Compositional Discovery: Unlocking Self-Organizing Reasoning in Frozen Parameter Spaces

Author: Focus Labs

Abstract

Standard deep learning conflates the processing of information (reasoning) with the storage of information (memorization), leading to massive training costs, catastrophic forgetting, and "black box" opacity. We introduce the Self-Organizing Mixture of Experts (SoME), an architecture that reframes learning not as weight adaptation, but as Compositional Discovery—the process of finding optimal pathways through a fixed, high-entropy library of computational primitives. SoME decouples the expert function from its address, utilizing a gradient-free "Knowledge Gravity" mechanism to cluster frozen experts into semantic guilds. Our experiments on the TinyStories benchmark demonstrate that SoME achieves competitive performance (PPL 2.04) while updating ~96% fewer parameters than standard dense models. We rigorously derive three dynamic laws governing routing-only networks—Router Capacity, Signal Attenuation, and Topological Saturation—and provide qualitative evidence of emergent hierarchical stratification. Our findings suggest a strategic pivot for sparse architectures: moving from monolithic "Knowledge Bases" to modular "Reasoning Cores" ideal for Retrieval-Augmented Generation systems.

1. Introduction

The foundational dogma of modern deep learning is the "Tabula Rasa" assumption: that intelligence must be constructed from scratch by iteratively adjusting the weights of a massive, randomly initialized neural network to minimize a loss function. This paradigm of Weight Adaptation has driven the exponential scaling of Large Language Models (LLMs) [1, 2], but it conflates two distinct computational needs: the processing of information (reasoning) and the storage of information (memorization). As models scale to trillions of parameters, this conflation leads to unsustainable training costs, opacity in decision-making, and the phenomenon of catastrophic forgetting [3].

Biological intelligence operates on a fundamentally different principle. The human brain does not generate new neurons for every new task; rather, it dynamically rewires connections between existing, stable functional columns [4]. Learning, in a biological context, is less about creating new machinery and more about discovering effective pathways through existing machinery.

In this work, we translate this biological insight into a computational architecture. We propose that deep learning can be reframed as a process of Compositional Discovery. We hypothesize that a sufficiently large, high-entropy pool of fixed, random functions—a "primordial soup" of computation—contains, by chance, the sub-circuits necessary to solve complex tasks. The challenge of learning, therefore, shifts from constructing these circuits to finding them.

The Mixture-of-Experts (MoE) architecture [5] offers a promising framework for conditional computation, activating only a sparse subset of parameters per token. However, current state-of-the-art MoEs, such as the Switch Transformer [6] or Mixtral [7], typically train both the experts and the routing mechanism via backpropagation. This creates a coupled optimization problem often plagued by "Router Collapse," where the gating network converges to a trivial solution (selecting the same few experts), necessitating complex auxiliary load-balancing losses to force diversity.

We introduce the Self-Organizing Mixture of Experts (SoME), an architecture that fundamentally decouples a function's knowledge from its address. In SoME, the expert networks are frozen at initialization, serving as an immutable library of primitives. Learning is restricted entirely to the routing layer, which evolves via a set of gradient-free, bio-inspired heuristic rules we term "Knowledge Gravity." These rules—Attraction, Peer-Pull, and Inertia—allow the system to spontaneously organize the frozen experts into semantic clusters ("Knowledge Galaxies") based on their utility to the incoming data stream.

Our contributions are as follows:

1. The SoME Architecture: We define a mechanism for training high-performance language models where the bulk of the parameters (96%+) are never updated by an optimizer.
2. The Physics of Routing: Through extensive ablation studies, we derive three dynamic laws governing routing-only networks: the Law of Router Capacity (necessitating non-linear manifold warping), the Law of Signal Attenuation (limiting network depth via semantic tethering), and the Law of Topological Saturation (governing the scaling of expert counts).
3. Emergent Stratification: We provide qualitative evidence that SoME spontaneously separates concerns, organizing experts into distinct hierarchical roles (e.g., "Lexical," "Conceptual," and "Narrative" layers) without supervision.

By treating the expert library as a fixed resource, SoME exposes a critical strategic pivot for neural architecture search: moving away from monolithic "Memorization Engines" toward modular, adaptable "Reasoning Cores" suitable for the next generation of Retrieval-Augmented Generation (RAG) systems.

