Dynamics of Concept Learning and Compositional Generalization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Prior work has shown that text-conditioned diffusion models can learn to identify and manipulate primitive concepts underlying a compositional data-generating process, enabling generalization to entirely novel, out-of-distribution compositions. Beyond performance evaluations, these studies develop a rich empirical phenomenology of learning dynamics, showing that models generalize sequentially, respecting the compositional hierarchy of the data-generating process. Moreover, concept-centric structures within the data significantly influence a model's speed of learning the ability to manipulate a concept. In this paper, we aim to better characterize these empirical results from a theoretical standpoint. Specifically, we propose an abstraction of prior work's compositional generalization problem by introducing a structured identity mapping (SIM) task, where a model is trained to learn the identity mapping on a Gaussian mixture with structurally organized centroids. We mathematically analyze the learning dynamics of neural networks trained on this SIM task and show that, despite its simplicity, SIM's learning dynamics capture and help explain key empirical observations on compositional generalization with diffusion models identified in prior work. Our theory also offers several new insights—e.g., we find a novel mechanism for non-monotonic learning dynamics of test loss in early phases of training. We validate our new predictions by training a text-conditioned diffusion model, bridging our simplified framework and complex generative models. Overall, this work establishes the SIM task as a meaningful theoretical abstraction of concept learning dynamics in modern generative models.

1 Introduction

Human cognitive abilities have been argued to generalize to unseen scenarios through the identification and systematic composition of primitive concepts that constitute the natural world (e.g., shape, size, color) [18, 19, 61, 20, 64, 21, 24]. Motivated by this perspective, the ability to compositionally generalize to entirely unseen, out-of-distribution problems has been deemed a desirable property for machine learning systems, leading to decades of research on the topic [68, 37, 28, 58, 58, 62, 38, 33, 14].

Recent work has shown that modern neural network training pipelines can lead to emergent abilities that allow a model to compositionally generalize when it is trained on a data-generating process that itself is compositional in nature [38, 59, 52, 41, 5, 78, 34]. For example, [52, 54] show that text-conditioned diffusion models can learn to identify concepts that constitute the training data and develop abilities to manipulate these concepts flexibly, enabling generations that represent novel compositions entirely unseen during training. These papers also provide a spectrum of intriguing empirical results regarding a model's learning dynamics in a compositional task. For example, they reveal that abilities to manipulate individual concepts are learned in a sequential order dictated by the

Submitted to the Mathematics of Modern Machine Learning Workshop at NeurIPS 2024. Do not distribute.

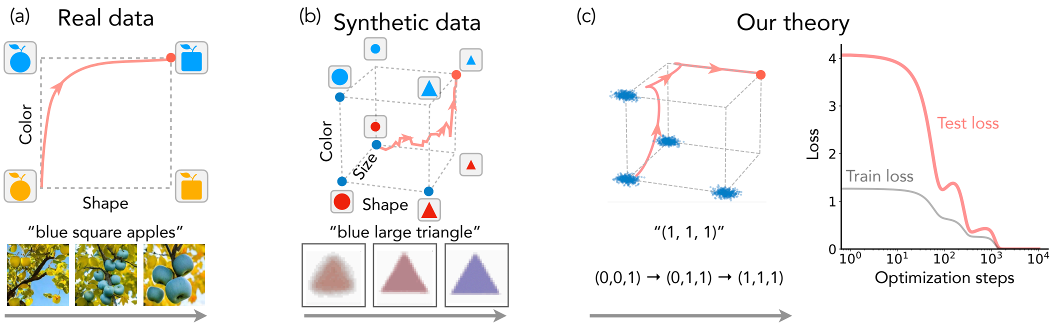

Figure 1: **Structured Identity Mapping Task and Non-Monotonic Generalization Dynamics.** (a) Given the input "blue square apples on a tree with circular yellow leaves," a multimodal model learns to generate concepts in the following order: "apple," "blue" (color), and "square" (shape) (example adapted from [43]). (b) A multimodal synthetic task introduced by [52, 54]. The training set of the task consists of four distinct compositions of concepts, depicted as blue nodes on a cubic graph. A diffusion model is trained on this dataset to systematically study the dynamics of concept learning. With the test prompt "small, blue, triangle," the diffusion model sequentially learns the correct size, shape, and finally color. (c) In this work, we introduce a structured identity mapping task as the foundation for a systematical and theoretical studying of the dynamics of concept learning. The model is trained on a Gaussian mixture data, where the centroids are positioned at certain nodes of a hypercube (blue dots) and is evaluated on an out-of-distribution test set (red dot). Our theoretical results not only reproduce and explain previously characterized empirical phenomena but also depict a comprehensive picture of the non-monotonic learning dynamics in the concept space and predict a "multiple-descent" curve of the test loss (red curve).

data-generating process; the speed of learning such abilities is modulated by data-centric measures (e.g., gradient of loss with respect to concept values, such as color of an object); and the most similar composition seen during training often controls performance on unseen compositions.

In this work, we aim to demystify the phenomenology of compositional generalization identified in prior work and better ground the problem (or at least a specific variant of it called systematicity) via a precise theoretical analysis. To that end, we instantiate a simplified version of the compositional generalization framework introduced by [52, 54]—called the "concept space" (see Fig. 1b)—that is amenable to theoretical analysis. In brief, a concept space is a vector space that serves as an abstraction of real concepts. For each concept (e.g., color), a binary number can be used to represent its value (e.g., 0 for red and 1 for blue). In this way, a binary string can be mapped to a tuple (e.g., $(1, 0, 1)$ might represent "big blue triangle") and then fed into the diffusion model as a conditioning vector. The model output is then passed through a classifier[1] which produces a vector indicating how accurately the corresponding concepts are generated (e.g. a generated image of big blue triangle might be classified as $(0.8, 0.1, 0.9)$). In this way, the process of generation becomes a vector mapping, and *a good generator essentially performs as an identity mapping in the concept space*.

We argue that in fact the salient characteristic of a concept space is its preemptively defined organization of concepts in a systematic manner, not the precise concepts used for instantiating the framework itself. Grounded in this argument, we define a learning problem called the *Structured Identity Mapping (SIM) task* wherein a regression model is trained to learn the identity mapping from points sampled from a mixture of Gaussians with structurally organized centroids (see Fig. 1c). Through a detailed analysis of the learning dynamics of MLP models, both empirically and theoretically, we find that SIM, despite its simplicity, can both capture the phenomenology identified by prior work and provide precise explanations for it. Our theoretical findings also lead to novel insights, e.g., predicting the existence of a novel mechanism for non-monotonic learning curves (similar to epochwise double-descent [51], but for *out-of-distribution data*) in the early phase of training, which we empirically verify to be true by training a text-conditioned diffusion model. Our contributions are summarized below.

- **Structured Identity Mapping (SIM): A faithful abstraction of concept space.** We empirically validate our SIM task by training Multi-Layer Perceptrons (MLPs), demonstrating the reproduction of key compositional generalization phenomena characterized in recent diffusion model

[1] Conceptually, we can think of an idealized perfect classifier here.

studies [52, 54]. Our findings show: (i) learning dynamics of OOD test loss respect the compositional hierarchical structure of the data generating process; (ii) the speed at which a model disentangles a concept and learns the capability to manipulate it is dictated by the sensitivity of the data-generating process to changes in values of said concept (called "concept signal" in prior work); and (iii) network outputs corresponding to weak concept signals exhibit slowing down in concept space. These results also suggest that the structured nature of the data, rather than specific concepts, drove observations reported in prior work.

