutations. This estimator measures topological compressibility and does not encode continuous edge weights.

Aggregation and Reproducibility. All models use arithmetic means across iterations. Empirical TEFI and KC bounds are pooled across every five-item observation prior to computing RM; because RM is affine in its three components, cell-level mean RM is computed directly from cell-level means of NMI, TEFI, and KC. Per-state RM trajectories take the maximum over embedding dimensions, whereas mean-NMI depth curves average over embedding dimensions. Accompanying reproducibility materials archive checkpoint identifiers, software environment records, input checksums, validation logs, cell/layer summary tables, and scripts to rebuild all tables and figures from saved simulation outputs without rerunning LLM inference.

Appendix 3. CET Pipeline Pseudocode

Algorithm 1 Construct Emergence Tracing (CET)

Require: Model $\mathcal{M}$ with L recorded states; item pool $\mathcal{P}$ with r attributes; $k = 5$ items per attribute; N_{iter} iterations; embedding range $[n_{\min}, n_{\max}]$
Ensure: State-by-embedding metrics $\{\text{NMI}, \text{TEFI}, \text{KC}\}_{\ell,n}$ and Representational Maturity RM

for $t = 1$ to N_{iter} **do**
  Sample k items per attribute without replacement from $\mathcal{P} \rightarrow$ item set $\mathcal{I}_t$, ground-truth labels $\mathbf{y}_t$
  for each item $x \in \mathcal{I}_t$ **do**
    Extract mean-pooled hidden states $\mathbf{h}_\ell(x) \in \mathbb{R}^d$ for each recorded state $\ell = 1, \ldots, L$
  end for
  Stack all item-by-state vectors; standardize each hidden unit across the stack
  for $\ell = 1$ to L **do**
    Select standardized rows for state ℓ
    for $n = n_{\min}$ to $n_{\max}$ **do**
      Construct time-delay embedding matrices $\mathbf{X}^{(n)}$ with n dimensions ($\tau = 1$)
      Estimate first-order derivatives via GLLA: $\mathbf{Y} = \mathbf{X}\mathbf{L}(\mathbf{L}'\mathbf{L})^{-1}$
      Estimate sparse Gaussian Graphical Model via EBICglasso
      Detect communities $\hat{\mathbf{c}}$ via Louvain
      Compute NMI($\hat{\mathbf{c}}, \mathbf{y}_t$), TEFI, and KC
    end for
  end for
end for
Aggregate NMI, TEFI, and KC via arithmetic means across iterations
Compute RM using empirical TEFI and KC bounds pooled across all models

Appendix 4. Component-Metric Landscapes

The following figures report the component metrics on their observed scales using the same five-item conditions and arithmetic aggregation as the main results.