---

2. Related Work

Our work sits at the intersection of conditional computation, reservoir computing, and self-organizing systems. We synthesize concepts from these distinct fields to propose a new framework for training sparse neural networks.

2.1 Mixture-of-Experts (MoE) in Transformers

The resurgence of conditional computation in deep learning is largely driven by the scaling limits of dense models. Seminal works such as the Sparsely-Gated Mixture-of-Experts [5] and GShard [8] demonstrated that model capacity could be decoupled from inference cost by activating only a subset of parameters per token. This lineage culminated in the Switch Transformer [6] and Mixtral [7], which established the standard paradigm: a trainable gating network selecting from a set of trainable expert networks.

However, this co-evolution of router and experts introduces significant optimization instability. To prevent "representational collapse"—where the router trivially selects a single expert for all inputs—these architectures rely on complex auxiliary load-balancing losses. SoME diverges fundamentally from this paradigm. By freezing the expert weights, we eliminate the non-stationary target problem for the router. Furthermore, our gradient-free "Knowledge Gravity" mechanism replaces explicit auxiliary losses with implicit topological constraints, ensuring diversity through the physics of the vector space rather than penalty terms in the objective function.

2.2 Random Projections and the Lottery Ticket Hypothesis

The efficacy of our "Primordial Soup" initialization—using frozen, random experts—builds upon the theoretical foundations of Reservoir Computing and the Lottery Ticket Hypothesis [9]. These fields posit that within a sufficiently large, randomly initialized parameter space, highly effective subnetworks exist prior to any training.

Previous approaches, such as Echo State Networks [10] or fixed-weight visual encoders, utilize random projections as passive feature extractors. SoME advances this by treating the random projection not as a fixed transformation, but as a searchable landscape. We do not merely project data through the reservoir; we learn a policy to navigate it. Our findings align with the "Supermask" approaches [11], but instead of masking weights within a dense network, we route tokens to modular functional blocks, preserving the sparsity benefits of MoEs.

2.3 Self-Organizing Maps and Competitive Learning

The learning mechanism in SoME is a modern adaptation of Competitive Learning, specifically drawing from Kohonen's Self-Organizing Maps (SOMs) [12] and Learning Vector Quantization (LVQ).

Standard SOMs are typically used for unsupervised dimensionality reduction or visualization. SoME repurposes these principles for deep neural routing. Our Attraction (Alpha) rule functions as an Inverse-Hebbian learning signal ("keys that are queried together move toward the query"), while our Peer-Pull (Beta) rule implements the neighborhood function essential for topological preservation. To our knowledge, this is the first application of SOM-style heuristics to organize the residual stream of a multi-layer Transformer, effectively embedding a differentiable associative memory directly into the reasoning path of the network.

3. Methodology: The SoME Architecture

The Self-Organizing Mixture of Experts (SoME) replaces the standard dense Feed-Forward Network (FFN) block of a Transformer with a sparse, topologically organized routing layer. The architecture is defined by two fundamental decoupling operations:

1. Decoupling Computation from Definition: The functional primitives (experts) are fixed at initialization.
2. Decoupling Optimization from Representation: The addressing system (keys) evolves via heuristic self-organization, separate from the gradient-based optimization of the query network.

[Reference Figure 1: Diagram of the SoME Layer illustrating the frozen expert pool, the dynamic key store, and the update vectors acting on the keys]

3.1 The Primordial Soup: Frozen Expert Library

Let $E=\{e1,e2,...,eN\}$ be a library of N expert networks. In our implementation, each expert ei is a standard FFN with a bottleneck architecture:

- $ei(x)=Wup(i)(\sigma(Wdown(i)x))$

where σ is a non-linear activation function (GELU).
Crucially, the weights $W(i)$ are initialized randomly and permanently frozen (requires_grad=False). This creates a "Primordial Soup" of computational primitives. Through ablation studies, we determined that Sparse Initialization yields superior performance compared to orthogonal or standard Gaussian initialization. Sparse random matrices act as compressive sensing projections, maximizing the functional entropy of the library and ensuring distinct "tools" for the router to discover.