- **Theoretical analysis reveals mechanisms underlying learning dynamics of a compositional task.** Building on the successful reproduction of phenomenology with MLPs trained on the SIM task, we further simplify the architecture to enable theoretical analysis. We demonstrate that: (i) analytical solutions with a linear regression model reproduce the observed phenomenology above, and (ii) the analysis of a symmetric 2-layer network ($f(\boldsymbol{x}; \boldsymbol{U}) = \boldsymbol{U}\boldsymbol{U}^\top \boldsymbol{x}$) identifies a novel mechanism of non-monotonic learning dynamics in generalization loss, which we term **Transient Memorization**. Strikingly, we show that the learning dynamics of compositional generalization loss can exhibit *multiple descents* in its early phase of learning, corresponding to multiple phase transitions in the learning process.
- **Empirical confirmation of the predicted Transient Memorization phenomenon in diffusion models.** We verify the predicted mechanism of Transient Memorization in text-conditioned diffusion models, observing the non-monotonic evolution of generalization accuracy for unseen combinations of concepts, as predicted by our theory.

In summary, our theoretical analysis of networks trained on SIM tasks provides mechanistic explanations for previously observed phenomenology in empirical works and introduces the novel concept of Transient Memorization. This mechanism is subsequently confirmed in text-conditioned diffusion models, bridging theory and practice in compositional generalization dynamics.

2 Preliminaries and Problem Setting

Throughout the paper, we use bold lowercase letters (e.g., $\boldsymbol{x}$) to represent vectors, and use bold uppercase letters (e.g., $\boldsymbol{A}$) to represent matrices. We use the unbold and lowercase version of corresponding letters with subscripts to represent corresponding entries of the vectors or matrices, e.g., x_i represent the i-th entry of $\boldsymbol{x}$ and $a_{i,j}$ represent the (i, j)-th entry of $\boldsymbol{A}$. For a vector $\boldsymbol{x}$ and a natural number k, we use $\boldsymbol{x}_{:k}$ to represent the k-dimensional vector that contains the first k entries of $\boldsymbol{x}$. For a natural number k, we use $[k]$ to represent the set $\{1, 2, \ldots, k\}$, and $\mathbf{1}_k$ to represent a vector whose entries are all 0 except the k-th entry being 1; the dimensionality of this vector is determined by the context if not specified. In the theory part, we frequently consider functions of time, denoted by variable t. If a function $g(t)$ is a function of time t, we denote the derivative of g with respect to t by $\dot{g}(t_0) = \left.\frac{\mathrm{d}g}{\mathrm{d}t}\right|_{t=t_0}$. Moreover, we sometimes omit the argument t, i.e., g means $g(t)$ for a time t determined by the context. For a statement ϕ, we define $\mathbb{1}_{\{\phi\}} = \begin{cases} 1 & \phi \text{ is true} \\ 0 & \phi \text{ is false} \end{cases}$ to be the indicator function of that statement.

2.1 Problem Setting

Now we formally define SIM, which is an abstraction of the concept space. For each concept class, we model them as a Gaussian cluster in the Euclidean space, placed along a unique coordinate direction. The distance between the cluster mean and the origin represents the strength of the concept signal, and the covariance of the Gaussian cluster represents the data diversity within the corresponding concept class. Additionally, we allow more coordinate directions than the clusters, meaning that some coordinate directions will not be occupied by a cluster, which we call non-informative directions, and they correspond to the free variables in the generalization task. See Fig. 1c for an illustration of the dataset of SIM.

Training Set. Let $d \in \mathbb{N}$ be the dimensionality of the input space and $s \in [d]$ be the number of concept classes, i.e., there are s Gaussian clusters, and $n \in \mathbb{N}$ number of samples from each cluster. The training set $\mathcal{D} = \bigcup_{p \in [s]} \left\{\boldsymbol{x}_k^{(p)}\right\}_{k=1}^n$ is generated by the following process: for each $p \in [s]$, each training point of the p-th cluster is sampled i.i.d. from a Gaussian distribution $\boldsymbol{x}_k^{(p)} \sim \mathcal{N}\left[\mu_p \mathbf{1}_p, \operatorname{diag}(\boldsymbol{\sigma})^2\right]$, where $\mu_p \geq 0$ is the distance of the p-th cluster center from the origin, and

$\boldsymbol{\sigma}$ is a vector with only the first s entries being non-zero, and σ_i^2 describing the data variance on the i-th direction. There is also optionally a cluster centered at $\mathbf{0}$.

Loss function. The training problem is to learn identity mapping on $\mathbb{R}^d$. For a model $f : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^d$ and a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^m$, we train the model parameters $\boldsymbol{\theta}$ via the mean square error loss.

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2sn} \sum_{p=1}^{s} \sum_{k=1}^{n} \left\| f\left(\boldsymbol{\theta}; \boldsymbol{x}_k^{(s)}\right) - \boldsymbol{x}_k^{(s)} \right\|^2 . \tag{2.1}$$

Evaluation. We evaluate the model at a Gaussian cluster centered at the point that combines the cluster means of all training clusters. When the variance of the test set is small, the expected loss within the test cluster is approximately equivalent to the loss at its cluster mean. Therefore, for simplicity, in this paper, we focus on the loss at the mean of the test cluster, which is a single test point $\widehat{\boldsymbol{x}} = \sum_{p=1}^{s} \mu_p \mathbf{1}_p$. We emphasize this point is outside of the training distribution—not just the training data, necessitating out-of-distribution generalization. In App. B, we report further results for the case of various combinations of training clusters, which leads to multiple OOD test points.

3 Observations on the SIM Task

We first begin by summarizing our key empirical findings on the SIM task. In all experiments we use MLP models of various configurations, including different number of layers and both linear and non-linear (specifically, ReLU) activations. Throughout this section and the subsequent sections, we frequently consider the model output at the test point $\widehat{\boldsymbol{x}}$ over training time, which we call **output trajectory** of the model.

Due to space constraints, we only present the results for a subset of configurations in the main paper and defer other results to App. F. We note that the findings reported in this section are in one-to-one correspondence with results identified using diffusion models in Sec. 5 and prior work [54].

3.1 Generalization Order Controlled by Signal Strength and Diversity

One interesting finding from previous work is that if we alter the strength of one concept signal from small to large, the contour of the learning dynamics would dramatically change [54]. Moreover, it is also commonly hypothesised that with more diverse data, the model generalizes better [23, 10]. Recall that in the SIM task, the distance μ_k of each cluster represents the corresponding signal strength, and the variance σ_k represents the data diversity. In Fig. 2, we present the output trajectory under the setting of $s = 2$, in which case the trajectory can be visualized in a plane. There are two components to be learned in this task and, from the contour of the curve, we can tell the order of different components being learned.