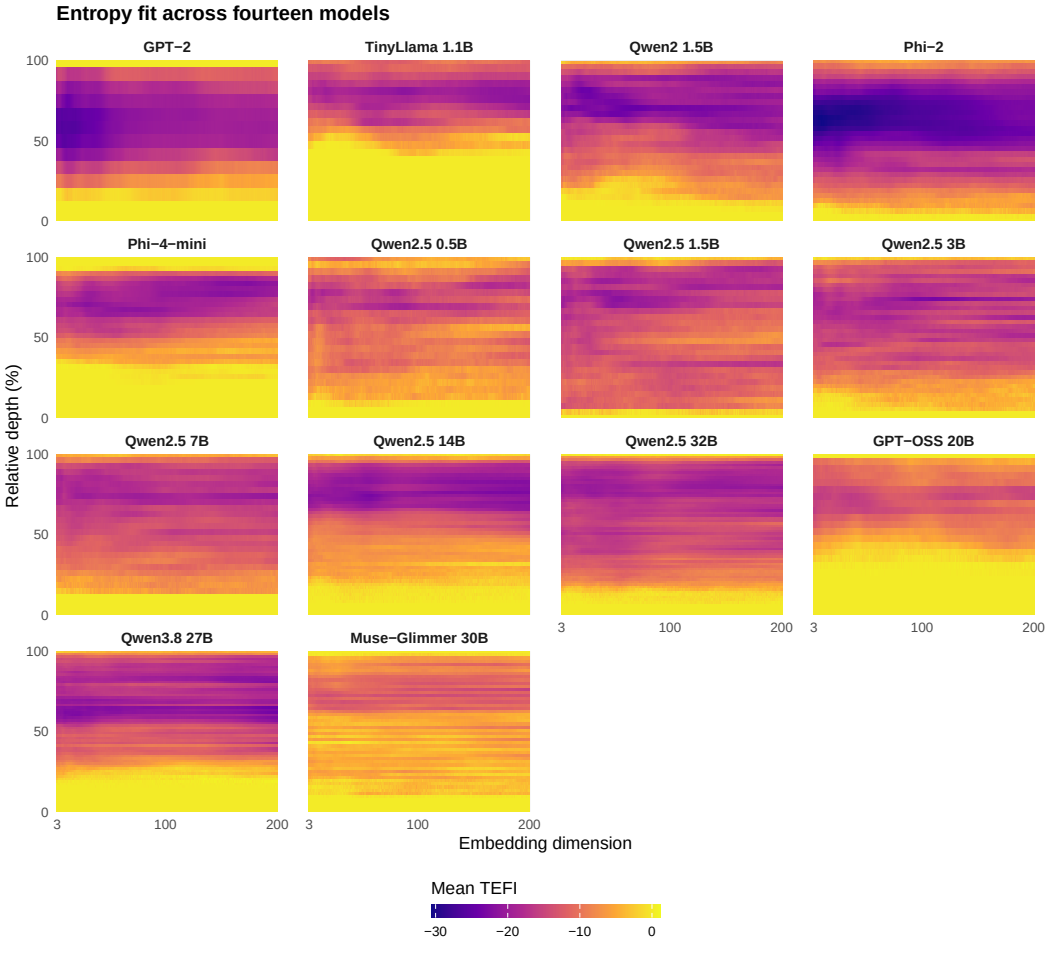

Figure 7. Mean TEFI across fourteen models. All panels use one color scale and the same relative-depth convention. Lower (more negative) values indicate better entropy-based structural fit under the implemented network analysis.

