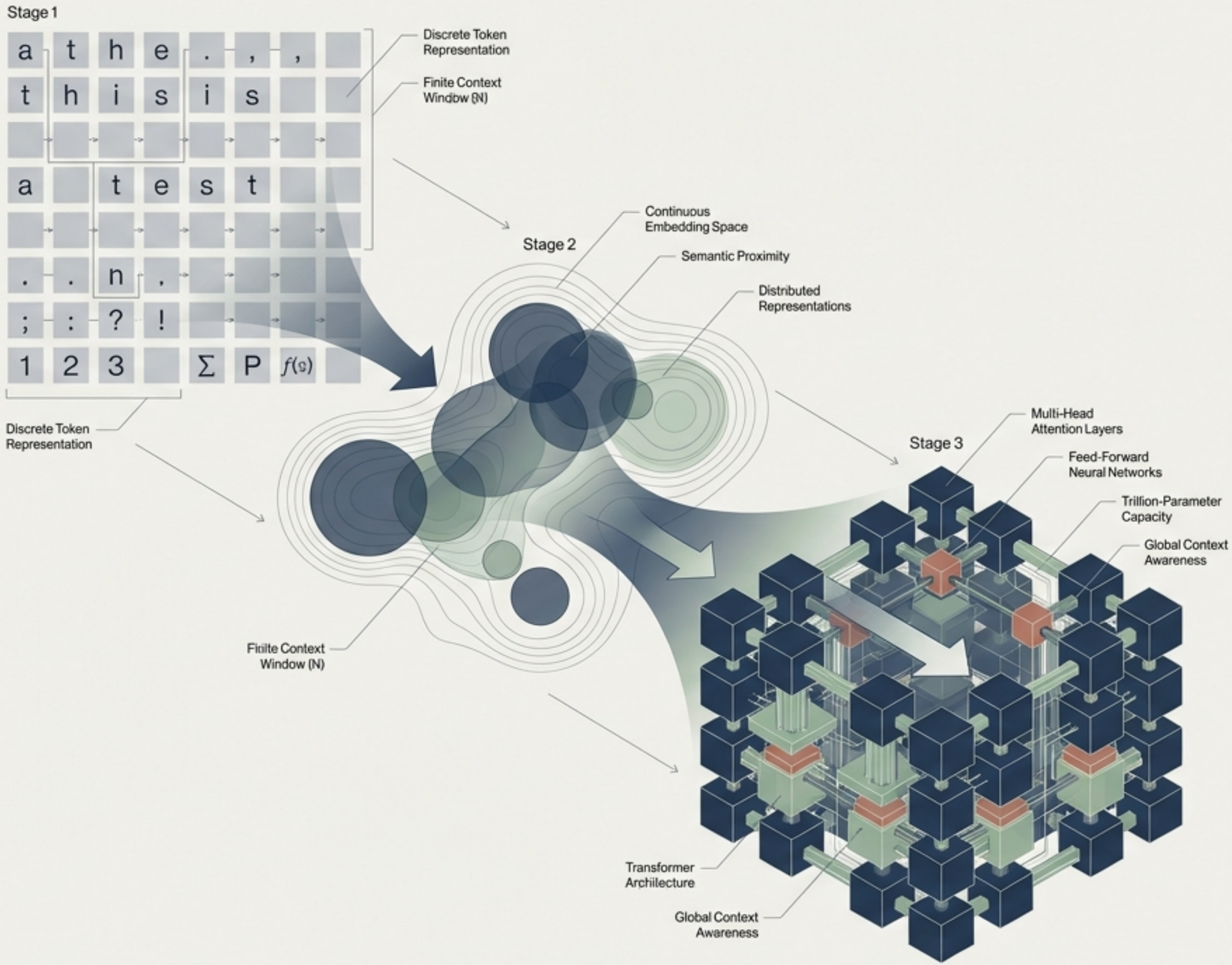
From N-Grams to Trillion-Parameter Giants

The mathematical and architectural evolution of Large Language Models.



NotebookLM

Defining the exact nature of language modeling

The entire scientific challenge reduces to estimating these conditional probabilities over a finite vocabulary.

The Goal

Capture the true data-generating distribution over all possible sequences.

$$P_\theta(\boldsymbol{x}) = \prod_{t=1}^{T} P_\theta(x_t | \boldsymbol{x}_{<t})$$

No Approximations

This decomposition is mathematically exact via the chain rule.

The Context Window

The prefix of previously observed tokens.