Fig. 2 (a) presents the output trajectory for a setting with a fixed and balanced $\boldsymbol{\sigma}$, and a varied $\boldsymbol{\mu}$. The results show that when $\mu_1 < \mu_2$, the dynamics exhibit an upward bulging, indicating a preference for the direction of stronger signal. As μ_1 is gradually increased, this contour shifts from an upward bulging to a downward concaving, and consistently maintains the stronger signal preference.

In Fig. 2 (b), the $\boldsymbol{\mu}$ is fixed to an unbalanced position, with one signal stronger than the other. As we mentioned above, when $\boldsymbol{\sigma}$ is balanced, the model will first move towards the cluster with a stronger signal strength. However, when the level of diversity of the cluster with weaker signal is gradually increased, the preference of the model shifts from one cluster to another.

A very concrete conclusion can be thus drawn from the results in Fig. 2 (a) and (b): the generalization order is jointly controlled by the signal strength and data diversity, and, generally speaking, the model prefers direction that has a stronger signal and more diverse data. We note that the conclusion here is more qualitative and in Sec. 4, we provide a more precise quantitative characterization of how these two values control the generalization order.

3.2 Convergence Rate Slow Down In Terminal Phase

In Fig. 2, the arrow-like markers on the line indicate equal training time intervals. In the later phase of training, we observe that the arrows get denser, indicating a slowing down of the learning dynamics. A close examination of the markers in Fig. 2 suggests that the deceleration is not determined by the distance of the current output to the target point (i.e. the loss value), but more depends on the data and training time. That is, there is a timescale determined by the training data such that if the model does not achieve OOD generalization within that period, significantly more computation

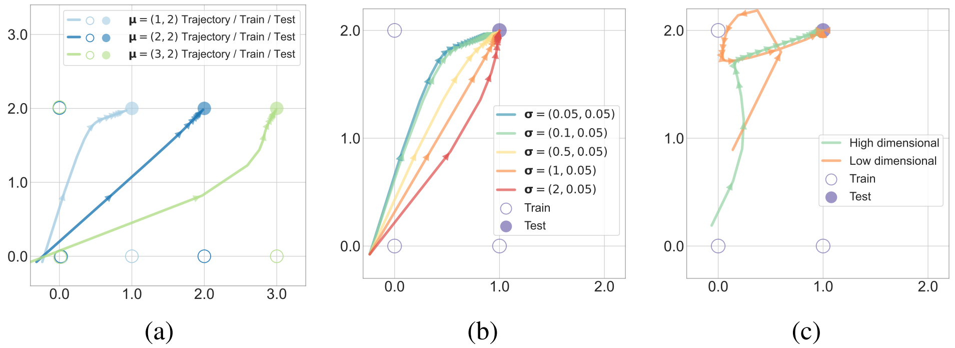

Figure 2: **Learning dynamics of MLP on SIM task.** The figures show the output trajectory of the MLP on a two-dimensional setting (i.e., $s = 2$), and each marker represents an optimization timepoint. Notice that we only plot the center of the training set as a circle, but the actual training set can have varied shapes based on the configuration of $\boldsymbol{\sigma}$. (a) one-layer linear model with $\boldsymbol{\sigma}_{:2} = (.05, .05)$ and varied $\boldsymbol{\mu}$. Concepts i with larger signal (μ_i) learnt first. (b) one-layer linear model with $\boldsymbol{\mu}_{:2} = (1, 2)$ and varied $\boldsymbol{\sigma}$. Concepts i with larger diversity (σ_i) learnt first. (c) 4 layer linear models under different dimensionality. high dim: $d = 64$, low dim: $d = 2$. Notice that (a) and (b) are both in high dim setting. The lower the dimensionality, the stronger Transient Memorization it has.

will be required for the model to achieve it. This effect can be observed in Fig. 2 (b) by comparing the output trajectories of $\boldsymbol{\sigma} = (0.05, 0.05)$ case and the $\boldsymbol{\sigma} = (2, 0.05)$ case. The trajectory with the lower concept signal (i.e. $\boldsymbol{\sigma} = (0.05, 0.05)$) yields insufficient OOD generalization until the dynamics slows down and thus requires many more time to approach the target point.

3.3 Transient Memorization

The results in Fig. 2 (a) and (b) are both performed with one-layer models and under a high dimensional setting ($d = 64$). Despite the overall trend being similar in other settings, it is worth exploring the change of trajectory as we increase the number of layers, and / or reduce the dimension.

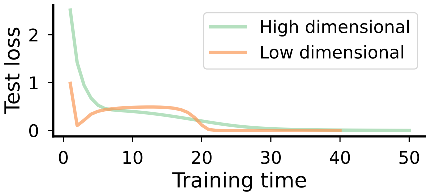

Figure 3: The test loss of multi-layer models.

In Fig. 2 (c), we perform experiments with deeper models, and optionally with a lower dimension. Under these changes, we find that the model shows an interesting irregular behavior, where it initially heads towards the OOD test point, but soon turns toward the training set cluster with the strongest signal. This indicates the model, while seems to be generalizing OOD at the beginning, is memorizing the train distribution and unable to generalize OOD at this point. However, with enough training, we find the model start to again move towards the test point and thus generalizes OOD. We call this overall dynamic of the output trajectory **Transient Memorization**, which we could be suggestive of a non-monotonic test loss curve (similar to epoch-wise double-descent [51, 53], but with an OOD test loss). To assess this further, we track the value of the loss function during training in Fig. 3, demonstrating a double-descent-like curve. We emphasize though that Transient Memorization is a distributional phenomenon and is different from what was called (epochwise) double-descent, i.e., the model memorizes the training *distribution* and is hence unable to generalize well OOD, while double-descent involves classical overfitting to the training *data* itself, affecting model's in-distribution generalization. We also note that the *Transient Memorization* phenomena seems to be strongest when dimensionality d of the dataset is low, and is rather modest with high dimensional settings. In the high dimensional setting, the OOD loss descent slows down at some point but does not actually exhibit non-monotonic behavior. This low dimensional preference can also be explained perfectly by our theory, further described in Sec. 4.

4 Theoretical Explanation

We next study the training dynamics of a specific class of linear models that are tractable on the SIM task and explain the empirical phenomenology of OOD learning dynamics seen in previous section. In Sec. 4.1, we first analyze a one-layer model whose dynamics can be solved analytically. We show that it can explain most phenomena observed in the experiment; however, it fails to reproduce Transient Memorization, suggesting that Transient Memorization is intrinsic to deep models, which highlights the fundamental difference between shallow and deep models. In Sec. 4.2, we further analyze the dynamics of a symmetric 2-layer linear model, which successfully captures Transient Memorization. Our theoretical results reveal a multi-stage behavior of the model Jacobian during training, which leads to the non-monotonic behavior in model output. We show that each stage in the Transient Memorization precisely corresponds to each stage in the Jacobian evolution.