3.2 Topological Routing (The Forward Pass)
The routing mechanism is framed as a high-dimensional similarity search rather than a classification problem.
The Query Network (The Explorer):
To retrieve an expert, the model must map the current token state $x \in Rdmodel$ into the routing space. Unlike standard linear routers, we employ a Multi-Layer Perceptron (MLP) for this projection:

- $q=Normalize(MLP\theta(x))$

This non-linear projection is critical. As established by our Law of Router Capacity, a simple linear map is insufficient to "unfold" the complex semantic manifold of the token stream onto the fixed topology of the expert keys. The MLP provides the necessary degrees of freedom to warp the input space to match the experts.
The Key Store (The Map):
Each expert ei is assigned a learnable address key $Ki \in Rdmodel$. The routing scores S are computed via the dot product between the query and the keys:

- $S=q \cdot KT$

We select the top-k experts with the highest scores. The final output is a weighted sum of the selected experts, composed via a residual connection:

- $y=x+\sum i \in top\text{-}kSoftmax(Si) \cdot ei(x)$

3.3 The Physics of Knowledge Gravity (The Update Rules)
While the Query Network parameters θ are updated via standard backpropagation to minimize the language modeling loss $LLLM$, the Key Store K is updated via a separate, gradient-free process we term Knowledge Gravity.
These heuristic rules are applied after every training step, physically moving the keys in the vector space to organize the library.

1. Attraction Rule (α):
This is the primary driver of specialization (Inverse-Hebbian learning). A key Ki moves toward the centroid of all queries q that selected it:

- $\Delta Ki=\alpha \cdot (q-Ki)$

This ensures that an expert's address becomes the semantic center of the tokens it processes best.

2. Peer-Pull Rule (β):

This is the engine of semantic clustering. If expert i and expert j are frequently co-activated in the top-k set for the same token, they are pulled toward each other:

- $\Delta Ki=\beta \cdot (Kj-Ki)$

This topology-preserving force causes functionally covariant experts to form dense clusters ("Knowledge Galaxies"), organizing the space into semantic neighborhoods.

3. Usage Inertia (Stability):

To prevent a "rich-get-richer" collapse where a single generalist expert dominates the space, we scale the learning rates α and β by the expert's usage frequency, tracked via an Exponential Moving Average (EMA). As an expert becomes massive (popular), it becomes harder to move:

- $\alpha\text{effective}=\alpha 1+\text{Usage}i$

This forces the router to explore lighter, under-utilized experts for new concepts, maintaining the Adaptive Equilibrium of the system.

---

4. Experimental Setup

To empirically validate the SoME architecture and derive the dynamic laws governing its behavior, we designed a two-phase experimental protocol. Phase 1 focuses on Internal Ablation, stress-testing the architecture to understand the physics of routing. Phase 2 focuses on External Comparison, benchmarking SoME against standard dense Transformers to quantify the trade-off between reasoning and memorization.

4.1 Datasets: The Composition vs. Memorization Spectrum

We selected two distinct datasets to isolate specific cognitive capabilities:

- TinyStories (The Reasoning Benchmark): A synthetic dataset of short stories generated by GPT-3.5/4 using a limited vocabulary [13].
    - Rationale: This dataset requires high compositional ability (grammar, narrative consistency, logical flow) but almost zero factual memorization. It is the ideal testbed for our hypothesis that SoME acts as a "Reasoning Engine."
- TinyTextbooks (The Memorization Benchmark): A collection of educational textbook-style data.
    - Rationale: This dataset requires the storage of specific definitions, facts, and rigid structural patterns. We utilize this to test the "Semantic Tether" limits of our frozen expert library.

4.2 Model Architectures & Baselines

We compare two primary architectural classes. To ensure a fair comparison of learning efficiency, we benchmark based on trainable parameters, acknowledging that SoME incurs a larger static memory footprint due to the frozen experts.

1. The Dense Baseline:

A standard decoder-only Transformer (GPT-2 style) with learned weights in all layers.

- Configuration: dmodel=256, Layers=4-6 (scaled to match SoME's trainable footprint).
- Scale: Approximately 33.6M trainable parameters.

2. The SoME Candidate (v4):

Our proposed architecture featuring the MLP Router and frozen sparse experts.

- Configuration (Run A4 "Hero"): dmodel=512, Layers=10, Heads=8.
- Expert Library: N=64 to 512 experts, dffn=1024.
- Scale: Total parameter count ranges from 700M to 1.6B (due to the frozen library), but the trainable parameter count is only ~18.9M.
- Key Characteristic: SoME operates with approximately 56% of the trainable parameters of the dense baseline, leveraging its massive frozen "long-term memory" instead of updating weights.