NotebookLM

The combinatorial explosion of count-based n-grams

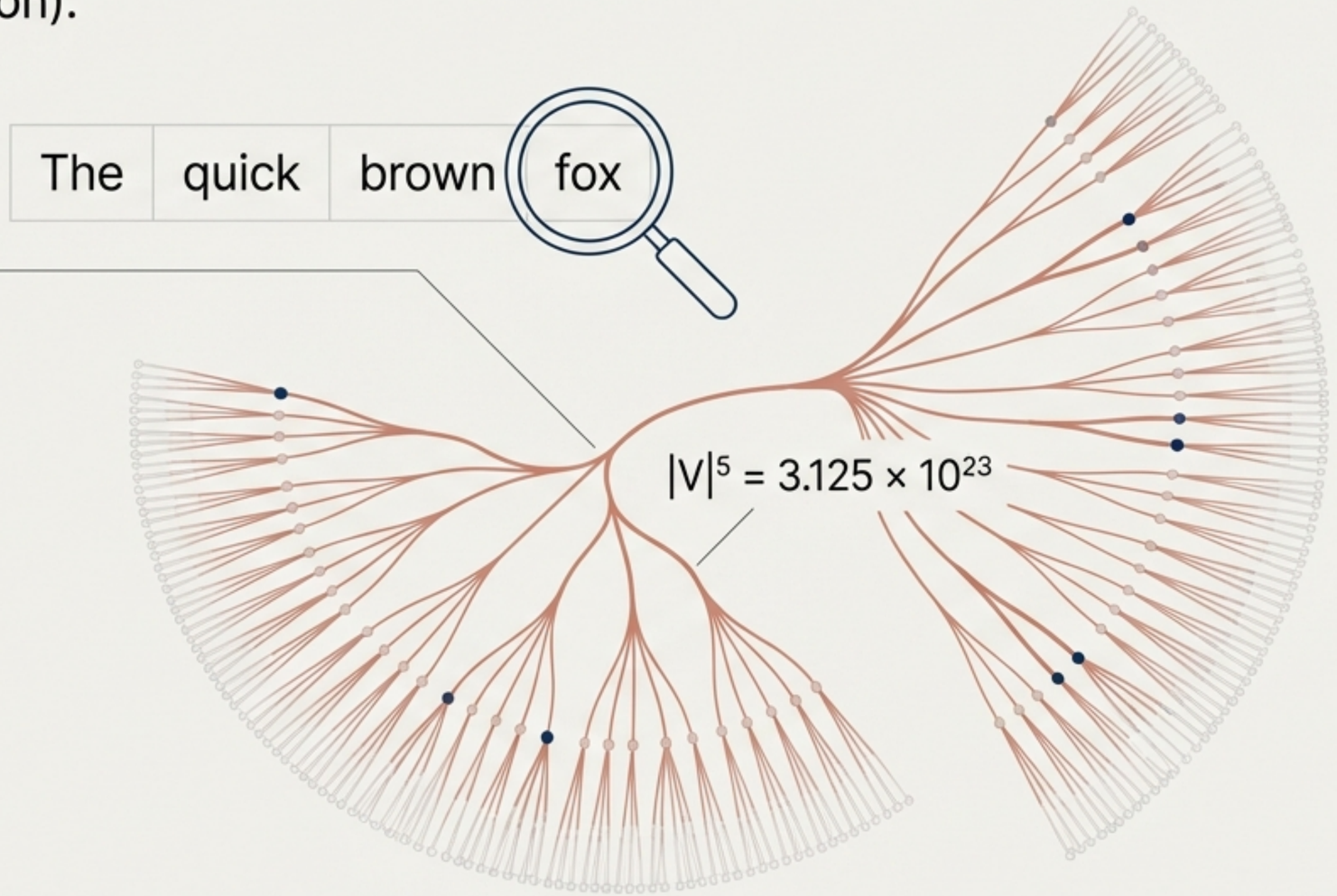
The Approach: Estimating probability by simply counting word occurrences in a fixed context window (Markov assumption).

The Sparsity Bottleneck

For a vocabulary of 50,000 words, a 5-gram model requires tracking $|V|^5 = 3.125 \times 10^{23}$ combinations.