Throughout this section, we assume $\boldsymbol{f}(\boldsymbol{\theta};\boldsymbol{x})$ is a linear function of $\boldsymbol{x}$. In this case the Jacobian of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$ is a matrix that is completely determined by $\boldsymbol{\theta}$, which we denote by $\boldsymbol{W}_{\boldsymbol{\theta}} = \frac{\partial f(\boldsymbol{\theta};\boldsymbol{x})}{\partial \boldsymbol{x}}$. In this way, the output of the model can be written as $\boldsymbol{f}(\boldsymbol{\theta};\boldsymbol{x}) = \boldsymbol{W}_{\boldsymbol{\theta}}\boldsymbol{x}$. Using the trace trick (with detailed calculations provided in App. C.1), it is easy to show that the overall loss function is equal to

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\left\|\left(\boldsymbol{W}_{\boldsymbol{\theta}} - \boldsymbol{I}\right)\boldsymbol{A}^{1/2}\right\|_{\mathcal{F}}^{2}, \tag{4.1}$$

where $\boldsymbol{A} = \frac{1}{sn}\sum_{p=1}^{s}\sum_{k=1}^{n}\boldsymbol{x}_k^{(p)}\boldsymbol{x}_k^{(p)\top}$ is the empirical covariance. In this section, we assume n is large, in which case $\boldsymbol{A}$ converges to the true covariance of the dataset $\boldsymbol{A} = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[\boldsymbol{x}\boldsymbol{x}^\top]$, which is a diagonal matrix $\boldsymbol{A} = \operatorname{diag}(\boldsymbol{a})$, defined by $a_p = \begin{cases}\sigma_p^2 + \frac{\mu_p^2}{s} & p \leq s\\ 0 & p > s\end{cases}$, for any $p \in [d]$.

Remark. Notice that in the linear setting we might not directly train $\boldsymbol{W}_{\boldsymbol{\theta}}$; instead, we train its components. For example, we might have $\boldsymbol{\theta} = (\boldsymbol{W}_1, \boldsymbol{W}_2)$ and have $\boldsymbol{W}_{\boldsymbol{\theta}} = \boldsymbol{W}_1\boldsymbol{W}_2$. Then, what we actually train is $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, instead of $\boldsymbol{W}_{\boldsymbol{\theta}}$. As many previous works have emphasized [3, 30, 4, 1], although the deep linear model has the same capacity as a one-layer linear model, their dynamics can be vastly different and the loss landscape of deep linear models can be non-convex.

4.1 A One-Layer Model Theory and Its Limitations

As a warm-up, we first study the dynamics of one-layer linear models, i.e., $f(\boldsymbol{W};\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x}$, in which case the Jacobian $\boldsymbol{W}_{\boldsymbol{\theta}}$ is simply $\boldsymbol{W}$. As we will show, this setting can already explain most of the observed phenomenology from the previous section including the order of generalization and the terminal phase slowing down, but fails to capture the Transient Memorization, which we will explore in next subsection. Here we present Theorem 4.1, which gives the analytical solution of the one-layer model on the SIM task.

Theorem 4.1. *Let $\boldsymbol{W}(t) \in \mathbb{R}^{d\times d}$ be initialized as $\boldsymbol{W}(0) = \boldsymbol{W}^{(0)}$, and updated by $\dot{\boldsymbol{W}} = -\nabla\mathcal{L}(W)$, with $\mathcal{L}$ be defined by eq.* (4.1) *with $f(\boldsymbol{W},\boldsymbol{z}) = \boldsymbol{W}\boldsymbol{z}$, then we have for any $\boldsymbol{z} \in \mathbb{R}^d$,*

$$f(\boldsymbol{W}(t),\boldsymbol{z})_k = \underbrace{\mathbb{1}_{\{k\leq s\}}\left[1 - \exp\left(-a_k t\right)\right] z_k}_{\widetilde{G}_k(t)} + \underbrace{\sum_{i=1}^{s}\exp\left(-a_i t\right) w_{k,i}(0) z_i}_{\widetilde{N}_k(t)}. \tag{4.2}$$

See App. C.2 for proof of Theorem 4.1. The Theorem shows that the k-th dimension of the output of a one-layer model evaluated on the test point $\widehat{\boldsymbol{x}}$ can be decomposed into two terms: the *growth term* $\widetilde{G}_k(t) = \mathbb{1}_{\{k\leq s\}}\left[1 - \exp\left(-a_k t\right)\right]\mu_k$, and the *noise term* $\widetilde{N}_k(t) = \sum_{i=1}^{s}\exp\left(-a_i t\right) w_{k,i}(0)\mu_i$. The following properties can be observed for these two terms: (i) the growth term converges to μ_k when $k \leq s$ and 0 when $k > s$, while the noise term converges to 0; (ii) both terms converge at an exponential rate; and (iii) the noise term is upper bounded by $\sum_{i=1}^{s} w_{k,i}(0)\mu_i$. If the model initialization is small in scale, specifically $w_{k,i}(0) \ll \frac{1}{s\max_{i\in[s]}\mu_i}$, then $\widetilde{N}_k(t)$ will always be small, and thus can be omitted. With this assumption in effect, the model output is dominated by the growth term. A closer look at the growth term then explains part of the observed phenomenology.

Generalization Order and Terminal Phase Slowing Down. It can be observed that $\widetilde{G}_k(t)$ converges at an exponential rate, which leads an exponential decay of evolution speed and *explains the terminal phase slowing down*. Moreover, the exponential convergence rate of $\widetilde{G}_k(t)$ is controlled by the coefficient $a_k = \frac{1}{s}\left(s\sigma_k^2 + \mu_k^2\right)$. Therefore, the direction with larger a_k, i.e., larger μ_k and / or σ_k, converges faster, *hence explaining the order of generalization to different concepts*. The theorem also reveals the proportional relationship between μ_k (concept signal strength) and σ_k (data diversity).

The Limitation of the One Layer Model Theory. While we have demonstrated that Theorem 4.1 effectively explains both the generalization order and the terminal phase slowing down, in the solution eq. (4.2), the learning of each direction is independent. This independence omits the possible interaction between the dynamics of different directions in deeper models, and leads to monotonic and rather regular output trajectory (this is verified by the experiment results in Sec. 3.1). However, as the experiments in Sec. 3.3 show, when the number of layers becomes larger, the model actually exhibits a non-monotonic trace that can have detours. The theory based on the one-layer model fails in capturing this behavior. In the subsequent subsection, we introduce a more comprehensive theory based on a deeper model, and demonstrate that this model explains all the phenomena observed in Sec. 3, especially the Transient Memorization.

4.2 A Symmetric Two-Layer Linear Model Theory

In this subsection, we analyze a symmetric 2-layer linear model, namely $f(\boldsymbol{U};\boldsymbol{x}) = \boldsymbol{U}\boldsymbol{U}^\top\boldsymbol{x}$, where $\boldsymbol{U} \in \mathbb{R}^{d\times d'}$ and $d' \geq d$. We demonstrate that it accurately captures all the observations presented in Sec. 3, and, more importantly, the theory derived from this model provides a comprehensive understanding of the evolution of the model Jacobian and output, offering a clear and intuitive explanation for the underlying mechanism of the model's seemingly irregular behaviors. Due to space constraints, we focus on providing an intuitive explanation of the multi-stage behavior of the model Jacobian and output, and defer the formal proofs to the appendix. It is also worth noting that this symmetric 2-layer linear model is a frequently studied model in theoretical analysis [44, 70, 31], and most existing theoretical results for this model focus on the implicit bias of the solution found, instead of on the non-monotonic behavior during training, which is the focus of our analysis.