4.3 Training Protocol & Hyperparameters

All models were trained using the AdamW optimizer with a cosine annealing schedule.

The Hybrid Optimization Loop:

A critical divergence from standard training is the SoME update cycle:

1. Gradient Step: Standard backpropagation updates the Query Network (θ) and the self-attention layers.
2. Heuristic Step: The "Knowledge Gravity" rules update the Key Store (K) entirely without gradients (@torch.no_grad).

SoME Hyperparameters:

Based on our preliminary tuning, we established the following defaults for the heuristic physics:

- Attraction (α): 0.01 (Controls the learning rate of specialization).
- Peer-Pull (β): 0.005 (Controls the strength of clustering).
- Inertia Decay: 0.99 (Controls the resistance of established experts).
- Router: MLP (d→2d→d) with GELU activation.

---

5. Results & Analysis

Our experiments reveal that training a routing-only network is not merely a matter of hyperparameter tuning, but of managing specific physical dynamics within the vector space. We categorize our findings into three governing laws of self-organization, followed by a comparative analysis of the architecture's capabilities.

5.1 The Law of Router Capacity (Manifold Warping)

Our initial hypothesis—that a simple linear projection could map token representations to random expert keys—was decisively refuted.

In our preliminary Linear Router experiments (Ablation Series V, Run V2), the model suffered catastrophic Router Collapse. With N=512 experts, the Gini coefficient rose to 0.980, indicating a near-total winner-take-all dynamic where the router utilized only a negligible fraction of the expert library. Consequently, validation perplexity stagnated at 5.09.

We hypothesized that this failure stemmed from a topological mismatch: the linear router lacked the degrees of freedom to "unfold" the complex semantic manifold of the input tokens to align with the fixed, random topology of the expert keys.

Validation: In Series A (Run A4), we introduced the MLP Router (d→2d→d) with non-linear activation. This single architectural change stabilized the system. Under comparable conditions, the Gini coefficient dropped to a healthy 0.851, and perplexity improved dramatically to 2.04.

This confirms the Law of Router Capacity: The expressive power of the routing mechanism must be proportional to the entropy of the expert landscape. The router does not just select; it performs a manifold warping operation to make data retrieval possible.

5.2 The Law of Signal Attenuation (The Semantic Tether)

While increasing network depth typically improves performance in standard Transformers, we discovered a hard "Depth Limit" in SoME, governed by the degradation of the heuristic signal.

- Optimal Depth: Run A4 (10 Layers) achieved our best internal perplexity of 2.04.
- The Anomaly: Run A6 (16 Layers) collapsed entirely, ending with a perplexity of 7.16 and a high Gini of 0.922.

Interpretation: The "Knowledge Gravity" update rules are local—keys update based on the query vector at their specific layer. As information propagates through deep networks, the residual stream undergoes significant rotation and abstraction. At extreme depths (L=16), the representation loses its Semantic Tether to the initialization space. The query vector q becomes highly abstract, and its correlation with the random "primordial soup" of keys becomes noise. The router, unable to find a strong gradient of utility, reverts to a safe, static equilibrium (high Gini), effectively ceasing to learn.

5.3 The Law of Topological Saturation (The Crowd Problem)

Scaling the number of experts (N) is the standard path to increasing capacity in MoEs. However, our Series B ablations reveal a "Crowd Problem" unique to coordinate-based routing.
Keeping the top-k selection constant at k=4:

- N=64 (Run B1): Gini 0.851 (Healthy distribution).
- N=256 (Run B2): Gini 0.914.
- N=512 (Run B3): Gini 0.972 (Critical Concentration).

As the number of experts increases within the finite volume of the vector space (hypersphere), the density of keys increases. Without increasing the search radius, the "gravitational wells" of the initial dominant experts overlap, creating an oligarchy that new, smaller experts cannot penetrate.

Mitigation: In Run C2, we increased the active experts to k=8 for the N=256 case. This reduced the Gini to 0.862 and restored perplexity parity with the smaller models. This establishes the Law of Topological Saturation: To maintain diversity, the routing search radius (k) must scale linearly with expert density (N).

