reEtym: A Natively Feature-Disentangled Transformer for Interpretability

A Metal Soul In My Hand

Hongyu Shi
Independent Researcher
recontinuac@gmail.com

Abstract

Based on the hypothesis that "human language is composed of fundamental semantic atoms," this paper proposes reEtym, a feature-disentangled architecture that modifies only the embedding layer. By factorizing the embedding matrix into a "recipe" matrix W_{recipe} and an "etymological basis" matrix W_{basis}, the model is guided to maintain a continuous set of semantic etymological bases in the latent space.

At 0.5B parameters and 50k pretraining steps, reEtym achieves near-lossless equivalence with conventional architectures on zero-shot benchmarks (fluctuations within ±2.4%), while improving topic coherence by 28.4% and reducing extreme failure cases by 98.6%. Concurrently, interpretable structures spontaneously emerge in the etymological space: semantic algebra (6/6 hits, including linguistic and arithmetic analogies), natural sparsity (11–13% activation rate), and signal-level causal traceability (ablating a single signal reduces prediction from 8.31% to 0.03%), revealing new avenues for exploration.

Unlike post-hoc reconstruction methods, the etymological space in reEtym is directly defined by the architecture and constitutes a native component of the model's computation. This enables audit findings to be directly translated into model modifications—adjusting recipes or bases can achieve behavioral steering such as sentiment manipulation and topic coherence enhancement, without retraining. Since modifications are confined to the embedding layer, this mechanism naturally extends to non-Transformer architectures such as Mamba and RWKV.

The complete source code, model weights, training logs, and an online interpretability platform are publicly available under the MIT license at: https://github.com/reEtym/reEtym.

The name reEtym derives from etymology, signifying the pursuit of tracing the fundamental building blocks of language—the etymological signal bases.

Terminology This paper adopts the following terms: etymological architecture (reEtym), etymological signal (Signal), signal basis (Signal Basis, i.e., W_{basis}), etymological recipe (Recipe, i.e., W_{recipe}), signal space (Signal Space). Throughout the text, abbreviated references to "signal," "basis," and "recipe" refer to the above concepts.

1 Introduction

In the standard Transformer paradigm, vocabulary representations are independently parameterized by an unconstrained embedding matrix $E \in \mathbb{R}^{V \times d}$. This high-dimensional discrete mapping lacks explicit structural constraints, causing semantic concepts to become entangled in the hidden space.

Existing interpretability methods (e.g., Sparse Autoencoders, SAE Bricken et al. (2023); Templeton et al. (2024)) extract semantic features through post-hoc approximation after

training is complete. Their analytical findings cannot be directly written back into the model, revealing a fundamental gap between interpretability and modifiability.

The core design philosophy of reEtym: Tokens are merely the surface manifestation of human language; the internal information evolution within a model should be modeled as the flow of continuous "etymological bases." By imposing factorization constraints at the architectural level, the etymological space becomes a native component of the model's computation rather than a post-hoc reconstructed approximation, thereby naturally unifying interpretability analysis with model modification.

1.1 Architecture Overview

The reEtymSignalEmbedding module decomposes the conventional mapping matrix into two sub-matrices with explicit semantic roles:

$$W_{recipe} \in \mathbb{R}^{V \times S}, \quad W_{basis} \in \mathbb{R}^{S \times d} \tag{1}$$

where S is a hyperparameter representing the total number of etymological signals maintained by the system. For any input token i, its initial hidden state is dynamically generated by the recipe:

$$e_i = \sum_{k=1}^{S} W_{recipe}^{(i,k)} \cdot W_{basis}^{(k,:)} \tag{2}$$

W_{basis} constitutes a globally shared signal library, and the i-th row of W_{recipe} defines the recipe for that token. This design guides the model to learn compositional rules from fundamental semantic atoms to complex concepts. All current experiments adopt $S = d$ (equal-rank setting); this decomposition is not a rank-reduction operation but a structured factorization.

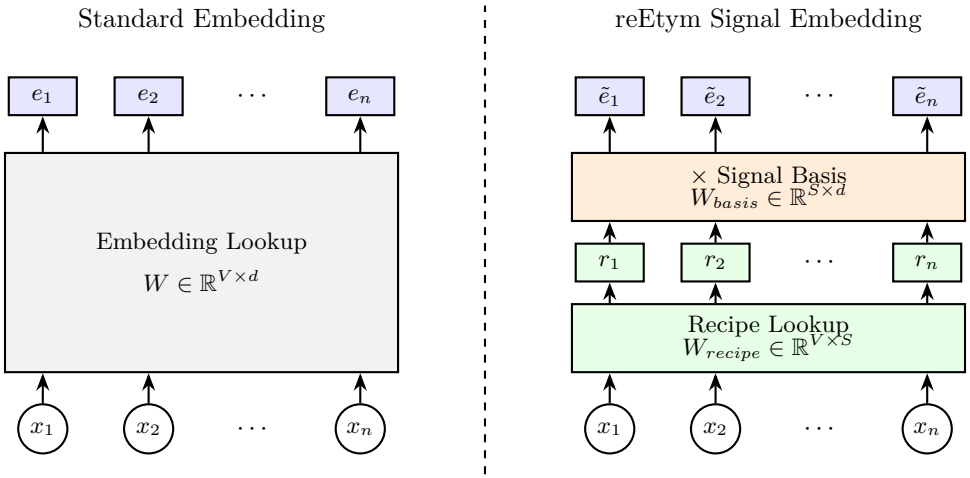

Figure 1: Architecture comparison: Standard embedding (left) performs direct table lookup; reEtym (right) generates embeddings via recipe × basis factorization.

During each forward pass, the dynamic vocabulary matrix $W_{vocab} = W_{recipe} \times W_{basis}$ is reconstructed in real time for output projection, achieving deep parameter sharing between input and output and forming a closed logical loop of "surface input → signal flow → surface output." This architecture also provides a natural observation window for tracking the process by which model predictions progressively consolidate across layers (referred to in this paper as information crystallization; see Section 3.4).

2 Experimental Validation

2.1 Training Setup

Considering effective allocation of computational resources and the depth of mechanistic analysis, this paper follows the common paradigm in mechanistic interpretability research Bricken et al. (2023); Templeton et al. (2024) and selects a 0.5B-scale model as the primary experimental subject, trained on OpenWebText Gokaslan & Cohen (2019) using 4× Tesla T4 GPUs for 50k steps (data preprocessing details in Appendix B). Comparison models include: GPT-2 Radford et al. (2019) (505.62M), GPT-2-New (514.01M, + RoPE Su et al. (2021)/SwiGLU Shazeer (2020)/RMSNorm Zhang & Sennrich (2019)), reEtym-1 (463.67M), reEtym-1-Big (515.06M), reEtym-1-Small (46.47M), and reEtym-1-Lite (413.34M, GQA with 4 heads).