The Flaw

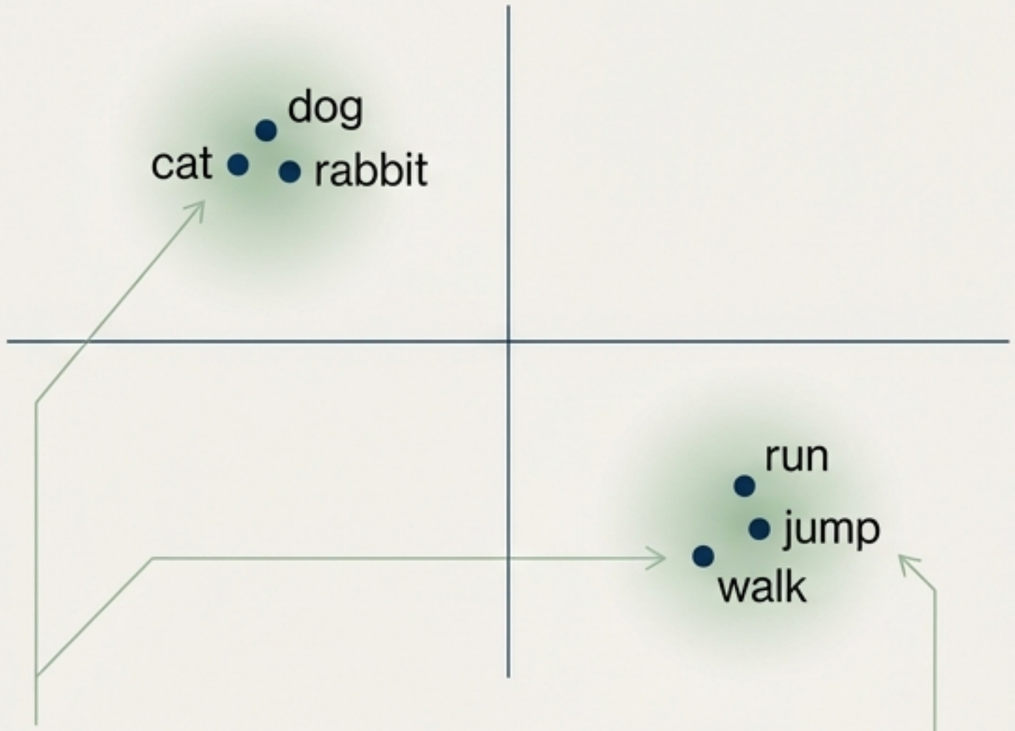
Almost all valid sequences have zero counts. Words are treated as discrete, atomic symbols with no underlying semantic relationship.



NotebookLM

Continuous embeddings solve sparsity but introduce the softmax bottleneck

The trade-off shifts from data sparsity to computational complexity in probabilistic language models.



The Breakthrough (Bengio, 2003)

Mapping words into continuous space e(x). Similar words receive similar vectors, generalizing across unseen contexts.

$$Z = \sum_{v=1}^{|V|} \exp(u_v^T h + d_v)$$

The New Bottleneck

The Softmax denominator requires a computationally crushing summation over 10^5 vocabulary words for every prediction step.

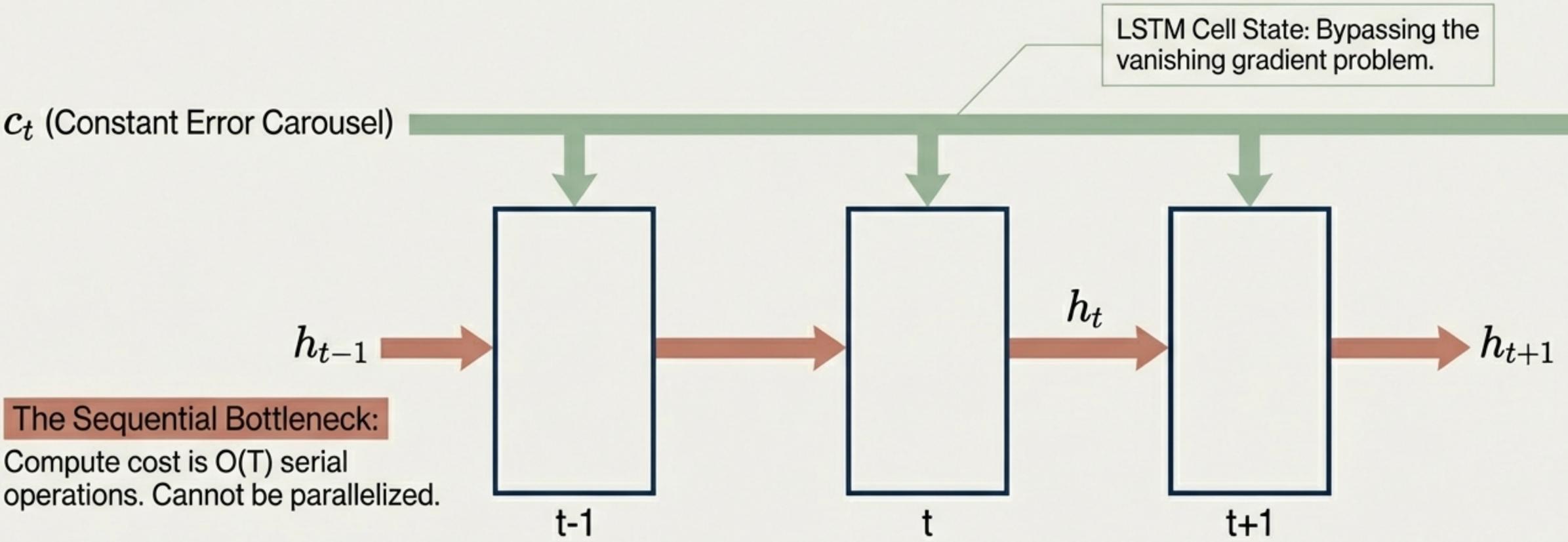
The Fixes: Engineering workarounds like Hierarchical Softmax and Noise Contrastive Estimation (NCE).

