GRCLM: A Geometric Routing & Continual Learning Model for Self-Organizing Language Models

Michael T. Herber II

November 25th, 2025

Abstract

Large language models (LLMs) have achieved impressive performance across a wide range of tasks, but they are typically trained in a static, offline regime and deployed as fixed artifacts. As a result, they cannot autonomously discover new domains, create new experts, or adapt their internal structure in response to evolving, open-world query distributions. This paper introduces the *Geometric Routing & Continual Learning Model* (GRCLM), a novel architecture that integrates geometric domain routing, emergent out-of-distribution (OOD) detection, and continual learning into a single self-organizing system.

GRCLM operates in an embedding space endowed with domain-specific centroids and covariance structures. A geometric domain router combines cosine similarity, Mahalanobis distance, predictive entropy, and empirical support ratios into a composite routing score used to select specialist experts or flag queries as belonging to novel domains. Unknown-domain queries are accumulated into emergent clusters; when stability and density criteria are satisfied, GRCLM spawns a new specialist head within a shiftable transformer architecture and triggers a continual learning cycle that updates the shared trunk and recalibrates the domain geometry.

We present a rigorous mathematical formulation of the GRCLM routing equations, domain topology, and emergence conditions, and we analyze key properties of the composite routing function. We formalize the continual learning loop that allows GRCLM to expand its domain taxonomy over time, and we outline experimental protocols for evaluating routing accuracy, emergent domain quality, and stability under domain drift. GRCLM provides a concrete and implementable foundation for lifelong-learning LLMs and distributed cognitive mesh systems.

1 Introduction

Large language models (LLMs) such as GPT-style transformers have become a central paradigm in modern artificial intelligence, exhibiting strong performance on language understanding, generation, and reasoning tasks [Brown et al., 2020, OpenAI, 2023]. Despite their capabilities, mainstream LLMs are fundamentally *static*: they are pretrained on a large corpus, optionally fine-tuned on specific tasks, and then deployed as fixed models. Updating such models requires repeating expensive training procedures and manual curation of new datasets.

In contrast, many real-world environments are *open-world* and *non-stationary*. New domains—such as emerging technologies, policies, or social phenomena—appear continuously. Query distributions shift over time. Rigid, static models fail to adapt gracefully to such evolving contexts.

This motivates the question:

Can we design an LLM architecture that can (i) recognize when it is operating in a previously unseen domain, (ii) autonomously create new experts for such domains, and (iii) integrate these experts into its behavior through continual learning, all while preserving performance on existing domains?

To address this question, we propose the **Geometric Routing & Continual Learning Model (GRCLM)**. GRCLM combines three core ideas:

(a) **Geometric Domain Routing:** A domain router operates in an embedding space, where each domain is associated with a centroid and covariance matrix. For a query embedding, the router computes similarity, divergence, uncertainty, and empirical support to form a composite routing score.

(b) **Emergent Expert Creation:** Queries that cannot be confidently assigned to any existing domain are accumulated into an emergent buffer. Once sufficient evidence is collected and the cluster exhibits low intra-variance, a new domain and specialist head are created.

(c) **Continual Learning in a Shiftable Transformer:** The model maintains a shared transformer trunk and domain-specific specialist heads. When a new expert is created, a continual learning cycle refines the trunk and trains the new head, while recalibrating the domain geometry.

By closing the loop between routing, emergence, and learning, GRCLM behaves as a *self-organizing expert system* capable of growing its internal domain taxonomy over time. This architecture is also compatible with broader visions of Cognitive Mesh Intelligence (CMI), in which multiple GRCLM nodes share domain knowledge in a distributed topology.

Contributions. This paper makes the following contributions:

- We introduce GRCLM, a new architecture for self-organizing domain specialization in LLMs, based on geometric routing and continual learning.
- We derive a composite routing function that integrates similarity, Mahalanobis divergence, predictive entropy, and support ratios, and we analyze its properties.
- We formalize emergent domain creation criteria using cluster density and variance constraints and relate them to OOD detection and clustering theory.