Architectural relationship: All reEtym variants are based on GPT-2-New, with only the embedding layer and output projection replaced by the factorized form ($W_{recipe} \times W_{basis}$); all other structures remain unchanged. The additional 1.05M parameters in reEtym-1-Big compared to GPT-2-New originate from the $W_{basis} \in \mathbb{R}^{S \times d}$ matrix introduced by the factorization. Detailed hyperparameters are provided in Appendix A.

2.2 Convergence and Zero-Shot Performance

Under matched depth and scale, the validation loss gap between reEtym-1-Big and GPT-2-New is approximately 1%. As shown in Figure 2, the validation losses of three reEtym variants at different scales strictly follow scaling laws Kaplan et al. (2020).

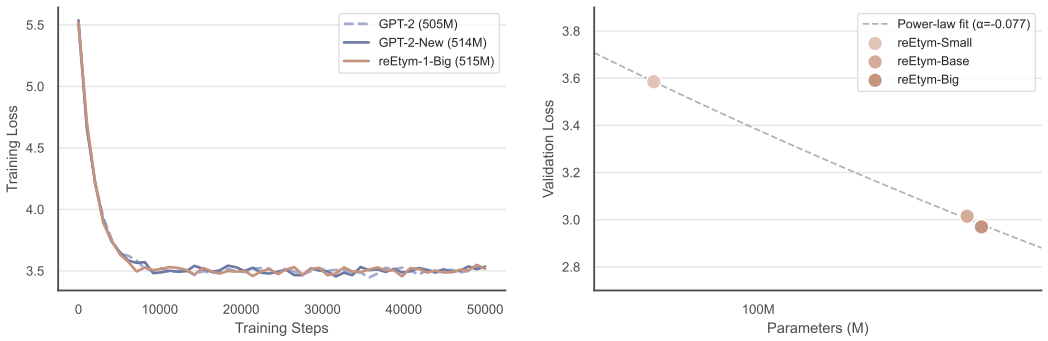

Figure 2: Model training and scaling analysis. Left: Training convergence comparison showing loss curves for the GPT-2 family and reEtym-1-Big. Right: Scaling law verification showing that the reEtym architecture follows power-law decay across different parameter scales, indicating that the factorization constraint preserves the model's scaling properties.

Topic Coherence Evaluation To validate the impact of etymological decomposition on long-text generation quality, we conducted 24,000 generations across 120 prompts, measuring coherence through three weighted metrics: embedding drift, keyword retention, and perplexity standard deviation (details in Appendix C).

Table 1 presents the zero-shot benchmark comparison: performance is nearly equivalent (fluctuations within ±2.4%, $|d| < 0.05$), topic coherence improves by 28.38%, and low-quality samples decrease by 98.6%.

Mechanistic Explanation The performance improvement stems from reEtym's structured signal representation. The conventional embedding matrix $\mathbf{W}_e \in \mathbb{R}^{V \times d}$ assigns an independent vector to each token without explicit semantic constraints. In contrast, reEtym's recipe × basis decomposition $\mathbf{W}_e = \mathbf{RB}$ ($\mathbf{R} \in \mathbb{R}^{V \times S}$, $\mathbf{B} \in \mathbb{R}^{S \times d}$) constrains the representation space to linear combinations of S-dimensional etymological bases. This structural constraint confines the generation process within a predefined etymological subspace, suppressing semantic drift by restricting representational degrees of freedom, thereby improving topic coherence in generated text.

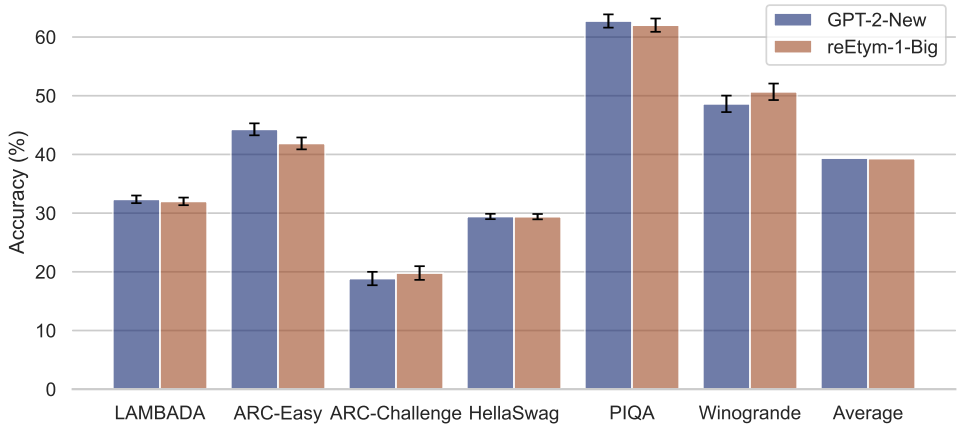

Figure 3: Zero-shot benchmark accuracy comparison (error bars indicate 95% confidence intervals).

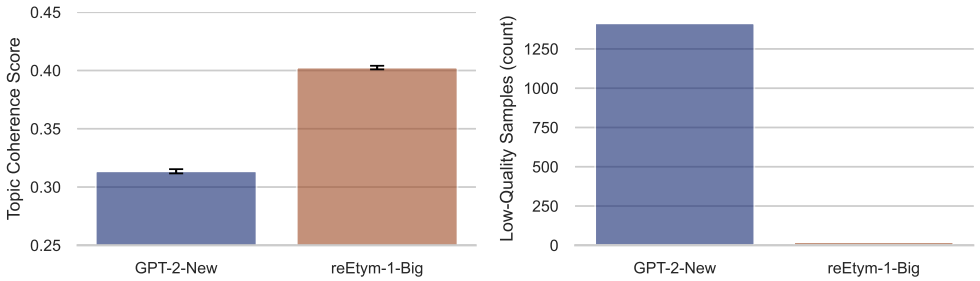

Figure 4: Generation quality comparison: left panel shows topic coherence (error bars indicate 95% CI), right panel shows low-quality sample counts.