For convenience, we denote the Jacobian of f at time point t by $\boldsymbol{W}(t) = \boldsymbol{W}_{\boldsymbol{U}(t)}$. The gradient flow update of the i,j-th entry of $\boldsymbol{W}$ is given by

$$\dot{w}_{i,j} = \underbrace{w_{i,j}(a_i + a_j)}_{G_{i,j}(t)} - \underbrace{\frac{1}{2}w_{i,j}\left[w_{i,i}(3a_i + a_j) + \mathbb{1}_{\{i\neq j\}}w_{j,j}(3a_j + a_i)\right]}_{S_{i,j}(t)} - \underbrace{\frac{1}{2}\sum_{\substack{k\neq i\\k\neq j}} w_{k,i}w_{k,j}(a_i + a_j + 2a_k)}_{N_{i,j}(t)}. \tag{4.3}$$

As noted in eq. (4.3), we decompose the update of $w_{i,j}$ into three terms. We call $G_{i,j}(t) = w_{i,j}(t)(a_i + a_j)$ the *growth term*, $S_{i,j}(t) = \frac{1}{2}w_{i,j}\left[w_{i,i}(3a_i + a_j) + \mathbb{1}_{\{i\neq j\}}w_{j,j}(3a_j + a_i)\right]$ the *suppression term*, and $N_{i,j}(t) = \frac{1}{2}\sum_{\substack{k\neq i\\k\neq j}} w_{k,i}(t)w_{k,j}(t)(a_i + a_j + 2a_k)$ the *noise term*. The name of these terms suggests their role in the evolution of the Jacobian: the growth term $G_{i,j}$ always has the same sign as $w_{i,j}$, and has a positive contribution to the update, so it always leads to the direction that **increases the absolute value** of $w_{i,j}$; the suppression term $S_{i,j}$ also has the same sign[2] as $w_{i,j}$, but has a negative contribution in the update of $w_{i,j}$, so it always leads to the direction that **decreases the absolute value** of $w_{i,j}$; and the effect direction of the noise term is rather arbitrary since it depends on the sign of $w_{i,j}$ and other terms. It is proved in Lemma D.8 that under mild assumptions, the noise term will never be too large; for brevity, we omit it in the following discussion and defer the formal treatment of it to the rigorous proofs in App. D.

4.2.1 The Evolution of Entries of Jacobian

In order to better present the evolution of the Jacobian, we divide the entries of the Jacobian into three types: the **major entries** are the first s diagonal entries, and the **minor entries** are the off-diagonal entries who are in the first s rows or first s columns, and other entries are **irrelevant**

[2] Notice that since $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{U}^\top$ is a PSD matrix, the diagonal entries are always non-negative.

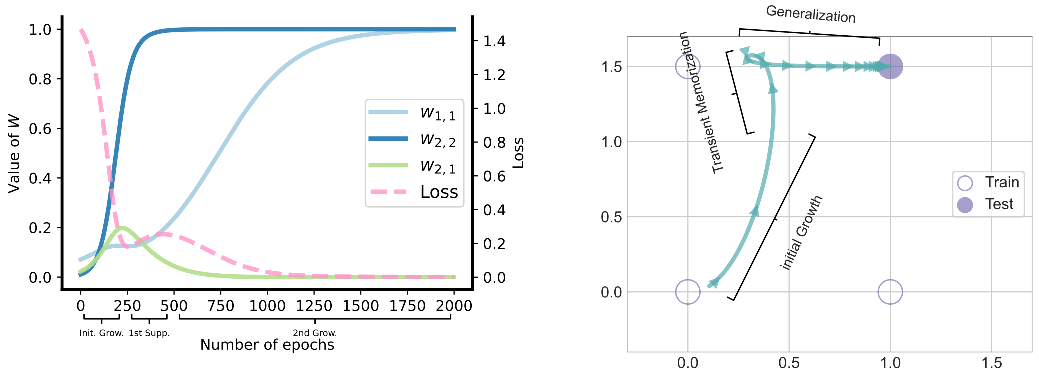

Figure 5: **The learning dynamics of a symmetric 2-layer linear model.** Left: The change of the test loss and the Jacobian entries with time predicted by the theory; Right: the corresponding model output trajectory. The figures are plotted under $s = 2$ and all entries of $\boldsymbol{W}$ are initialized positive.

entries. Notice that the irrelevant entries do not contribute to the output of the test point so we will not discuss them. Moreover, we also divide minor entries into several groups. The minor entries in the p-th row or column belongs to the p-th group (thus each entry belongs to two groups). See Fig. 4 for an illustration of the division of the entries.

Initial Growth. In this section, we assume $w_{i,j} \forall i, j$ are initialized around a very small value ω such that $\omega \ll \frac{1}{d \max_{i\in[s]} a_i}$ (See App. D.1 for specific assumptions). It is evident that when all $w_{i,j}$ are close to ω (we call this period the **initial phase**), the growth term is $\Theta(\omega)$, while the suppression term and the noise term are $\Theta(\omega^2)$. This suggests that the evolution of $w_{i,j}$ is dominated by the growth term. Therefore, in the initial phase, every value in the Jacobian grows towards the direction of increasing its absolute value, with the speed determined by $a_i + a_j$. Since we assumed that $\boldsymbol{a}$ is ordered in a descending order, it is evident that each entry grows faster than those below it or to its right. The Initial Growth stage is formally characterized by Lemmas D.1 to D.3.

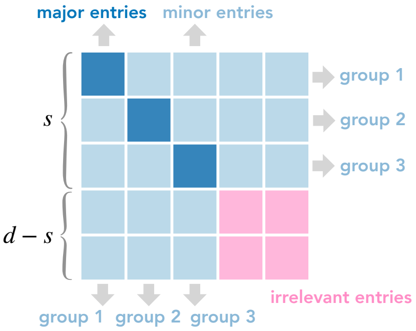

Figure 4: An illustration of the entries of the Jacobian.

First Suppression. In the Initial Growth stage, the first major entry will be the one that grows exponentially faster than all other entries, making it the first one that leaves the initial phase. Once the first major entry becomes significant and non-negligible, it will effect on the suppression term of all minor entries in the first group. When the difference between a_1 and a_2 is large enough, the first major entry is able to flip the growth direction of the first group of minor entries and push their values to 0. The suppression stages are characterized by Lemma D.7.

Second Growth and Cycle. Once the suppression of the first group of minor entries takes effect, the second major entry becomes the one that grows fastest. Thus, the second major entry will be the second one that leaves the initial stage. Again, when the second major entry becomes large enough, it will suppress the second group of minor entries and push their value to 0. This process continues like this: the growth of a major entry is followed by the suppression of the corresponding group of minor entries, which, in turn, leaves space for the growth of the next major entry. The general growth stages are characterized by Lemma D.4 and the fate of off-diagonal entries is characterized by Lemma D.8.

Growth Slow Down and Stop. Notice that the suppression term of a major entry is also influenced by its own magnitude. Therefore, when a major entries becomes significantly large, it also suppresses itself, leading to the slowing down of its growth. Note that this effect only slows down the growth but will not reverse the direction, since for major entries the suppression term is always smaller than the growth term, until $w_{i,i}$ becomes 1 where the growth and suppression terms are equal and the evolution stops. The terminal stage of the growth of major entries are characterized by Lemma D.5.