NotebookLM

Recurrence captures long contexts but creates a sequential trap

The Breakthrough: RNNs eliminated the fixed-context window via a hidden state. LSTMs stabilized training gradients to capture long-range dependencies.

The Information Bottleneck: Forcing a book's worth of context into a single fixed-dimensional vector limits effective memory to ~500 tokens.



The Sequential Bottleneck:

Compute cost is O(T) serial operations. Cannot be parallelized.

NotebookLM

Self-attention eliminates the sequential bottleneck entirely

The Breakthrough: O(1) path length for long-range dependencies and 100% parallelizable training. Recurrence is officially obsolete.

Query

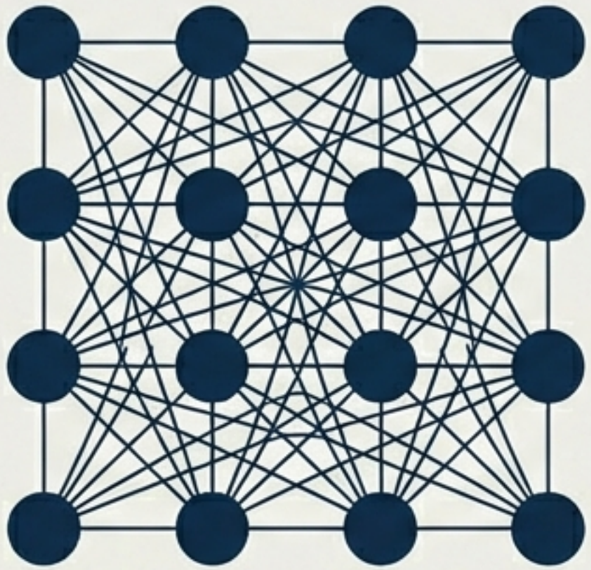
Key

Value

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\,\mathbf{K}^T}{\sqrt{d_k}}\right) V$$

The critical scaling factor.