Table 1: Zero-shot benchmark performance comparison

Benchmark	GPT-2-New	reEtym-1-Big	Δ	$\lvert d \rvert$
OWT Val. Loss ↓	2.9098	2.9363	+0.91%	–
LAMBADA Acc. ↑	32.35%	32.00%	-0.35%	0.008
ARC-Easy ↑	44.28%	41.88%	-2.40%	0.024
ARC-Challenge ↑	18.86%	19.80%	+0.94%	0.023
HellaSwag ↑	29.43%	29.40%	-0.03%	0.001
PIQA ↑	62.73%	62.02%	-0.71%	0.015
Winogrande ↑	48.62%	50.67%	+2.05%	0.041
Topic Coherence ↑	0.3135	0.4025	+28.38%	–
Low-Quality Samples ↓	1,413	20	-98.6%	–
Distinct-1 ↑	0.0662	0.0674	+1.8%	–
Distinct-2 ↑	0.3826	0.3853	+0.7%	–

Note: $\lvert d \rvert$ denotes Cohen's d effect size. All zero-shot benchmarks have $\lvert d \rvert < 0.05$, well below the negligible difference threshold (0.2), indicating near-equivalent performance.

3 Interpretability Experiments

The preceding experiments demonstrate that reEtym achieves near-equivalent performance with conventional architectures while significantly improving generation quality. To systematically validate the interpretability of reEtym, we designed 12 experiments (details in Appendix D), organized into four groups:

Signal Ontology (Experiments 1–3): Recipe space atlas, signal sparsity analysis, and signal basis geometry, validating the intrinsic organization of the etymological space.

Semantic Properties (Experiments 4–6): Semantic constellation map (PCA/t-SNE), semantic algebraic operations, and spelling robustness, testing whether semantic structure emerges in the recipe space.

Mechanistic Analysis (Experiments 7–9): Layer-wise probability evolution, signal flow tracing, and causal ablation curves, revealing the causal role of signals during inference.

Intervention Validation (Experiments 10–12): Task crystallization boundary shift, concept injection, and gene pool hijacking, validating the controllability of etymological space interventions and the task-dependence of crystallization boundaries.

This section highlights the core findings; complete experimental details are provided in the appendix.

3.1 Semantic Structure and Robustness of the Recipe Space

If the factorization in reEtym were merely a semantically unconstrained numerical reparameterization, recipe vectors should exhibit no meaningful geometric structure. Experimental results reject this null hypothesis.

Recipe Nearest-Neighbor Structure Computing cosine similarity of recipe vectors across the entire vocabulary, all Top-20 nearest-neighbor pairs are semantically plausible associations. Table 2 lists representative results.

Semantic Clustering Visualization Experiment 4 applies PCA/t-SNE dimensionality reduction to visualize recipe vectors of 60 probe words, revealing an initial separation trend among semantic categories such as countries, animals, colors, and emotions in the recipe space, with a silhouette score of 0.1052. Notably, this clustering structure emerges spontaneously after only 50k training steps, indicating that the factorization architecture imposes an effective semantic inductive bias, guiding the etymological space to form meaningful organizational structure early in training (details in Appendix D).

Table 2: Representative nearest-neighbor pairs in the recipe space and vocabulary-wide neighbor examples

Rank	Word Pair	Cosine Sim.	Probe Word	Top-4 Vocabulary Neighbors
1	three ↔ four	0.7551	China	Chinese, Beijing, Japan, Russia
2	four ↔ five	0.7201	France	Spain, Germany, Italy, French
3	two ↔ three	0.6723	Russia	Russian, Russians, Moscow, Russian†
5	boy ↔ girl	0.5792	India	Indian, Australia, Canada, Pakistan
8	king ↔ queen	0.5428	dog	dogs, Dog, Dogs, canine
9	France ↔ Germany	0.5318	cat	cats, Cat, Cat†, dog
10	China ↔ Beijing	0.5084	red	blue, Red, yellow, green
13	black ↔ white	0.4881	Germany	Italy, France, Germans, German

†Homographic tokens with/without a leading space in the BPE tokenizer are distinct tokens; after stripping spaces, they appear as duplicates.

Semantic Algebra Table 3 presents the semantic algebra test results: 3/3 hits for linguistic analogies and 3/3 hits for arithmetic analogies, with all expected values within the Top-5. Unlike conventional word vector algebra, reEtym performs operations in the etymological

recipe space: combining and subtracting individual etymological signals as basic units, then retrieving the nearest-neighbor token. The successful hits on arithmetic analogies further indicate that ordinal numerical relationships spontaneously emerge in the etymological space (details in Appendix D).

Spelling Robustness Experiment 6 shows that the deep etymological representations of misspelled sentences have higher cosine similarity to those of correctly spelled sentences than to semantically unrelated sentences, though the margin is small and may be limited by the representational granularity of the BPE tokenizer (details in Appendix D).

Table 3: Semantic algebra operation results

Expression	Expected	Rank	Top-5
king + woman − man	queen	#1	queen, fml, independ, Dise, iosyncr
walked + running − walking	ran	#1	ran, Running, runs, Running, running
Paris + China − France	Beijing	#2	Chinese, Beijing, China, Asia, Shanghai
3 + 4 − 2	5	#1	5, 6, 7, 8, 9
4 + 7 − 9	2	#4	5, 6, 3, 2, 8
3 + 9 − 4	8	#2	7, 8, 2, 6, 5

3.2 Natural Sparsity and Signal Utilization

Natural Sparsity Using mean + 1σ as the activation threshold, each token in reEtym-1-Big activates only 116.6/1024 signals. This sparsity emerges naturally without any external constraint. Figure 5 compares four reEtym variants: despite parameter scales ranging from 46.47M to 515.06M and signal dimensions from 512 to 1024, the activation rates of all variants cluster within the 11–13% range with approximately normal distributions. This indicates that natural sparsity is an intrinsic property of the factorized architecture, rather than an incidental consequence of specific model configurations or training hyperparameters.

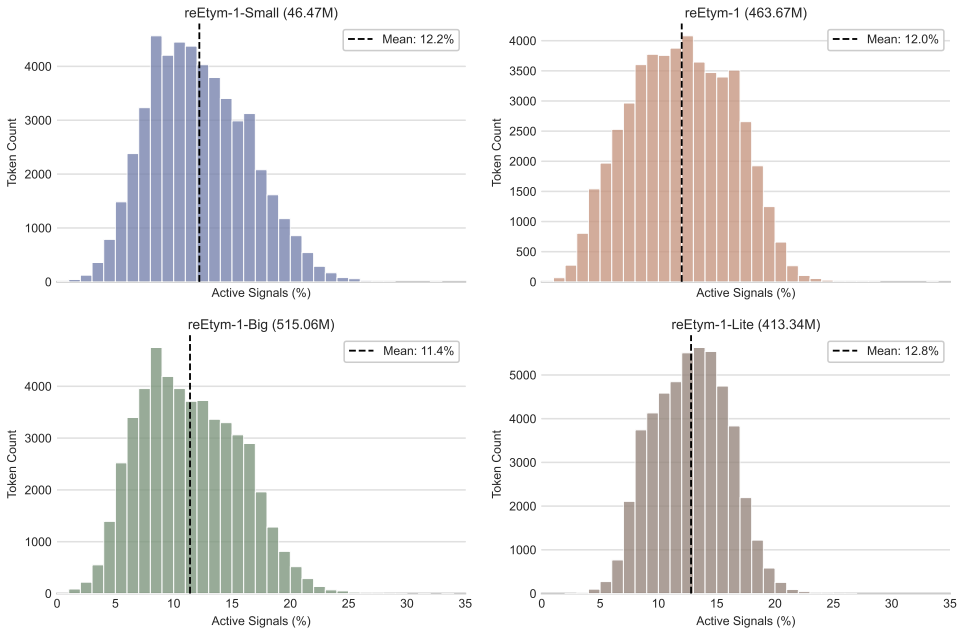

Figure 5: Distribution of active signal proportions per token across four reEtym variants. Dashed lines mark the mean activation rate for each variant.