- We define a shiftable transformer architecture with a shared trunk and specialist heads, and we describe a continual learning loop that allows the model to incorporate new experts over time.
- We propose experimental designs and evaluation metrics for assessing GRCLM's routing accuracy, emergent domain quality, and stability under domain drift.

2 Related Work

2.1 Mixture-of-Experts and Sparse Models

Mixture-of-Experts (MoE) architectures introduce multiple expert subnetworks with a gating mechanism that selects a subset of experts per input [Jacobs et al., 1991]. In large-scale language modeling, Lepikhin et al. [2020] and Fedus et al. [2022] demonstrated that MoE layers can significantly increase model capacity while maintaining computational efficiency. GLaM [Du et al., 2022] and BASE Layers [Lewis et al., 2021] further refined sparse expert models for language tasks.

These systems, however, share important limitations: the set of experts is typically fixed at training time, and the gating network is trained jointly with the experts. Adding new experts after deployment is non-trivial and generally requires retraining the gate on large volumes of data. GRCLM departs from this paradigm by using a *geometric* router instead of a learned gate, enabling dynamic addition of new experts without retraining a gating network.

2.2 Continual Learning

Continual learning (or lifelong learning) studies how models can acquire new knowledge over time without catastrophic forgetting of previous tasks. Techniques include regularization-based approaches such as Elastic Weight Consolidation (EWC) [Kirkpatrick et al., 2017], memory-based replay [Lopez-Paz and Ranzato, 2017, Rebuffi et al., 2017], and modular architectures with task-specific components [Rusu et al., 2016].

In the context of LLMs, parameter-efficient fine-tuning methods such as adapters [Houlsby et al., 2019] and LoRA [Hu et al., 2022] offer mechanisms for specialization with limited additional parameters. However, these methods generally assume predefined tasks or domains. GRCLM contributes a task-agnostic continual learning mechanism where "tasks" are *emergent domains* discovered from the query stream itself.

2.3 Out-of-Distribution Detection

OOD detection aims to determine when an input does not belong to the training distribution. Hendrycks and Gimpel [2017] studied OOD detection using softmax confidences, while Lee et al. [2018] proposed a Mahalanobis distance-based method operating in feature space. Energy-based methods [Liu et al., 2020] and density-based approaches have also been explored.

GRCLM uses Mahalanobis distance augmented with similarity, entropy, and empirical support to detect unknown domains. Unlike typical OOD methods that simply reject or flag outliers, GRCLM uses OOD signals as a driver for creating new experts.

2.4 Self-Improving and Adaptive Systems

Self-improving systems, such as AlphaGo-style self-play [Silver et al., 2017] and self-training pipelines [He et al., 2020], allow models to generate or curate their own training data. More recently, LLM-based agents have been proposed that refine internal tools or policies over time [Schick et al., 2023, Shinn et al., 2023].

GRCLM shares the spirit of self-improvement but is distinctive in that it *modifies its own architecture* over time by spawning new domain experts and updating its domain topology based on geometric signals in embedding space.

2.5 Geometric Methods and Representation Topology

Centroid-based classifiers and nearest-class-mean methods are classical in pattern recognition [Hastie et al., 2009]. Subspace methods and principal angles are widely used to compare linear subspaces in signal processing and computer vision [Edelman et al., 1998]. Representation space geometry has been used to interpret neural networks [Raghu et al., 2017].

GRCLM incorporates these geometric ideas by assigning each domain a centroid and covariance structure and treating domains as manifolds in embedding space. The router uses both point-wise geometry (similarity and Mahalanobis distance) and subspace geometry (principal angles) to maintain a coherent domain topology.

3 Mathematical Foundations of GRCLM

3.1 Embedding Space and Domain Distributions

Let $\mathcal{X}$ denote the space of text inputs, and let $E : \mathcal{X} \to \mathbb{R}^d$ be an embedding model that maps each input $x \in \mathcal{X}$ to a vector $q = E(x) \in \mathbb{R}^d$. We assume E is fixed for routing purposes, although in practice it may share parameters with the transformer trunk.