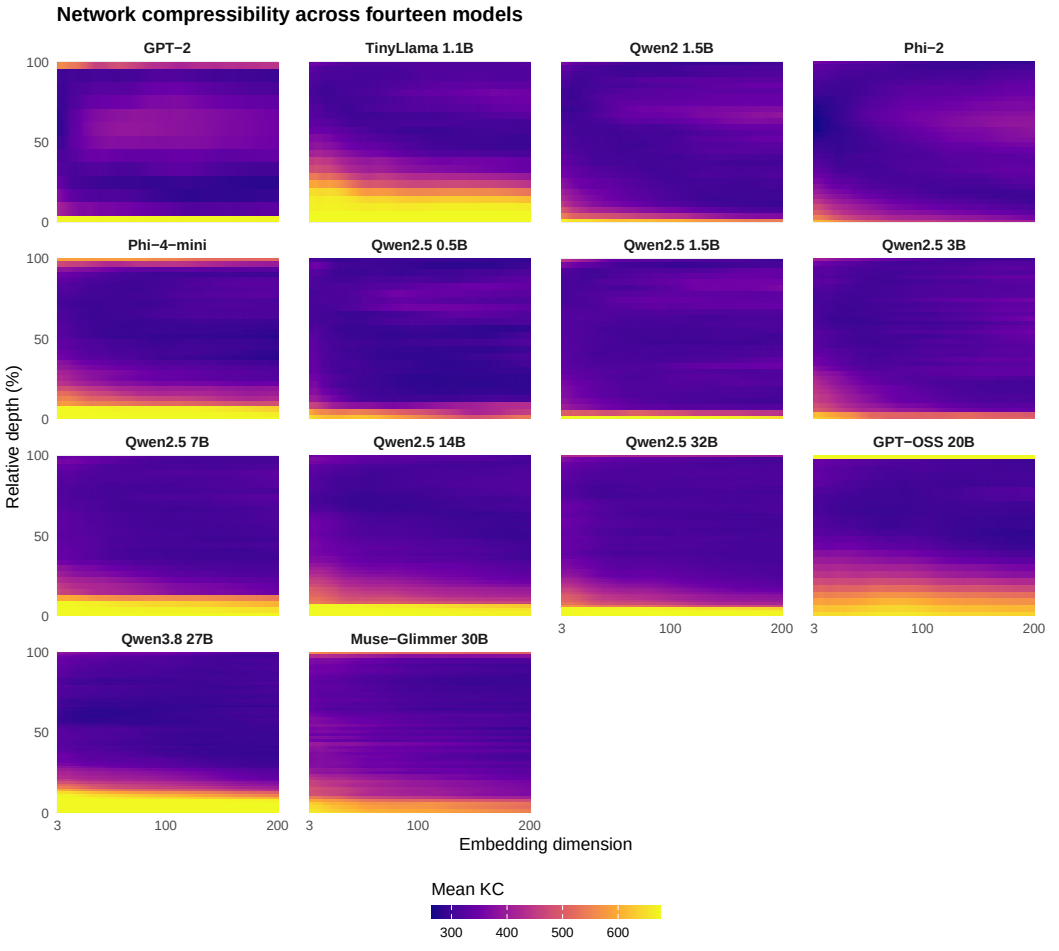

Figure 8. Mean KC across fourteen models. Colors represent gzip-compressed length for the topology encoding. Lower values indicate more compressible network topologies rather than reduced neural storage requirements.

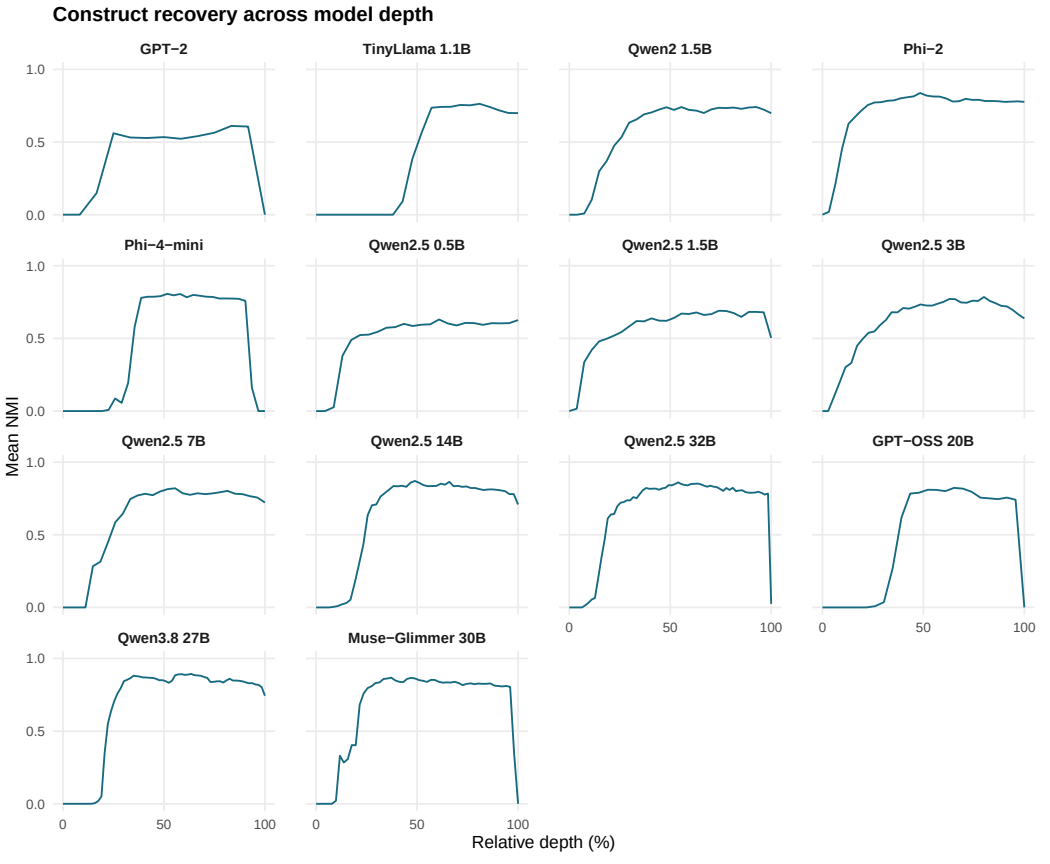

Figure 9. Mean NMI across relative depth for all fourteen models. Curves average over embedding dimensions and iterations, complementing the per-state maxima used for RM trajectories.