Signal Utilization Balance Experiment 3 shows that the effective rank of the signal basis matrix is 856.7/1024 (83.7%), with uniform utilization across signal dimensions and no degeneracy observed. The signal variance analysis in Experiment 1 further confirms a Gini

coefficient of 0.085, indicating that each signal's discriminative contribution to the vocabulary is highly balanced.

3.3 Causal Traceability

Layer-Wise Probability Evolution Experiment 7 tracks the layer-by-layer evolution of target word probability across 36 layers, with Shannon entropy monotonically decreasing and predictions beginning to converge at middle layers (details in Appendix D).

Signal Flow Tracing Experiment 8 projects hidden states at each layer into the etymological space and finds that a small number of high-variance signals remain persistently active across layers, forming the backbone channels for semantic transmission (details in Appendix D).

Concentration of Causal Ablation For "The capital of France is" (baseline prediction "the," 8.31%), we perform per-signal ablation: ablating only 1 key signal (#89, corresponding to the function word channel {the, a, in, to, an, at}) reduces the probability to 0.03%, shifting the prediction to "famous."

Figure 6 shows ablation curves for three groups, demonstrating that causal importance is concentrated in a few key signals, and that reEtym's inference process possesses signal-level causal traceability.

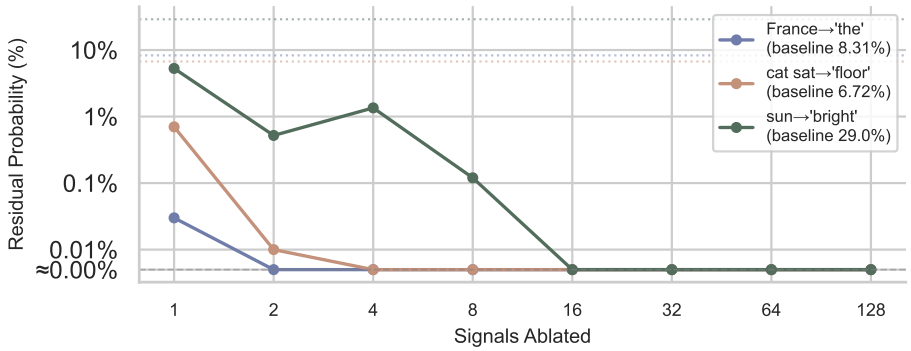

Figure 6: Causal ablation curves (log-log scale): decay trajectories of residual target word probability after ablating key signals.

3.4 Controllability and Information Crystallization

Signal-level causal traceability provides the experimental foundation for precise behavioral control. If ablating key signals can significantly alter predictions, then the reverse operation—amplifying or injecting specific signals—should enable controllable behavioral steering. Experiments 10–12 validate this hypothesis.

Task-Dependence of Information Crystallization Boundaries Experiment 10 progressively delays the injection starting point layer by layer at a fixed strength of $1.2\alpha_{min}$, recording the first failure layer as the crystallization boundary. Results: short context L18.5, long context L26.2, code tasks L7.8, indicating that the crystallization boundary is a function of task complexity (details in Appendix D).

Concept Injection The core question of Experiment 11 is: at a fixed injection position, what intensity is required to flip the prediction? Through binary search, the critical intervention intensity α_c is determined: flipping the prediction of "The capital of France is" from "the" to "London" ($\alpha_c = 18.6$), "The cat sat on the" from "floor" to "moon" ($\alpha_c = 19.2$), and "The sun is very" from "bright" to "cold" ($\alpha_c = 17.4$). The critical values concentrate in the 17–19 range (mean 18.4), exhibiting a predictable dose-response relationship.

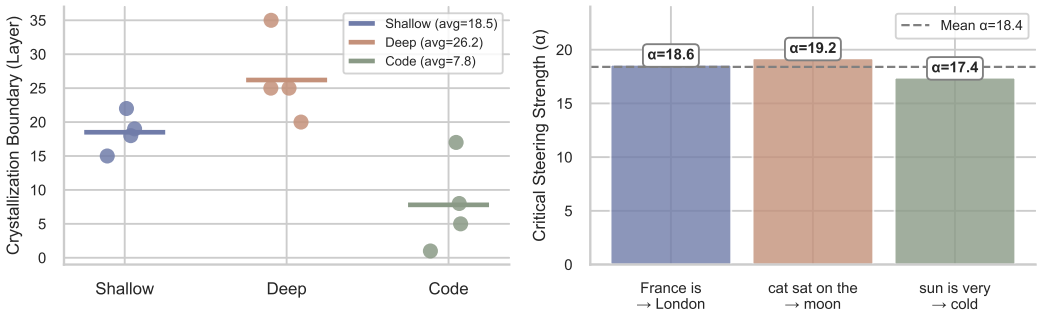

Figure 7: Left: Distribution of task crystallization boundaries (Experiment 10), with scatter points representing per-sample boundary layers and horizontal lines indicating group means. Right: Critical intensity for concept injection (Experiment 11).

Gene Pool Hijacking This experiment directly modifies the recipe matrix W_{recipe} across the entire vocabulary by superimposing a sentiment-reversal vector ($\mathbf{v}_{pos} - \mathbf{v}_{neg}$) at intensity $\alpha = 1.5$. Table 4 presents the comparison results for 5 prompts: the control group produces semantically uncontrolled continuations, while the intervention group consistently steers toward positive semantics with preserved grammatical coherence.

Table 4: Gene pool hijacking experiment: comparison between control and intervention groups ($\alpha = 1.5$)

Prompt	Generated Continuation
The food was disgusting.	Control: …I was so sick. I was so sick. Intervention: …I had a lot of fun time with my friends.
I failed the exam again.	Control: …I was told I had to go to the police station. Intervention: …I was so happy. I was so happy.
The weather is terrible today.	Control: …The wind is blowing. The sun is shining. Intervention: …The clouds are beautiful. The sun are bea…
The project was a complete failure.	Control: …The project was a complete failure. Intervention: …The project was a complete success.
I hate waiting in long lines.	Control: …The restaurant's owner, who declined to be… Intervention: …I love the way the ball is played.