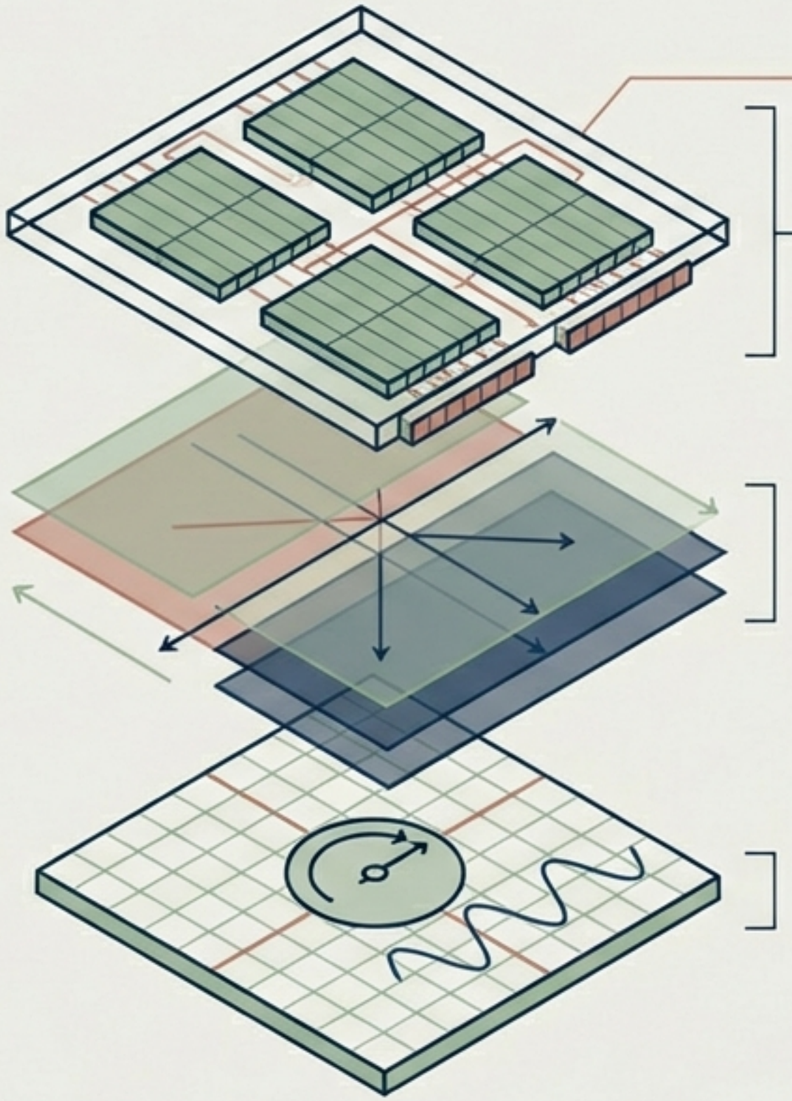
Normalizes variance to prevent softmax gradients from vanishing.



Every position attends to every other position simultaneously.

NotebookLM

Anatomy of the modern Transformer engine



Feed-Forward Networks (FFNs)

Acting as massive key-value memories ($d_{\text{ff}} = 4 \cdot d_{\text{model}}$).

Storing facts and concepts independent of sequential context.

Multi-Head Attention

Each head learns a different linear subspace, simultaneously capturing syntactic, semantic, and positional relational patterns.

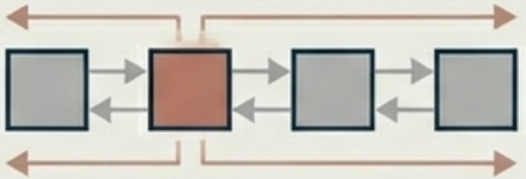
Positional Encoding

Injecting order into an order-blind system. Modern models use RoPE (rotating vectors in 2D subspaces) or ALiBi (linear biases).

NotebookLM

Three distinct architectural lineages emerged from the Transformer

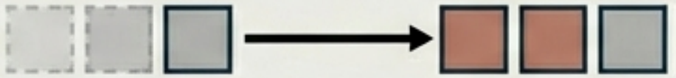
The Breakthrough: O(1) path length for long-range dependencies and 100% parallelizable training. Recurrence is officially obsolete.



Encoder-Only (BERT)

Bidirectional context via Masked Language Modeling (MLM).

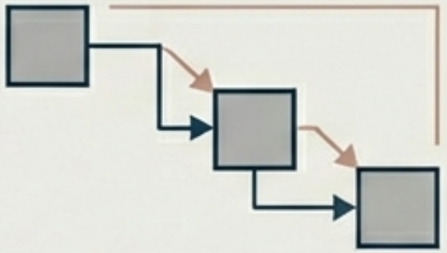
Excels at reading comprehension. Suboptimal for generation due to pseudo-likelihood approximations.



Encoder-Decoder (T5)

Span corruption objectives.

Unified all NLP tasks (translation, summarization) into a singular text-to-text format.



Decoder-Only (The GPT Lineage)

Strictly autoregressive left-to-right generation.