4.2.2 Explaining Model Behavior

Recall that we have $f\left(\boldsymbol{U}(t);\widehat{\boldsymbol{x}}\right)_k = \sum_{p=1}^{s} w_{k,p}(t)\mu_p$. We now explain how the stage-wise evolution of Jacobian described in Sec. 4.2.1 determines the evolution of the model output.

Generalization Order and Terminal Phase Slowing Down. From the discussions in Sec. 4.2.1, by the end of the training, all the major entries converge to 1 and all minor entries converge to 0. The major entries grows in the order of corresponding a_p, which is determined by μ_p and σ_p, and slows down when approaching the terminal. This explains our observation that directions with larger μ_p and / or σ_p is learned first, as well as the terminal phase slowing down.

Transient Memorization and Non-monotonic Loss Curve. We argue the Transient Memorization and the non-monotonic loss curve is caused by the multi-stage major growth vs. minor growth / suppression process. Importantly, in certain configurations, minor entries growing towards larger absolute values (which is the incorrect solution) can lead to the decay of the OOD test loss, and cause an "illusion of generalizing" that the output trajectory is moving towards improving OOD generalization. However, this effect is later eliminated by the suppression of the corresponding minor entries, leading to a double (or multiple) descent-like loss curve and a reversal in the output trajectory.

More concretely, consider the first (initial) growth stage as an example. In this stage, for each $k \in [s]$, $f\left(\boldsymbol{U}(t);\widehat{\boldsymbol{x}}\right)_k$ is dominated by $w_{k,1}(t)\mu_k$, since $w_{k,1}$ grows fastest among all the entries in the k-th row. If $w_{k,1}$ happens to be initialized positive, then $f\left(\boldsymbol{U}(t);\widehat{\boldsymbol{x}}\right)_k$ grows towards 1, which is the correct direction[3], and loss thus decays. Since in a symmetric initialization, each entry has equal chance of being initialized positive or negative, when s is small, it is easy to have many minor entries initialized positive, whose growth contributes to the decaying of loss. *This causes an illusion that the model is going towards the right direction of OOD generalization.* After the minor entries of the first group are suppressed, their contribution to the decaying of the loss is canceled, which leads to the output trajectory turning back to the direction of memorizing a training cluster and a transient loss increase.

Fig. 5 presents the loss curve and the Jacobian entry evolution predicted by the theory with a specific initialization. Notice how, as claimed above, the first and second descending of loss accurately corresponds to the initial and second growth of the major entries, and the ascending of the loss corresponds to the suppression of the minor entries. When $s > 2$, there are multiple turns of growth and suppression stages and can possibly leads to a multiple-descent-like loss curve, which we confirm and illustrate in App. E.1.

Remark on Failure Modes. We note that our theory also provides an explanation on instances when the model fails to achieve OOD generalization when one or more of our assumptions outlined in App. D.1 breakdown. A specific case is when a major entry $w_{k,k}$ is overly suppressed by a corresponding minor entry before it can begin to grow, causing the growth term $G_{k,k}$ becomes nearly zero. Consequently, the model output at $\widehat{\boldsymbol{x}}$ in that direction converges to 0, instead of μ_k as expected. See App. E.3 for more discussions and illustrations.

Remark on Existing Work. There has been extensive research on the non-monotonic behavior of linear neural networks (in various settings). We note that existing studies either focus on one-layer networks [57, 26] or diagonally initialized networks [53, 39, 16, 55], which essentially make the evolution of each direction decoupled. This decoupling simplifies the learning dynamics and can overlook critical aspects thereof (as we discussed in the preceding subsection). In contrast, our analysis, through a careful treatment of each entry of the Jacobian, does not need to make the diagonal initialization assumption, hence allowing us to capture and characterize the rich behaviors that arise from the interaction between different directions.

[3] Notice that this is true even when $k \neq 1$, i.e. $w_{k,1}$ is a minor entry.

5 Diffusion Model Results

Tying back to our original motivation of devising an abstraction of *concept space* first explored in text-to-image generative diffusion models, we now aim to verify if our novel theoretical findings can be reproduced in a more involved empirical setup with diffusion models. To this end, we borrow the setup from [54, 52] and train conditional image diffusion models on two concepts—`size` and `color`.

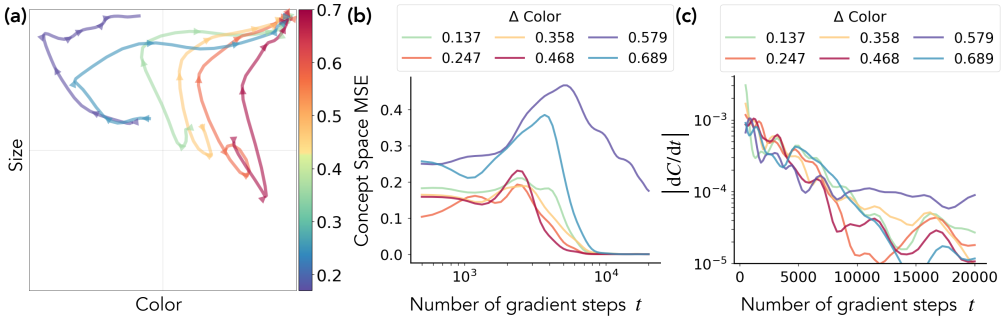

Figure 6: **Main observations reproduced on prompt-to-image diffusion models.** (a) Signal strength controls generalization speed and order. (b) Transient Memorization: Diffusion models also undergo a Transient Memorization phase. This induces a double-descent like curve for the concept space MSE. (c) Concept learning gradually slows down. Our theory predicts speed of concept learning slows down at an exponential rate, which broadly matches the experimental results. For details of the experiment, please see App. G.

Fig. 6 illustrates the three main observations from our theory reproduced with a prompt-to-image diffusion model. We trained diffusion models to compose two concepts, `color` and `size`, as in [54]. See App. G for experimental details. Fig. 6 (a) shows that the level of concept signal, here corresponding to the difference of class mean pixel values, largely alters the generalization dynamics. Specifically, we see that the speed and order of compositional generalization is determined by concept signal, and the signal intensity can reverse the order. *The latter is especially important since we show the findings from our theory and the SIM task, where we abstracted concepts into coordinates and Gaussian clusters, generalize to two naturalistic concepts*: `color` and `size`. We also see *Transient Memorization* occurs in diffusion training, where the generalization dynamics show a bend towards the concept with stronger signal. This bend is transient and the generated class eventually converges to the intended concept space coordinate. Fig. 6 (b) quantifies this further via a concept space MSE metric. We observe that the concept space MSE has a phase where it increases before entering a generalization phase. This is well aligned with our findings in Sec. 4.2. Fig. 6 (c) confirms that the speed of compositional generalization, quantified by the absolute of concept space traversal distance per step, decelerates at an exponential rate, as expected from our theoretical findings (Theorem 4.1).