4 Theoretical Foundations of Etymological Space Intervention

The fundamental distinction between reEtym and post-hoc analysis methods such as SAE lies in that: the etymological space in reEtym is a native component of the model's computation, rather than an approximate representation reconstructed after training. This design yields three advantages: (1) the etymological space is directly defined by the architecture, requiring no additional training; (2) signal activations directly participate in the forward computation, providing causal interpretability; (3) modifications to W_{recipe} or W_{basis} act directly on the model itself, without retraining.

4.1 Factorized Computation Flow and Intervention Point Localization

reEtym reformulates the forward computation as a three-stage factorized form:

$$\begin{aligned} \text{Stage I (Recipe Retrieval):} \quad & \mathbf{r}_i = W_{recipe}[i,:] \in \mathbb{R}^S \\ \text{Stage II (Signal Projection):} \quad & \mathbf{h}_0 = \mathbf{r}_i \cdot W_{basis} \in \mathbb{R}^d \\ \text{Stage III (Layer-wise Transformation):} \quad & \mathbf{h}_{\ell+1} = \mathcal{F}_\ell(\mathbf{h}_\ell; \Theta_\ell) \end{aligned} \tag{3}$$

Core Proposition: Each layer $\mathcal{F}_\ell$ learns general processing logic over signal vectors, rather than hard-coded mappings for specific tokens. Consequently, modifying $\mathbf{r}_i$ changes only that

token's initial representation: processing capacity is conserved while semantic tendency is altered (detailed proof in Appendix E).

4.2 Intervention Operators and Scope Analysis

Based on the above theoretical framework, the etymological space supports three classes of precise interventions (complete derivations in Appendix E):

Recipe-level intervention (token-local): Modifying $W_{recipe}[i, :]$ confines the scope strictly to the initial embedding of token i; all other tokens and all layer parameters remain unchanged.

Basis-level intervention (globally synchronized): Modifying $W_{basis}[k, :]$ simultaneously affects all tokens in the vocabulary that activate signal k, enabling one-shot editing of entire semantic lexical clusters.

Inference-time injection (zero-parameter): Superimposing $\mathbf{h}_\ell \leftarrow \mathbf{h}_\ell + \alpha \cdot \mathbf{b}_k$ at any layer ℓ, without modifying model parameters. Experiment 11 validates its dose-response relationship (mean critical $\alpha = 18.4$).

5 Related Work

Language model foundations: reEtym uses GPT-2 Radford et al. (2019) as its base architecture, with training code implemented on nanoGPT Karpathy (2022) and training data from OpenWebText Gokaslan & Cohen (2019).

Embedding factorization: ALBERT Lan et al. (2020) uses $E \ll H$ for parameter efficiency; reEtym uses $S = d$ for semantic structure, where the factors carry explicit semantic roles and the same factorized product serves as both the input embedding and the output projection.

Sparse Autoencoders (SAE) Bricken et al. (2023); Templeton et al. (2024): SAEs train post-hoc decoders to reconstruct activations, potentially suffering from significant reconstruction loss and dead feature problems Bricken et al. (2023); reEtym's etymological space is directly defined by the architecture, with a Gini coefficient of 0.085 and all signals being utilized. The key distinction: SAE findings cannot be directly written back into the model; reEtym audit findings can directly modify the model without retraining.

6 Conclusion

reEtym, through factorizing the embedding matrix into $W_{recipe} \times W_{basis}$, enables the model to spontaneously develop internal representations that are semantically organized, causally traceable, and behaviorally controllable, while preserving language modeling capability. Core findings: validation loss gap of approximately 1%; semantic algebra 6/6 hits; natural sparsity 11–13%; topic coherence +28.4%, extreme failure cases reduced by 98.6%.

Since reEtym's modifications are strictly confined to the embedding layer, its core mechanism is decoupled from model depth and width, providing the structural prerequisites for scaling to larger models and enabling a broad range of applications.

References

Trenton Bricken, Adly Templeton, Joshua Batson, et al. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits Thread, 2023.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. 2019. URL http://Skylion007.github.io/OpenWebTextCorpus.

Jared Kaplan, Sam McCandlish, Tom Henighan, et al. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361, 2020.

Andrej Karpathy. nanogpt, 2022. URL https://github.com/karpathy/nanoGPT.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In International Conference on Learning Representations, 2020.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9, 2019.

Noam Shazeer. GLU variants improve transformer. arXiv preprint arXiv:2002.05202, 2020.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. arXiv preprint arXiv:2104.09864, 2021.

Adly Templeton, Tom Conerly, Jonathan Marcus, et al. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. Transformer Circuits Thread, 2024.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In Advances in Neural Information Processing Systems, volume 32, 2019. arXiv:1910.07467.

A Detailed Hyperparameter Configuration

All models use the AdamW optimizer ($\beta_1 = 0.9, \beta_2 = 0.95$), weight decay 0.1, gradient clipping 1.0, trained for 50k steps on 4× Tesla T4 GPUs. The learning rate schedules differ across models due to architectural differences; see Table 5 and Table 6 for details.

Table 5: Model architecture hyperparameters

Model	Layers	d_{model}	Heads	d_{ff}	S	Params
GPT-2	36	1024	16	4096	-	505.62M
GPT-2-New	36	1024	16	2816	-	514.01M
reEtym-1-Small	6	512	8	1536	512	46.47M
reEtym-1-Lite	32	1024	16/4	2816	1024	413.34M
reEtym-1	32	1024	16	2816	1024	463.67M
reEtym-1-Big	36	1024	16	2816	1024	515.06M

Note: reEtym-1-Lite heads 16/4 indicates 16 query heads and 4 KV heads (GQA). The SwiGLU expansion factor for reEtym-1-Lite is 2.66 (intermediate value 2723, rounded up to the nearest multiple of 256, yielding 2816).

Table 6: Training hyperparameters

Model	LR	Min LR	Cosine Decay	Warmup	Batch Size
GPT-2	3e-4	3e-5	✓	1000	262k
GPT-2-New	3e-4	3e-5	✓	1000	262k
reEtym-1-Small	1.5e-4	6e-5	–	2000	262k
reEtym-1-Lite	1e-4	1e-5	✓	2000	164k
reEtym-1	1.5e-4	6e-5	–	2000	262k
reEtym-1-Big	3e-4	3e-5	✓	1000	262k

Note: Batch size is measured in tokens (= batch_size × block_size × gradient_accumulation_steps × world_size).