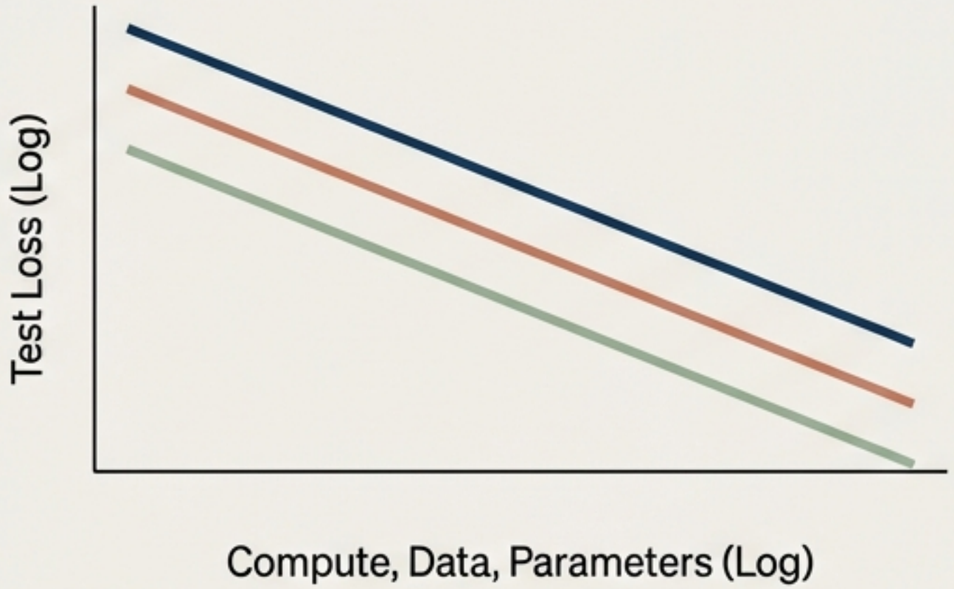
The ultimate architecture for zero-shot transfer and open-ended, in-context learning.

NotebookLM

Loss scales predictably, but capabilities emerge abruptly

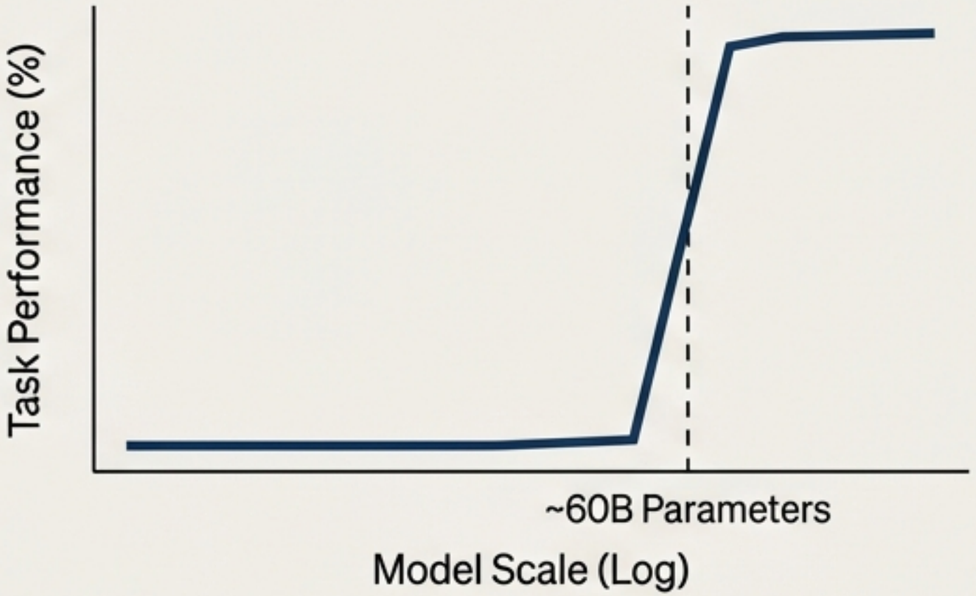
The **Controversy:** Schaeffer et al. argue emergence may just be an artifact of discontinuous, binary evaluation metrics rather than a sudden phase transition.

The Power Laws (Kaplan)



Loss is strictly dictated by the most constrained resource.

Emergent Abilities (Wei)



Skills like chain-of-thought and multi-step arithmetic appear abruptly.