6 Conclusion

In this paper, we propose SIM task as a further abstraction of the "concept space" previously explored by [52, 54]. We conduct comprehensive investigation into the behaviors of a regression model trained on SIM, both empirically and theoretically, demonstrating that the learning dynamics on SIM effectively captures the phenomena observed on image generation task, establishing SIM as a basis for studying compositional generalization. We make a comprehensive. Critically, our theoretical analysis uncovers the underlying causes of several phenomena that previously observed on compositional generalizations, as well as predicting new ones that characterizes the multi-stage and non-monotonic learning dynamics, which have been largely overlooked in earlier research. Our diffusion model experiments further verify the validity of our analysis. Additional discussions and potential future work directions can be found in App. E.

References

[1] Madhu S Advani, Andrew M Saxe, and Haim Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020.

[2] Jacob Andreas. Measuring compositionality in representation learning. *arXiv preprint arXiv:1902.07181*, 2019.

[3] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. *arXiv preprint arXiv:1810.02281*, 2018.

[4] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems*, 32, 2019.

[5] Sanjeev Arora and Anirudh Goyal. A theory for emergence of complex skills in language models. *arXiv preprint arXiv:2307.15936*, 2023.

[6] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

[7] Emanuele Bugliarello and Desmond Elliott. The role of syntactic planning in compositional image captioning. *arXiv preprint arXiv:2101.11911*, 2021.

[8] Colin Conwell and Tomer Ullman. Testing relational understanding in text-guided image generation. *arXiv preprint arXiv:2208.00005*, 2022.

[9] Colin Conwell and Tomer Ullman. A comprehensive benchmark of human-like relational reasoning for text-to-image foundation models. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023.

[10] José F Díez-Pastor, Juan J Rodríguez, César I García-Osorio, and Ludmila I Kuncheva. Diversity techniques improve the performance of the best imbalance learning ensembles. *Information Sciences*, 325:98–117, 2015.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[12] Simon Du and Wei Hu. Width provably matters in optimization for deep linear neural networks. In *International Conference on Machine Learning*, pages 1655–1664. PMLR, 2019.

[13] Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. *arXiv preprint arXiv:2302.11552*, 2023.

[14] Yilun Du and Leslie Kaelbling. Compositional generative modeling: A single model is not all you need. *arXiv preprint arXiv:2402.01103*, 2024.

[15] Yilun Du, Shuang Li, Yash Sharma, Josh Tenenbaum, and Igor Mordatch. Unsupervised learning of compositional energy concepts. *Advances in Neural Information Processing Systems*, 34:15608–15620, 2021.

[16] Mathieu Even, Scott Pesme, Suriya Gunasekar, and Nicolas Flammarion. (s) gd over diagonal linear networks: Implicit bias, large stepsizes and edge of stability. *Advances in Neural Information Processing Systems*, 36:29406–29448, 2023.

[17] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022.

[18] Jerry Fodor. Language, thought and compositionality. *Royal Institute of Philosophy Supplements*, 48:227–242, 2001.

[19] Jerry A Fodor et al. *The language of thought*, volume 5. Harvard university press Cambridge, MA, 1975.

[20] Steven M Frankland and Joshua D Greene. Concepts and compositionality: in search of the brain's language of thought. *Annual review of psychology*, 71:273–303, 2020.

[21] Nicholas T Franklin and Michael J Frank. Compositional clustering in task structure learning. *PLoS computational biology*, 14(4):e1006116, 2018.

[22] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022.

[23] Zhiqiang Gong, Ping Zhong, and Weidong Hu. Diversity in machine learning. *Ieee Access*, 7:64323–64350, 2019.

[24] Noah D Goodman, Joshua B Tenenbaum, Thomas L Griffiths, and Jacob Feldman. Compositionality in rational analysis: Grammar-based induction for concept learning. *The probabilistic mind: Prospects for Bayesian cognitive science*, 2008.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[26] Reinhard Heckel and Fatih Furkan Yilmaz. Early stopping in deep networks: Double descent and how to eliminate it. *arXiv preprint arXiv:2007.10099*, 2020.

[27] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2023.

[28] Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. Compositionality decomposed: How do neural networks generalise? *Journal of Artificial Intelligence Research*, 67:757–795, 2020.

[29] Ben Hutchinson, Jason Baldridge, and Vinodkumar Prabhakaran. Underspecification in scene description-to-depiction tasks, 2022.

[30] Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. *arXiv preprint arXiv:1810.02032*, 2018.

[31] Jikai Jin, Zhiyuan Li, Kaifeng Lyu, Simon Shaolei Du, and Jason D Lee. Understanding incremental learning of gradient descent: A fine-grained analysis of matrix sensing. In *International Conference on Machine Learning*, pages 15200–15238. PMLR, 2023.

[32] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.

[33] Simran Kaur, Simon Park, Anirudh Goyal, and Sanjeev Arora. Instruct-skillmix: A powerful pipeline for llm instruction tuning. *arXiv preprint arXiv:2408.14774*, 2024.

[34] Mikail Khona, Maya Okawa, Jan Hula, Rahul Ramesh, Kento Nishi, Robert Dick, Ekdeep Singh Lubana, and Hidenori Tanaka. Towards an understanding of stepwise inference in transformers: A synthetic graph navigation model. *arXiv preprint arXiv:2402.07757*, 2024.

[35] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

[36] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. *arXiv preprint arXiv:2303.13516*, 2023.

[37] Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pages 2873–2882. PMLR, 2018.

[38] Brenden M Lake and Marco Baroni. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121, 2023.

[39] Andrew K Lampinen and Surya Ganguli. An analytic theory of generalization dynamics and transfer learning in deep linear networks. *arXiv preprint arXiv:1809.10374*, 2018.

[40] Evelina Leivada, Elliot Murphy, and Gary Marcus. Dall-e 2 fails to reliably capture common syntactic processes. *arXiv preprint arXiv:2210.12889*, 2022.

[41] Michael A Lepori, Thomas Serre, and Ellie Pavlick. Break it down: Evidence for structural compositionality in neural networks. *arXiv preprint arXiv:2301.10884*, 2023.

[42] Martha Lewis, Qinan Yu, Jack Merullo, and Ellie Pavlick. Does clip bind concepts? probing compositionality in large image models. *arXiv preprint arXiv:2212.10537*, 2022.

[43] Hao Li, Yang Zou, Ying Wang, Orchid Majumder, Yusheng Xie, R Manmatha, Ashwin Swaminathan, Zhuowen Tu, Stefano Ermon, and Stefano Soatto. On the scalability of diffusion-based text-to-image generation. *arXiv preprint arXiv:2404.02883*, 2024.

[44] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. *arXiv preprint arXiv:2012.09839*, 2020.

[45] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, pages 423–439. Springer, 2022.

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019.

[47] Ekdeep Singh Lubana, Kyogo Kawaguchi, Robert P Dick, and Hidenori Tanaka. A percolation model of emergence: Analyzing transformers trained on a formal language. *arXiv preprint arXiv:2408.12578*, 2024.

[48] Hartmut Maennel, Olivier Bousquet, and Sylvain Gelly. Gradient descent quantizes relu network features. *arXiv preprint arXiv:1803.08367*, 2018.

[49] Gary Marcus, Ernest Davis, and Scott Aaronson. A very preliminary analysis of dall-e 2. *arXiv preprint arXiv:2204.13807*, 2022.

[50] Hancheng Min, Enrique Mallada, and René Vidal. Early neuron alignment in two-layer relu networks with small initialization. *arXiv preprint arXiv:2307.12851*, 2023.