5.4 Comparative Analysis: Reasoning vs. Memorization

Finally, we benchmarked a 1.6B parameter SoME model (18M trainable) against a standard Dense Transformer (33M trainable) to quantify the functional trade-offs.

1. The Reasoning Benchmark (TinyStories):

- Dense Baseline: PPL 2.39
- SoME Candidate: PPL 2.77
- Result: SoME achieves competitive performance, trailing the dense baseline by only ~15% in perplexity despite having ~45% fewer trainable parameters. This confirms that for compositional tasks—grammar, narrative structure, and logic—a fixed library of random functions provides sufficient computational primitives.

2. The Memorization Benchmark (TinyTextbooks):

- Dense Baseline: PPL 8.04
- SoME Candidate: PPL 10.70
- Result: The gap widens significantly (~33%). SoME struggles to match the dense model. This verifies our hypothesis that SoME functions as a Composition Engine, not a Memorization Engine. Because the experts are frozen, the model cannot "write" new facts (e.g., historical dates, specific definitions) into its weights. It can only arrange existing logic gates to approximate the data.

---

6. Discussion: The Theory of Compositional Discovery

The empirical results presented above validate the fundamental thesis of this work: that a neural network can learn to solve complex tasks not by creating new computational circuits, but by discovering and organizing existing ones. This reframes the "Black Box" of deep learning into a transparent, observable system of Compositional Discovery.

6.1 Emergent Functional Stratification

The most compelling evidence for the efficacy of our "Knowledge Gravity" heuristics lies in the qualitative analysis of the expert traces. The system did not merely minimize perplexity; it spontaneously organized language into a functional hierarchy, validating our Distinct Primitives Hypothesis.

Our multi-layer generative trace analysis reveals a clear separation of concerns that emerged without explicit supervision:

- The Lexicon (Layer 1): We observed experts (e.g., [39, 43, 30, 42]) that activate consistently based on syntactic identity and adjacent dependencies. These experts function as the model's tokenizer-level processors, handling the raw mechanics of the input.
- The Mind's Eye (Layer 4-5): This is where the Peer-Pull (Beta) rule demonstrates its power. We identified a stable cluster of experts—specifically the group [55, 20, 23, 1]—that formed a "Noun Guild." This cluster activates principally for abstract conceptual entities (e.g., "fox," "robot," "cake"), regardless of their specific context. This proves that the system physically migrated functionally covariant experts into the same topological neighborhood, effectively "growing" a semantic processing lobe in a random network.
- The Storyteller (Layer 7-9): In the deeper layers, the routing logic shifts from content to context. We observed a distinct set of experts (e.g., [58, 50, 6, 26]) that track narrative flow, activating on conjunctions, prepositions, and structural markers. These experts appear to manage the "state" of the story, independent of the specific nouns involved.

This hierarchical emergence confirms that SoME is not just memorizing statistical correlations; it is decomposing the problem space into "what is it" (Lexical/Conceptual) and "where does it fit" (Narrative) sub-problems, and assigning specific regions of the random expert library to solve them.

6.2 The Strategic Pivot: From "Model" to "Reasoning Core"

The comparative performance on TinyStories vs. TinyTextbooks (Section 5.4) exposes the defining constraint of the SoME architecture, which paradoxically reveals its greatest strategic value.

Standard Transformers are Memorization Engines; they encode world knowledge (facts, dates, definitions) directly into their synaptic weights. SoME, with its frozen "primordial soup" core, is a poor memorizer. It cannot "write" a new phone number into a neuron because that neuron is immutable. This explains the performance lag on the fact-heavy TinyTextbooks dataset.

However, SoME excels as a Composition Engine. It learns the logic of how to process information (grammar, narrative consistency, causal reasoning) by stringing together fixed logic gates.

The Implication for RAG:

This limitation positions SoME as the ideal architecture for the Reasoning Core of a Retrieval-Augmented Generation (RAG) system. In the current paradigm, Large Language Models suffer from "hallucination" because they conflate their internal parametric memory with external data. A SoME-based system decouples these entirely:

1. Facts are stored externally (in a Vector Database).
2. Logic is stored internally (in the SoME Routing Policy).

Because a SoME model cannot memorize training data in its weights, it is structurally resistant to overfitting on facts and arguably incapable of "hallucinating" knowledge it does not possess. It provides the pure, untainted reasoning circuitry required to synthesize retrieved information, offering a potential architectural solution to the reliability crisis in generative AI.