NotebookLM

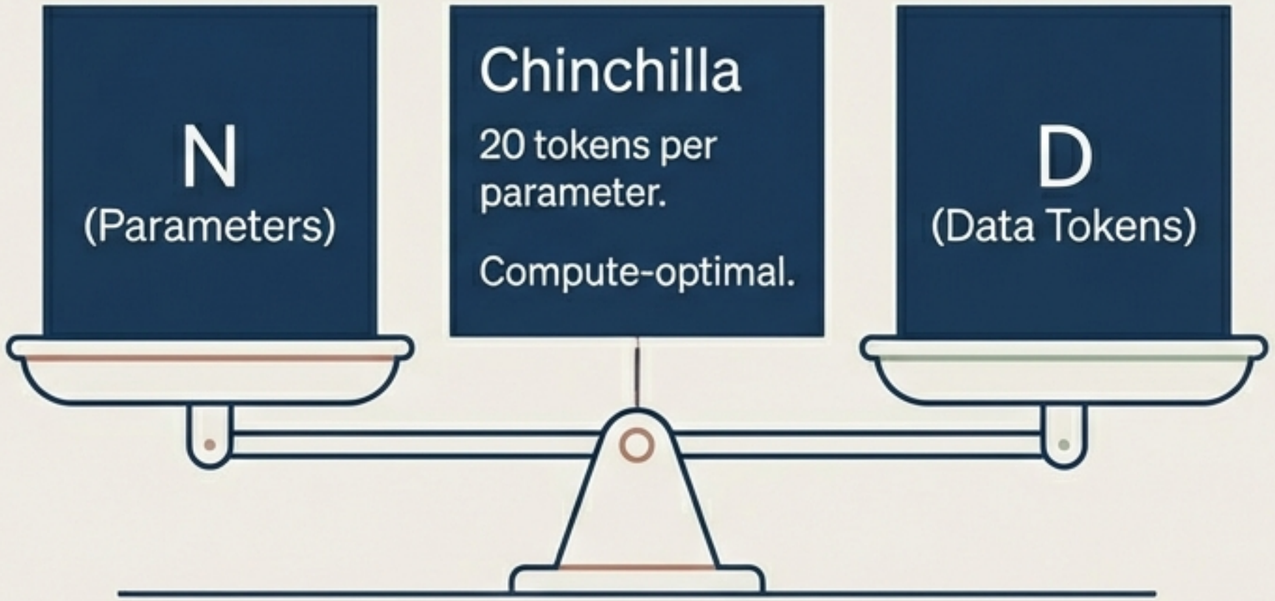
Compute-optimal training demands vastly more data

The **Chinchilla Correction:** Kaplan's early laws drastically under-allocated data. Parameters and data must scale equally.

GPT-3

1.7 tokens per parameter.

Severely under-trained.



LLaMA-3

1875 tokens per parameter.

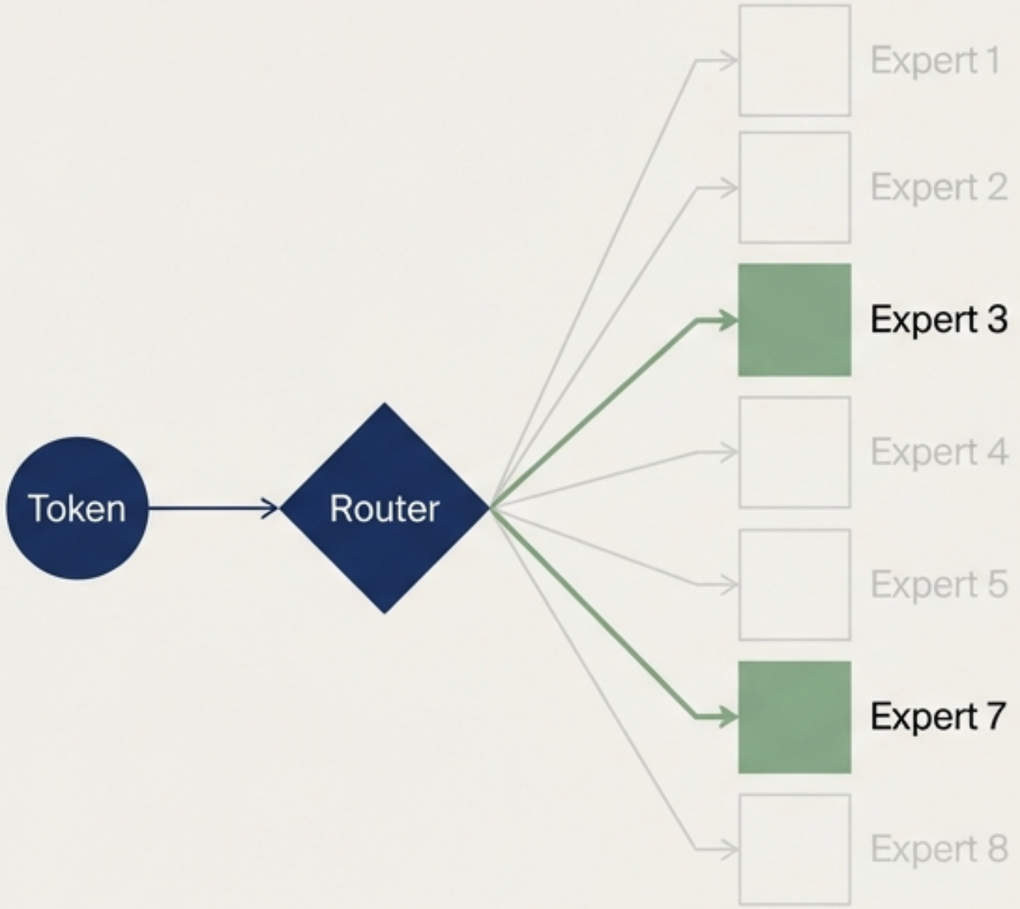
Inference-Optimal: Pushed past optimal to lower downstream deployment costs.