Definition 1 (Domain)**.** A *domain k* is characterized by a probability distribution $\mathcal{P}_k$ over $\mathbb{R}^d$ (the embedding space) together with estimated moments: a centroid $c_k \in \mathbb{R}^d$ and a covariance matrix $\Sigma_k \in \mathbb{R}^{d \times d}$.

We interpret c_k as the mean embedding of the domain and Σ_k as capturing its local geometry. In practice, we maintain empirical estimates based on a corpus $\mathcal{C}_k$ of embeddings associated with domain k:

$$c_k = \frac{1}{|\mathcal{C}_k|} \sum_{q \in \mathcal{C}_k} q, \tag{1}$$

$$\Sigma_k = \frac{1}{|\mathcal{C}_k| - 1} \sum_{q \in \mathcal{C}_k} (q - c_k)(q - c_k)^\top + \lambda I_d, \tag{2}$$

where $\lambda > 0$ is a small regularization constant ensuring invertibility.

3.2 Routing Metrics

Given a query embedding $q \in \mathbb{R}^d$ and domain statistics (c_k, Σ_k), GRCLM computes several metrics.

Cosine similarity.

$$s_k(q) = \frac{q^\top c_k}{\|q\|\|c_k\|}. \tag{3}$$

This measures directional alignment between q and the domain centroid.

Mahalanobis distance.

$$m_k(q) = \sqrt{(q - c_k)^\top \Sigma_k^{-1} (q - c_k)}. \tag{4}$$

Assuming $\mathcal{P}_k$ is approximately Gaussian, $m_k(q)^2$ is proportional to the squared number of standard deviations from the mean in the domain's metric.

Predictive entropy. Let $p_{k,i}(q)$ denote the predictive probability for token i from specialist head k given context corresponding to q. We define:

$$H_k(q) = -\sum_i p_{k,i}(q) \log p_{k,i}(q), \tag{5}$$

which serves as a proxy for epistemic uncertainty: high entropy suggests the specialist is unsure about its prediction.

Support ratio. We also maintain an empirical support score $r_k \in [0, 1]$ for each domain, reflecting how often the router has successfully assigned queries to domain k in the recent past. One simple formulation is:

$$r_k = \frac{N_k}{\sum_j N_j}, \tag{6}$$

where N_k is the number of queries routed to domain k over a sliding window.

3.3 Composite Routing Score

GRCLM combines these metrics into a composite routing score.

Definition 2 (Composite routing score)**.** Given weights $\alpha, \beta, \gamma, \delta \geq 0$, the composite routing score for domain k and query embedding q is

$$R_k(q) = \alpha s_k(q) - \beta m_k(q) - \gamma H_k(q) + \delta r_k. \tag{7}$$

The chosen domain is

$$\hat{k}(q) = \arg\max_k R_k(q). \tag{8}$$

Intuitively, R_k increases with similarity and support, and decreases with distance and uncertainty. The following result formalizes monotonicity.

Proposition 1 (Monotonicity properties)**.** *Fix all quantities except one metric. Then for any domain k and query q:*

1. $\frac{\partial R_k(q)}{\partial s_k} = \alpha \geq 0$*, so R_k is non-decreasing in similarity.*
2. $\frac{\partial R_k(q)}{\partial m_k} = -\beta \leq 0$*, so R_k is non-increasing in Mahalanobis distance.*
3. $\frac{\partial R_k(q)}{\partial H_k} = -\gamma \leq 0$*, so R_k is non-increasing in entropy.*
4. $\frac{\partial R_k(q)}{\partial r_k} = \delta \geq 0$*, so R_k is non-decreasing in support.*

Proof. Immediate from the linear form of R_k. □

3.4 Unknown Domain Criteria

GRCLM must determine when a query does not belong to any known domain. We formalize this as follows.