---

7. Conclusion & Future Work

We have introduced the Self-Organizing Mixture of Experts (SoME), an architecture that challenges the dominant paradigm of weight adaptation in deep learning. By treating the bulk of a neural network as a fixed, high-entropy resource and restricting learning to a topological routing layer, we have demonstrated that intelligence can emerge from Compositional Discovery rather than construction.

Our extensive ablation studies have formalized the physics of this new learning mode. We identified the Law of Router Capacity, proving that non-linear manifold warping is essential for navigating static functional landscapes. We established the Law of Signal Attenuation, creating a roadmap for scaling network depth by maintaining semantic tethers. Finally, the emergence of the "Noun Guild" and "Storyteller" clusters provides the first tangible evidence that gradient-free "Knowledge Gravity" heuristics can spontaneously organize a chaotic parameter space into a hierarchical reasoning engine.

While SoME currently trails dense transformers in rote memorization tasks, it offers a distinct advantage in parameter efficiency and compositional reasoning. It points toward a future where "training a model" does not mean updating billions of weights, but simply learning to navigate the ones we already have.

7.1 Future Work

The principles uncovered in this work provide a clear roadmap for advancing the SoME paradigm:

1. Grounding the Router (Solving the Depth Limit):
To mitigate the signal attenuation observed in deep networks (Run A6), future work will implement Residual Query Connections. By effectively "grounding" the router's input with a skip-connection to the original token embedding or earlier layer representations, we hypothesize we can maintain the "scent" of the initialization space even at extreme depths (L>50), stabilizing the "Knowledge Gravity" signal.

2. Scaling to Millions of Experts (ANN Routing):
Our current implementation relies on exact matrix multiplication for routing (O(N)), which hits a computational bottleneck as N scales. The "lookup" nature of SoME is perfectly suited for Approximate Nearest Neighbor (ANN) algorithms. Replacing the exact dot product with a dynamic HNSW (Hierarchical Navigable Small World) index would allow the expert library to scale to millions of primitives (N>106) with logarithmic search time (O(logN)), unlocking the true potential of massive sparse models.

3. The "Graveyard of Models" (Transfer Learning):
Perhaps the most exciting frontier is replacing the random "Primordial Soup" with a "Junkyard" of pre-trained parts. Instead of initializing experts with random noise, we propose populating the expert library with frozen FFN layers extracted from obsolete or specialized open-source models (e.g., BERT, Llama-2-7B, ResNet). In this "Meta-Model" configuration, SoME would act as a Universal Composer, learning to route information through a heterogeneous library of high-quality, pre-trained circuits, effectively recycling global compute efforts into a single, adaptable system.

References

[1] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., ... & Amodei, D. (2020). Scaling laws for neural language models. arXiv preprint arXiv:2001.08361.
[2] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.
[3] McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. The psychology of learning and motivation, 24, 109-165.
[4] Mountcastle, V. B. (1997). The columnar organization of the neocortex. Brain, 120(4), 701-722.
[5] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., & Dean, J. (2017). Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538.
[6] Fedus, W., Zoph, B., & Shazeer, N. (2021). Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. Journal of Machine Learning Research, 23(120), 1-39.
[7] Jiang, A. Q., Sablayrolles, A., Roux, A., Mensch, A., Savarese, B., Bamford, C., ... & Sayed, W. E. (2024). Mixtral of experts. arXiv preprint arXiv:2401.04088.
[8] Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., ... & Zhifeng, C. (2020). GShard: Scaling giant models with conditional computation and automatic sharding. arXiv preprint arXiv:2006.16668.

[9] Frankle, J., & Carbin, M. (2018). The lottery ticket hypothesis: Finding sparse, trainable neural networks. arXiv preprint arXiv:1803.03635.
[10] Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. Bonn, Germany: German National Research Center for Information Technology GMD Technical Report, 148(34), 13.
[11] Ramanujan, V., Wortsman, M., Kembhavi, A., Farhadi, A., & Rastegari, M. (2020). What's hidden in a randomly weighted neural network?. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
[12] Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. Biological cybernetics, 43(1), 59-69.
[13] Eldan, R., & Li, Y. (2023). Tinystories: How small can language models be and still speak coherent english?. arXiv preprint arXiv:2305.07759.