A.1 Initialization and Variance Control

In deep neural networks, controlling activation variance during forward propagation is a necessary condition for stable gradient propagation. Since the embedding layer in reEtym involves a matrix multiplication, standard Gaussian initialization would cause variance drift.

Assume that the elements of W_{recipe} and W_{basis} follow independent and identically distributed zero-mean Gaussian distributions $R_{i,k} \sim \mathcal{N}(0, \sigma^2)$ and $B_{k,j} \sim \mathcal{N}(0, \sigma^2)$, respectively. For the resulting word embedding vector element $E_{i,j} = \sum_{k=1}^{S} R_{i,k} B_{k,j}$, by the variance property of products of independent random variables:

$$\mathrm{Var}(R_{i,k} B_{k,j}) = \sigma^4 \tag{4}$$

Summing over S independent signal dimensions:

$$\mathrm{Var}(E_{i,j}) = S \cdot \sigma^4 \tag{5}$$

Introducing a scaling control parameter V_s, the initialization standard deviation is defined as $\sigma = \sqrt{V_s/\sqrt{S}}$. Then $\sigma^2 = V_s/\sqrt{S}$, $\sigma^4 = V_s^2/S$, and substituting yields:

$$\mathrm{Var}(E_{i,j}) = S \cdot \sigma^4 = S \cdot \frac{V_s^2}{S} = V_s^2 \tag{6}$$

With $V_s = 0.02$, the final embedding variance is 4×10^{-4}, consistent with Xavier/He small initialization strategies, ensuring training stability.

B Data Preprocessing

We use the OpenWebText dataset Gokaslan & Cohen (2019) with the GPT-2 BPE tokenizer (original vocabulary size 50,257; padded to 50,304 on the model side for alignment to

multiples of 64, improving matrix operation efficiency). The data processing pipeline: (1) download raw text; (2) perform BPE encoding using tiktoken; (3) concatenate into continuous sequences and partition into fixed-length blocks (1024 tokens); (4) randomly shuffle and split into training/validation sets (99%/1%). No content filtering or deduplication was performed.

C Topic Coherence Evaluation Methodology

C.1 Experimental Design

We constructed 120 prompts spanning 6 major categories with 20 prompts each, covering science & technology, emotion & personal, narrative & story, philosophy & abstraction, business & society, and nature & environment. Each prompt was sampled 100 times at temperature 0.8 with Top-K=200, yielding 24,000 total generations across the two models, each generating a fixed length of 80 tokens.

C.2 Scoring Metrics

The topic coherence score is a weighted combination of three sub-metrics:

Embedding Drift (40%) Measures the semantic distance between the prompt and successive segments of the generated text. A sliding window (window size 20 words, stride 10 words) extracts segments from the generated text, and the cosine distance between each segment embedding and the prompt embedding is computed. The final score is $1 - \text{mean\_drift}$. Lower drift indicates that the generated content remains semantically closer to the prompt topic.

Keyword Retention (30%) Calculates the retention rate of keywords (words with length > 3) from the prompt in the generated text. The metric is defined as:

$$\text{Score} = \frac{|\text{prompt\_keywords} \cap \text{generated\_words}|}{|\text{prompt\_keywords}|}$$

This metric reflects whether the model continues to attend to the core concepts in the prompt.

Perplexity Standard Deviation (30%) Computes the standard deviation of per-token perplexity during the generation process. A lower standard deviation indicates better fluency and stability. The score is computed via an inverse transformation to convert it into a positive metric:

$$\text{Score} = \frac{1}{1 + \text{std}(\text{perplexities})}$$

The final coherence score is the weighted sum of the three sub-metrics:

$$\text{coherence\_score} = 0.4 \times (1 - \text{drift}) + 0.3 \times \text{retention} + 0.3 \times \frac{1}{1 + \text{ppl\_std}} \tag{7}$$

C.3 Diversity Verification

To rule out the possibility that the model achieves coherence by converging to high-frequency safe phrases, we simultaneously compute Distinct-N metrics:

$$\text{Distinct-}N = \frac{|\text{unique } N\text{-grams}|}{|\text{total } N\text{-grams}|} \tag{8}$$

Distinct-1 measures lexical diversity, and Distinct-2 measures phrasal diversity.

D Detailed Interpretability Experiments

The 12 experiments are organized into four groups. Below we provide detailed parameter configurations and measured results for each experiment:

D.1 Signal Ontology (Experiments 1–3)

Experiment 1: Recipe Space Atlas

- Probe vocabulary: 60 high-frequency words spanning countries, animals, numbers, colors, emotions, verbs, and adjectives
- Similarity computation: Cosine similarity based on L2-normalized recipe vectors
- Clustering method: Hierarchical clustering (Ward linkage), distance matrix $D = 1 - \text{cosine\_sim}$
- Visualization: Heatmap at 200 DPI, RdBu_r colormap (red-blue divergent)

Table 7 lists the complete Top-20 nearest-neighbor pairs. Semantic associations cover numerical sequences (#1–4, 7, 16, 19), gender (#5–6, 20), geography (#9–12, 14, 17–18), colors (#13, 15), and other categories, with no noise or uninterpretable pairings.

Table 7: Recipe space Top-20 nearest-neighbor pairs (complete)

Rank	Word Pair	Cosine	Rank	Word Pair	Cosine
1	three ↔ four	0.7551	11	Tokyo ↔ Beijing	0.5028
2	four ↔ five	0.7201	12	Japan ↔ Tokyo	0.4897
3	two ↔ three	0.6723	13	black ↔ white	0.4881
4	three ↔ five	0.6684	14	China ↔ Japan	0.4587
5	boy ↔ girl	0.5792	15	blue ↔ yellow	0.4571
6	woman ↔ girl	0.5631	16	five ↔ ten	0.4546
7	two ↔ four	0.5547	17	Japan ↔ Germany	0.4515
8	king ↔ queen	0.5428	18	Tokyo ↔ Berlin	0.4416
9	France ↔ Germany	0.5318	19	two ↔ five	0.4402
10	China ↔ Beijing	0.5084	20	man ↔ woman	0.4398

Signal Variance Analysis Variance statistics are computed for each column of W_{recipe} (i.e., each signal dimension) across the entire vocabulary. The 10 signals with the highest variance (most discriminative): #898, #822, #277, #774, #201, #348, #101, #424, #375, #932; the 10 signals with the lowest variance (approximately constant): #534, #324, #801, #340, #737, #29, #993, #258, #61, #780. The variance Gini coefficient is 0.085, indicating that each signal's discriminative contribution to the vocabulary is highly balanced, with no degeneracy where a few signals dominate.