The Compute-Optimal Rule of Thumb:

$$D^* \approx 20 \cdot N^*$$

NotebookLM

Optimizing massive scale with sparse routing and attention tweaks



Sparse routing via Top-K gating. Example: Mixtral 8x7B boasts 46.7B total parameters, but only 12.9B are active per token.

Mixture-of-Experts (MoE)

Sparse routing via Top-K gating.

Example: Mixtral 8x7B boasts 46.7B total parameters, but only 12.9B are active per token.

Attention Efficiency

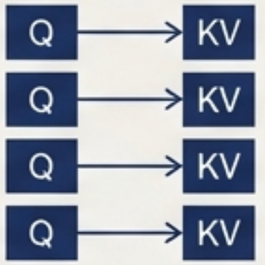


Diagram 1: MHA

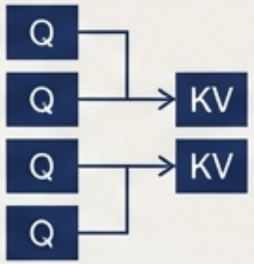


Diagram 2: GQA

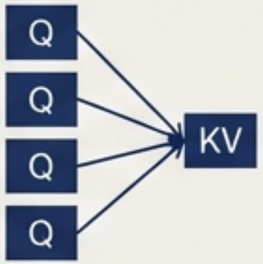


Diagram 3: MQA

Replacing Multi-Head Attention (MHA) with Grouped Query (GQA) or Multi-Query (MQA) trades fractional quality for massive KV-cache memory reductions during inference.

NotebookLM

Transforming base predictors into instruction-following assistants

1. Supervised Fine-Tuning (SFT)

- Show me how to answer.
- Training strictly on highly curated instruction-response data pairs.

2. RLHF

- Learn what humans prefer.
- Training a distinct Reward Model, then using PPO to optimize the policy. Bound by a KL penalty to prevent reward hacking.

3. Direct Preference Optimization (DPO)

- The modern fix.
- Bypassing the reward model entirely. Optimizes the policy directly on human preferences via a simple binary cross-entropy loss.

NotebookLM

The frontier of trillion-parameter and sparse architectures

The frontier is defined by strategic trade-offs between dense predictability, MoE sparsity, and aggressive data over-training.

Model	Parameters (Total / Active)	Training Tokens	Context	Alignment
GPT-4	~1.8T / ~280B (MoE) ~280B	~13T	128K	RLHF
LLaMA-3 70B	70B / 70B (Dense)	15T	128K	RLHF + DPO
Mixtral 8x7B	46.7B / 12.9B (MoE) 12.9B	~2T	32K	SFT + DPO
DeepSeek-V3	671B / 37B (MoE) 37B	14.8T	128K	GRPO

NotebookLM

The scientific journey of sequential computation and scale

1. It is exactly a probability distribution.

Despite emergent capabilities, language models remain strictly autoregressive estimators of conditional probabilities.

2. Progress is removing bottlenecks.

The jump from n-grams, to RNNs, to Transformers was a continuous mission to eliminate sequential constraints and fixed memory limits.

3. Destiny is dictated by Power Laws.

Compute-optimal performance requires balancing parameters and data equally, but deployment economics favor extreme data over-training.

4. Utility requires alignment.

Raw likelihood optimization produces a predictor; specialized post-training bridges the gap between raw statistical probability and human utility.

NotebookLM