Definition 3 (Unknown-domain condition)**.** Given thresholds $\tau_s, \tau_m, \tau_h, \tau_r$, a query embedding q is classified as *unknown-domain* if

$$\max_k s_k(q) < \tau_s \quad \vee \quad \min_k m_k(q) > \tau_m \quad \vee \quad \min_k H_k(q) > \tau_h \quad \vee \quad \max_k r_k < \tau_r. \tag{9}$$

Thus, a query is considered unknown if it is not sufficiently similar to any centroid, is too distant in Mahalanobis space, induces high uncertainty in all specialists, or no domain has adequate empirical support.

Remark 1. In practice, GRCLM may use a subset of these criteria or combine them into a scalar OOD score. The disjunctive form in (9) is conservative: it favors declaring a query unknown when any metric strongly suggests novelty.

3.5 Emergent Domain Clusters

Unknown-domain queries are not discarded; instead, GRCLM accumulates them into a buffer that may form the seed of a new domain.

Let $\mathcal{B}$ be the emergent buffer, storing embeddings $q_1, \ldots, q_n$. Define the empirical mean and variance:

$$\bar{q} = \frac{1}{n}\sum_{i=1}^{n} q_i, \tag{10}$$

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} \|q_i - \bar{q}\|^2. \tag{11}$$

Definition 4 (Emergent domain condition)**.** Given a minimum cluster size $N_{\min}$ and maximum variance $\sigma^2_{\max}$, GRCLM spawns a new domain when

$$n = |\mathcal{B}| \geq N_{\min} \quad \text{and} \quad \sigma^2 \leq \sigma^2_{\max}. \tag{12}$$

This encodes the intuition that a new domain should be supported by enough examples and that these examples should be coherent in embedding space.

Proposition 2 (Concentration under isotropic assumptions)**.** *Suppose unknown-domain embeddings are sampled i.i.d. from an isotropic Gaussian distribution $\mathcal{N}(\mu^\star, \sigma^2_\star I_d)$. Then for any $\epsilon > 0$, there exists $N_{\min}$ such that with probability at least $1 - \epsilon$, the empirical variance σ^2 satisfies $|\sigma^2 - d\sigma^2_\star| \leq \epsilon$ for all $n \geq N_{\min}$.*

Proof. This follows from standard concentration bounds for sample covariance of Gaussian distributions; see, e.g., Vershynin [2018]. □

While real embeddings are not Gaussian, this heuristic underlies the use of variance as a coherence measure for emergent domains.

4 Geometric Topology of Domains

So far, we have treated domains as point clusters with centroids and covariance matrices. We now describe the geometric relationships between domains.

4.1 Centroid Angles

For domains j and k, the angle between centroids is

$$\theta_{jk} = \arccos\left(\frac{c_j^\top c_k}{\|c_j\|\|c_k\|}\right). \tag{13}$$

Small angles indicate similar central directions in embedding space, suggesting higher overlap in semantic content.

4.2 Domain Subspaces and Principal Angles

Consider the eigendecomposition of each covariance matrix:

$$\Sigma_k = U_k \Lambda_k U_k^\top, \tag{14}$$

where U_k contains eigenvectors and Λ_k eigenvalues. The leading eigenvectors span a subspace capturing principal directions of variation.

Definition 5 (Domain subspace)**.** Let $U_k^{(r)}$ denote the matrix of the top r eigenvectors of Σ_k. The column space $\mathcal{S}_k = \operatorname{span}(U_k^{(r)})$ is the r-dimensional *domain subspace* of domain k.

To compare two subspaces $\mathcal{S}_j$ and $\mathcal{S}_k$, we compute their principal angles [Edelman et al., 1998]. Let $U_j^{(r)}$ and $U_k^{(r)}$ be orthonormal bases; compute the singular value decomposition

$$(U_j^{(r)})^\top U_k^{(r)} = P \Sigma Q^\top, \tag{15}$$

where $\Sigma = \operatorname{diag}(\sigma_1, \dots, \sigma_r)$ with $\sigma_\ell \in [0, 1]$. Then the principal angles $\phi_1, \dots, \phi_r$ are defined by

$$\phi_\ell = \arccos(\sigma_\ell). \tag{16}$$

Remark 2. Principal angles near zero indicate strong alignment between domains' principal directions, while angles near $\pi/2$ indicate near-orthogonality. GRCLM can use these angles to enforce topological coherence, e.g., by discouraging the creation of new domains too close to existing ones.