Experiment 2: Signal Sparsity Analysis

- Activation threshold: $\tau = \mu + \sigma$, where μ is the mean absolute value of the recipe matrix and σ is the standard deviation
- Statistical sample: Recipe vectors of all 50,257 tokens in the vocabulary
- reEtym-1-Big measured values: $\mu = 0.0152$, $\sigma = 0.0145$, $\tau = 0.0297$
- Activation rate: Average of 116.6/1024 signals per token (11.38%), standard deviation 43.4

Experiment 3: Signal Basis Geometry

- SVD decomposition: Full SVD, retaining all 1024 singular values
- Effective rank computation: Shannon entropy formula $\exp(-\sum p_i \log p_i)$, where $p_i = s_i / \sum s_j$
- Control experiment: Standard Gaussian random matrix of the same dimensions ($\mathcal{N}(0, 1)$), single-sample comparison

- Measured effective rank: 856.7/1024 (83.7%); Gini coefficient of intra-basis-vector variance: 0.3092

Note: The Gini coefficient of 0.085 cited in the main text under "Signal Utilization Balance" comes from Experiment 1, measuring the uniformity of each signal's discriminative contribution to the vocabulary (based on the variance distribution of signals across the vocabulary). The Gini coefficient of 0.3092 from Experiment 3 measures the concentration of intra-dimensional variance within each basis vector; the two have different meanings.

D.2 Semantic Properties (Experiments 4–6)

Experiment 4: Semantic Constellation Map

- PCA parameters: 2 principal components, explained variance ratio 21.2%
- t-SNE parameters: perplexity=10, max_iter=1000, learning_rate=200, random_state=42
- Clustering evaluation: Silhouette score = 0.1052, computed in the original high-dimensional recipe space
- Note: 50k steps constitutes an early training stage; the silhouette score reflects that semantic clustering structure has already begun to emerge, and is expected to strengthen further with continued training
- Visualization: adjustText library for automatic label placement to avoid overlaps

Experiment 5: Semantic Algebra

- Test cases: 6 analogy operations, including 3 linguistic analogies (gender, tense, geography) and 3 arithmetic analogies (integer addition/subtraction)
- Retrieval scope: First 50,257 valid vocabulary tokens
- Exclusion mechanism: Token IDs of input operands (e.g., king, woman, man) are set to -1 in the similarity vector to avoid trivial solutions
- Success criterion: Expected word appears in the Top-5 predictions

Unlike conventional word vector algebra, reEtym's operations are performed in the etymological recipe space: the model decomposes tokens into etymological recipe vectors, combines and subtracts individual etymological signals as basic units to synthesize a virtual recipe, then retrieves the cosine nearest neighbor. The successful hits on arithmetic analogies indicate that ordinal numerical relationships spontaneously emerge in the etymological space—the recipe vectors of number tokens form an order-preserving linear structure that makes addition and subtraction approximately valid. This property is not explicitly designed but rather an emergent mathematical property that the model self-organizes from the corpus under the factorization constraint.

Experiment 6: Spelling Robustness

- Test sentence pairs: Normal sentence ("The scientist is very intelligent"), misspelled sentence ("The scientsit is vary intellgent"), semantically unrelated sentence ("The dog runs in the park")
- Similarity computation: Each of the three sentences is passed through the full Transformer layers, then the final hidden states are projected into the etymological space and cosine similarity is computed
- Robustness metric: $\Delta = \text{sim}(\text{normal}, \text{typo}) - \text{sim}(\text{normal}, \text{different})$; a positive value indicates semantic robustness
- Measured results: Normal vs. misspelled similarity 0.5468, normal vs. semantically unrelated similarity 0.5254, $\Delta = 0.0214 > 0$, though the margin is small and may be limited by the representational granularity of the BPE tokenizer

D.3 Mechanistic Analysis (Experiments 7–9)

Experiment 7: Layer-Wise Probability Evolution

- Test prompts: 3 representative sentences (geographical common knowledge, physical scene, natural attribute)
- Tracking target: Top-6 predicted word probability trajectories at each layer's output
- Entropy computation: Shannon entropy $H = -\sum p_i \log p_i$, in nats
- Visualization: Dual subplot (probability curves + entropy decay), complete tracking across all 36 layers

Measured Results Final predictions and probabilities for the three prompt groups: "The capital of France is" $\rightarrow$ "the" ($p = 8.31\%$); "The cat sat on the" $\rightarrow$ "floor" ($p = 6.72\%$); "The sun is very" $\rightarrow$ "bright" ($p = 29.00\%$). Shannon entropy monotonically decreases across the 36 layers, with predicted word probabilities beginning to converge significantly at middle layers (approximately L15–L20) and stabilizing at deeper layers, exhibiting a clear information crystallization process.

Experiment 8: Signal Flow Tracing

- Signal projection: Hidden state $\mathbf{h}$ projected into etymological space $\mathbf{s} = \mathbf{h}W_{basis}^T$
- Cross-layer analysis: Top 15 highest-variance signals selected, cross-layer heatmap plotted
- Cross-token analysis: Top 20 highest-variance signals selected, cross-token heatmap plotted
- Normalization: RMSNorm applied to each layer's output before projection

Observations The cross-layer heatmap reveals that the high-variance signals identified in Experiment 1 (#898, #822, #277, etc.) maintain high activation throughout all 36 layers, forming horizontal bright bands that traverse the entire network and constitute the backbone channels for semantic transmission. Different token positions activate different signal subsets, but the activity patterns of backbone signals remain consistent across positions, indicating that these signals carry global semantic structure independent of specific tokens.

Experiment 9: Causal Ablation

- Ablation strategy: Signals ranked by contribution to the target word, progressively zeroed out
- Ablation steps: 1, 2, 4, 8, 16, 32, 64, 128 signals (logarithmic scale)
- Contribution computation: $c_k = s_k \cdot W_{recipe}[\text{target}, k]$, where s_k is the signal activation
- Key signal analysis: Extracting the recipe space activation pattern of the highest-contribution signal (Top-8 vocabulary items)

Complete Ablation Data for Three Groups Table 8 presents the stepwise ablation results for three prompt groups. All cases exhibit a highly concentrated causal structure: 1–2 signals suffice to reduce the target word probability to near zero.