5 Shiftable Transformer Architecture

GRCLM employs a *shiftable transformer language model* (STLM) with a shared trunk and specialist heads.

Let T_θ denote the shared transformer trunk with parameters θ, and let H_{ϕ_k} denote the specialist head for domain k with parameters ϕ_k. Given an input sequence x mapped to token embeddings, we compute:

$$h = T_\theta(x), \tag{17}$$

$$y = H_{\phi_k}(h), \tag{18}$$

where y represents the logits or probabilities over the vocabulary.

5.1 Parameter Sharing and Specialization

The trunk T_θ captures general linguistic and reasoning capabilities and is shared across all domains. Each specialist H_{ϕ_k} focuses on fine-grained adjustments that capture domain-specific lexical or structural patterns.

Definition 6 (Shiftable model)**.** A model is *shiftable* if it can switch the active specialist head H_{ϕ_k} at inference time based on a routing decision, without modifying the underlying trunk T_θ.

GRCLM is shiftable by design: the geometric router selects domain $\hat{k}$ and activates $H_{\phi_{\hat{k}}}$ for generation.

6 Continual Learning in GRCLM

6.1 Training Objective

Let $\mathcal{D}$ denote the union of all domain corpora and the general corpus. Each training example is a triple (x, y, d), where x is input, y is target output, and d is the domain label (possibly "general").

The primary loss is next-token prediction loss:

$$\mathcal{L}_{\text{base}}(\theta, \{\phi_k\}) = \mathbb{E}_{(x,y,d)\sim\mathcal{D}}[\ell(H_{\phi_d}(T_\theta(x)), y)]. \tag{19}$$

To mitigate catastrophic forgetting, we may add regularization terms. For example, an EWC-style penalty [Kirkpatrick et al., 2017]:

$$\mathcal{L}_{\text{reg}}(\theta) = \sum_i \frac{\lambda_{\text{ewc}}}{2} F_i(\theta_i - \theta_i^{\text{old}})^2, \tag{20}$$

where F_i are Fisher information coefficients and θ^{old} are previous trunk parameters.

The total objective becomes

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \mathcal{L}_{\text{reg}}. \tag{21}$$

6.2 Update Schedule

When a new domain $k^\star$ emerges, GRCLM triggers a continual learning cycle:

1. Initialize a new specialist head $H_{\phi_{k^\star}}$ (randomly or from a template).
2. Construct a training set including:
   - Emergent corpus for $k^\star$.
   - Subsampled data from existing domains.
   - General corpus (optional).
3. Optimize $\mathcal{L}$ with respect to θ and $\phi_{k^\star}$ (and optionally refine ϕ_k for existing domains).
4. Recompute centroids and covariances for all domains.
5. Update router thresholds and support ratios.

Algorithm 1 GRCLM Routing and Emergence

```
Initialize domains, router statistics, trunk $T_\theta$, and heads $H_{\phi_k}$.
Initialize emergence buffer $\mathcal{B} \leftarrow \emptyset$.
while system is running do
    Receive query $x$ and compute embedding $q \leftarrow E(x)$.
    Compute routing metrics $s_k(q)$, $m_k(q)$, $H_k(q)$, $r_k$ for all domains $k$.
    if unknown-domain condition (9) holds then
        Append $(x, q)$ to buffer $\mathcal{B}$.
        if $|\mathcal{B}| \geq N_{\min}$ and variance $\sigma^2 \leq \sigma^2_{\max}$ then
            Spawn new domain $k^\star$ with corpus from $\mathcal{B}$.
            Initialize specialist head $H_{\phi_{k^\star}}$.
            Run continual learning update on $T_\theta$ and $\phi_{k^\star}$.
            Recompute centroids and covariances for all domains.
            Reset $\mathcal{B} \leftarrow \emptyset$.
        end if
    else
        Route to $\hat{k} = \arg\max_k R_k(q)$ and generate response using $H_{\phi_{\hat{k}}}$.
        Update support ratio $r_{\hat{k}}$.
    end if
end while
```

6.3 Stability Considerations

Updating the shared trunk risks degrading performance on previously learned domains. GRCLM can modulate this trade-off via:

- Lower learning rates for θ than for $\phi_{k^\star}$.
- Partial freezing of trunk layers.
- Stronger regularization penalties for deviations from θ^{old}.

7 Algorithmic Workflow

Algorithm 1 summarizes the GRCLM routing and emergence loop.

8 Experimental Design

We briefly outline experiments suitable for evaluating GRCLM.

8.1 Synthetic Multi-Domain Benchmark

Construct a synthetic benchmark with several distinct domains (e.g., programming, medical, legal, fiction), each with its own training corpus. Train GRCLM on a subset of domains, then introduce queries from a novel domain and allow the system to create a new specialist.

Metrics:

- Routing accuracy before and after emergence.
- Specialist perplexity on its domain.
- Impact on existing domains' performance.

8.2 Domain Drift

Simulate domain drift by gradually changing the distribution of queries within a domain. Measure:

- Centroid motion and covariance change over time.
- Router's ability to maintain correct routing.
- Whether GRCLM decides to split or merge domains.

8.3 Ablation Studies

Evaluate GRCLM variants:

- Without entropy term ($\gamma = 0$).
- Without support ratio term ($\delta = 0$).
- With frozen trunk versus adaptive trunk.

9 Discussion

GRCLM proposes a shift from static LLM deployment to self-organizing expert architectures. By rooting domain routing and emergence in explicit geometric quantities, GRCLM offers transparency and controllability that are difficult to obtain with purely learned gates. At the same time, the architecture inherits challenges from both continual learning and sparse expert systems, including complexity of orchestration, sensitivity to hyperparameters, and the need for robust monitoring in deployment.

10 Future Work

Future directions include:

- Hierarchical geometric routing with multi-level domain trees.
- Distributed GRCLM nodes exchanging domain summaries in a cognitive mesh.
- Parameter-efficient specialists using LoRA or adapters to scale to hundreds or thousands of domains.
- Incorporating human feedback for domain naming, merging, or pruning.

11 Conclusion

We have presented GRCLM, a Geometric Routing & Continual Learning Model for self-organizing language model specialization. GRCLM unifies geometric domain routing, emergent domain creation, and continual learning in a shiftable transformer architecture. This work lays the mathematical and architectural foundation for LLMs that can grow, adapt, and restructure themselves in response to open-world demands.

References

Tom Brown et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 2020.

Nan Du et al. Glam: Efficient scaling of language models with mixture-of-experts. *International Conference on Machine Learning*, 2022.

Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 1998.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 2022.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2009.

Xiaojin He et al. Revisiting self-training for semi-supervised learning. In *International Conference on Learning Representations*, 2020.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017.

Neil Houlsby et al. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, 2019.

Edward J Hu et al. Lora: Low-rank adaptation of large language models. *International Conference on Learning Representations*, 2022.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural Computation*, 1991.

James Kirkpatrick et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 2017.

Kimin Lee, Kibok Lee, et al. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, 2018.

Dmitry Lepikhin et al. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

Mike Lewis et al. Base layers: Simplifying training of large, sparse models. *International Conference on Machine Learning*, 2021.

Weitang Liu et al. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2020.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, 2017.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Maithra Raghu et al. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, 2017.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, et al. icarl: Incremental classifier and representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Andrei A Rusu et al. Progressive neural networks. In *arXiv preprint arXiv:1606.04671*, 2016.

Timo Schick et al. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*, 2023.

Noah Shinn et al. Reflexion: An autonomous agent with dynamic memory and self-reflection. *arXiv preprint arXiv:2303.11366*, 2023.

David Silver et al. Mastering the game of go without human knowledge. *Nature*, 2017.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*. Cambridge University Press, 2018.