D.4 Intervention Validation (Experiments 10–12)

Experiment 10: Task Crystallization Boundary Shift

- Task groups: Short context (4 samples), long context (4 samples), code (4 samples)
- Steering vector: $\mathbf{v}_{steer} = W_{recipe}[\text{target}] - W_{recipe}[\text{base}]$, injected by superimposing $\alpha \cdot \mathbf{v}_{steer} \cdot W_{basis}$ onto the hidden states

Table 8: Complete causal ablation data for three groups

Prompt	Key Signal	Ablated	Residual p	New Prediction
"The capital of France is" Baseline: "the" ($p = 8.31\%$) Key signal #89	codebook: the, a, in, to, an, at	1	0.03%	famous
		2	<0.01%	situated
		4	<0.01%	resists
		128	<0.01%	resists
"The cat sat on the" Baseline: "floor" ($p = 6.72\%$) Key signal #822	codebook: , , and, (, in	1	0.70%	,
		2	0.01%	,
		4	<0.01%	,
		128	<0.01%	,
"The sun is very" Baseline: "bright" ($p = 29.00\%$) Key signal #542	codebook: J, T, -, „ D, W	1	5.29%	,
		2	0.52%	,
		8	0.12%	,
		128	<0.01%	,

- Injection method: Starting from layer L, the steering vector is continuously superimposed at every subsequent layer, testing whether the prediction is successfully flipped
- Critical layer definition: The earliest layer at which flipping fails (even with continuous injection)
- Intensity calibration: First search for the minimum effective α_{min} at L0 with step size 2.0 (search range [2, 50]), then multiply by a 1.2× margin as the fixed intensity

This design, by fixing the intervention intensity and varying only the injection starting point, eliminates the confound of intensity differences in cross-task comparisons. The target words for code tasks are Python syntax continuations (e.g., None), whose structural constraints are stronger than those of natural language, hence the earlier crystallization boundary (L7.8). Each group contains 4 samples; the results are directionally indicative.

Per-Sample Crystallization Boundary Data Table 9 lists the complete boundary layer data for all 12 samples (fixed $\alpha = 2.4$).

Table 9: Per-sample task crystallization boundary data ($\alpha = 2.4$)

Task Group	Prompt (truncated)	Original Pred.	Target	Boundary Layer
Short Context	The capital of France is	the	London	L15
	The cat sat on the	floor	moon	L19
	The sky is	the	red	L22
	Open the door with a	smile	car	L18
			Mean	L18.5
Long Context	When the geography teacher…	Paris	London	L25
	After carefully reviewing…	not	guilty	L35
	When you look outside…	beautiful	red	L20
	I was locked out of…	key	car	L25
			Mean	L26.2
Code	def add(a, b): return a +	b	None	L5
	x = 1 + 2\ny =	3	None	L8
	for i in range(10):…	i	None	L1
	if x > 0:\n result =	0	None	L17
			Mean	L7.8

Experiment 11: Concept Injection

- Test cases: 3 groups of counter-commonsense injections (geography, physics, attributes)
- Injection position: Fixed at the etymological space (final hidden state after all Transformer layers), directly superimposing $\alpha \cdot W_{recipe}$[target]

- Critical value search: Binary search with precision 0.1, search range [0, 200]
- Success criterion: Target word becomes the Top-1 prediction (argmax logits)
- Dose-response curve: 50 uniformly sampled points, plotting target word probability as a function of α

Experiment 12: Gene Pool Hijacking

- Sentiment vector construction: $\mathbf{v}_{pos}$ is the mean of recipe vectors for {excellent, perfect, wonderful, amazing}, $\mathbf{v}_{neg}$ is the mean of recipe vectors for {terrible, bad, disgusting, awful}
- Global modification: $W_{recipe} \leftarrow W_{recipe} + \alpha(\mathbf{v}_{pos} - \mathbf{v}_{neg})$, applied to all tokens in the vocabulary
- Intervention intensity: $\alpha = 1.5$ (empirically tuned)
- Generation parameters: max_tokens=15, greedy decoding (argmax)
- Test prompts (5 groups): "The food was disgusting.", "I failed the exam again.", "The weather is terrible today.", "The project was a complete failure.", "I hate waiting in long lines."
- Safety recovery: Original weights are immediately restored after the experiment to prevent contamination of subsequent experiments

E Detailed Theoretical Derivations

E.1 Factorized Computation Flow

reEtym reformulates the standard Transformer forward propagation into a factorized form:

$$\text{Stage I (Recipe Retrieval):} \quad \mathbf{r}_i = W_{recipe}[i, :] \in \mathbb{R}^S \tag{9}$$

$$\text{Stage II (Signal Projection):} \quad \mathbf{h}_0 = \mathbf{r}_i \cdot W_{basis} = \sum_{k=1}^{S} r_{i,k} \cdot \mathbf{b}_k \in \mathbb{R}^d \tag{10}$$

$$\text{Stage III (Layer-wise Transformation):} \quad \mathbf{h}_{\ell+1} = \mathcal{F}_\ell(\mathbf{h}_\ell; \Theta_\ell), \quad \ell = 0, \ldots, L-1 \tag{11}$$

E.2 Proposition

The Transformer backbone layers learn general processing logic over signal vectors $\mathcal{F}_\ell : \mathbb{R}^d \to \mathbb{R}^d$, rather than hard-coded mappings for specific tokens.

Proof sketch: The parameters Θ_ℓ of layer ℓ are optimized via backpropagation, with gradients derived from the hidden states $\mathbf{h}_\ell$ of all tokens at that layer. Since $\mathbf{h}_\ell$ is a linear combination of etymological bases $\{\mathbf{b}_k\}_{k=1}^S$ (after nonlinear transformations by preceding layers), each layer effectively learns to process vectors in the etymological space—the self-attention layers compute similarities and aggregate information along signal dimensions, while feedforward networks perform nonlinear feature extraction on signal combinations. These operations are defined over the continuous $\mathbb{R}^d$ space, not over the discrete vocabulary index set $\{1, \ldots, V\}$.

Corollary: Modifying the recipe $\mathbf{r}_i$ of token i changes only its initial representation $\mathbf{h}_0$; all layer processing functions $\mathcal{F}_\ell$ and parameters Θ_ℓ remain entirely unchanged. Therefore:

- Processing capacity is conserved: The transformation capabilities of attention mechanisms, feedforward networks, and RMSNorm are fully preserved.
- Processing tendency is altered: As the input signal combination weights $\mathbf{r}_i$ change, the model's semantic tendency when processing that token changes accordingly, while generation fluency and grammatical correctness remain unaffected.

E.3 Intervention Operators

Recipe-level intervention (local modification):

$$W_{recipe}[i,:] \leftarrow W_{recipe}[i,:] + \delta \mathbf{r}_i \tag{12}$$

Scope: Only the embedding of token i is changed.

Basis-level intervention (globally synchronized):

$$W_{basis}[k,:] \leftarrow W_{basis}[k,:] + \delta \mathbf{b}_k \tag{13}$$

Scope: All tokens that activate signal k are simultaneously affected.

Inference-time injection (zero-parameter):

$$\mathbf{h}_\ell \leftarrow \mathbf{h}_\ell + \alpha \cdot \mathbf{b}_k \tag{14}$$

No parameter modification; dynamically injected. Experiments demonstrate a predictable dose-response relationship (mean critical $\alpha = 18.4$).