[51] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.

[52] Maya Okawa, Ekdeep Singh Lubana, Robert P. Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. *Advances in Neural Information Processing Systems*, 2023.

[53] Amanda Olmin and Fredrik Lindsten. Towards understanding epoch-wise double descent in two-layer linear neural networks. *arXiv preprint arXiv:2407.09845*, 2024.

[54] Core Francisco Park, Maya Okawa, Andrew Lee, Ekdeep Singh Lubana, and Hidenori Tanaka. Emergence of hidden capabilities: Exploring learning dynamics in concept space. *Advances in Neural Information Processing Systems*, 2024.

[55] Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. *Advances in Neural Information Processing Systems*, 36:7475–7505, 2023.

[56] Scott Pesme, Loucas Pillaud-Vivien, and Nicolas Flammarion. Implicit bias of sgd for diagonal linear networks: a provable benefit of stochasticity. *Advances in Neural Information Processing Systems*, 34:29218–29230, 2021.

[57] Mohammad Pezeshki, Amartya Mitra, Yoshua Bengio, and Guillaume Lajoie. Multi-scale feature learning dynamics: Insights for double descent. In *International Conference on Machine Learning*, pages 17669–17690. PMLR, 2022.

[58] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021.

[59] Rahul Ramesh, Mikail Khona, Robert P Dick, Hidenori Tanaka, and Ekdeep Singh Lubana. How capable can a transformer become? a study on synthetic, interpretable tasks. *arXiv preprint arXiv:2311.12997*, 2023.

[60] Royi Rassin, Shauli Ravfogel, and Yoav Goldberg. Dalle-2 is seeing double: flaws in word-to-concept mapping in text2image models. *arXiv preprint arXiv:2210.10606*, 2022.

[61] Carlo Reverberi, Kai Görgen, and John-Dylan Haynes. Compositionality of rule representations in human prefrontal cortex. *Cerebral cortex*, 22(6):1237–1246, 2012.

[62] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[64] Jacob Russin, Sam Whitman McGrath, Ellie Pavlick, and Michael J Frank. Is human compositionality meta-learned? *Behavioral and Brain Sciences*, 47:e162, 2024.

[65] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[66] Lukas Schott, Julius Von Kügelgen, Frederik Träuble, Peter Gehler, Chris Russell, Matthias Bethge, Bernhard Schölkopf, Francesco Locatello, and Wieland Brendel. Visual representation learning does not generalize strongly within the same domain. *arXiv preprint arXiv:2107.08221*, 2021.

[67] Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate dall-e learns to compose. *arXiv preprint arXiv:2110.11405*, 2021.

[68] Paul Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, 46(1-2):159–216, 1990.

[69] Sam Spilsbury and Alexander Ilin. Compositional generalization in grounded language learning via induced model sparsity. *arXiv preprint arXiv:2207.02518*, 2022.

[70] Dominik Stöger and Mahdi Soltanolkotabi. Small random initialization is akin to spectral learning: Optimization and generalization guarantees for overparameterized low-rank matrix reconstruction. *Advances in Neural Information Processing Systems*, 34:23831–23843, 2021.

[71] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022.

[72] Vishaal Udandarao, Ameya Prabhu, Adhiraj Ghosh, Yash Sharma, Philip HS Torr, Adel Bibi, Samuel Albanie, and Matthias Bethge. No" zero-shot" without exponential data: Pretraining concept frequency determines multimodal model performance. *arXiv preprint arXiv:2404.04125*, 2024.

[73] Josef Valvoda, Naomi Saphra, Jonathan Rawski, Adina Williams, and Ryan Cotterell. Benchmarking compositionality with formal languages. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6007–6018, 2022.

[74] Thaddäus Wiedemer, Prasanna Mayilvahanan, Matthias Bethge, and Wieland Brendel. Compositional generalization from first principles. *Advances in Neural Information Processing Systems*, 36, 2024.

[75] Guangyue Xu, Parisa Kordjamshidi, and Joyce Chai. Prompting large pre-trained vision-language models for compositional concept learning. *arXiv preprint arXiv:2211.05077*, 2022.

[76] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bag-of-words models, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

[77] Tian Yun, Usha Bhalla, Ellie Pavlick, and Chen Sun. Do vision-language pretrained models learn primitive concepts? *arXiv preprint arXiv:2203.17271*, 2022.

[78] Hattie Zhou, Arwen Bradley, Etai Littwin, Noam Razin, Omid Saremi, Josh Susskind, Samy Bengio, and Preetum Nakkiran. What algorithms can transformers learn? a study in length generalization. *arXiv preprint arXiv:2310.16028*, 2023.

A Related Work

In this section, we provide some context for this paper by reviewing some existing work on compositional generalization and the study of deep linear networks.

Compositional Generalization. Prior work on compositionality has often focused on benchmarking of pretrained models [71, 2, 42, 37, 77, 41, 32, 8, 76, 66, 22, 73] or proposition of protocols that allow generation of compositional samples [15, 13, 45, 75, 76, 7, 69, 36, 14]. While perfect compositionality in natural settings is still lacking [49, 40, 8, 9, 22, 13, 45, 67, 60, 17, 29], several works have demonstrated via use of toy settings that this is unlikely to be an expressibility issue, as was hypothesized, e.g., by [19], since the model can in fact learn to perfectly compose in said toy settings. The ability to compose is in fact rather distinctly emergent [52, 47] and the model learning it often correlates with distinctive patterns in the learning dynamics, as identified by [54]. We note that there has in fact been some work on understanding compositional generalization abilities in neural networks [74, 72, 59], but, unlike us, the focus of these papers is not on the model's learning dynamics.

Learning Dynamics of Deep Linear Networks. Deep linear networks has been a commonly studied model for learning dynamics, and existing works mostly focus on the final solution found by the model, which primarily concerns the stationary point of the dynamics [3, 30, 12, 4, 1]. There have also been works that try to characterize the full learning dynamics; however, they generally require the learning of each direction (neuron) to be decoupled [65, 53, 39, 16, 55, 56], which can be realized through a specific initialization choice. The decoupling assumption ignores the interaction between different neurons and highly simplify the dynamics, and as we mentioned in Sec. 4.1, make it unable to capture some important phenomena in practice. The symmetric 2-layer linear model is also a specific model that is frequently studied, especially in matrix sensing [44, 70, 31], and as we noted in Sec. 4.2, current theoretical results of this model focus on the implicit biases in the solutions learned, while our analysis, on the other hand, aims at characterizing the full learning dynamics and focus on its OOD behavior.

B Model Compositionally Generalize in Topologically Constrained Order

In this section, we introduce another phenomenon observed on SIM task learning that we do not put in the main paper: the order of compositional generalization happens in a topologically constrained order.

In this section, instead of the single test point $\widehat{\boldsymbol{x}}$, we introduce a hierarchy of test points. Specifically, let $\mathcal{I} = \{0,1\}^s$ be the index set of test points. For each $\boldsymbol{v} \in \mathcal{I}$, we define a test point

$$\widehat{\boldsymbol{x}}^{(\boldsymbol{v})} = \sum_{p=1}^{s} v_p \mu_p \mathbf{1}_p, \tag{B.1}$$

and call $\widehat{\boldsymbol{x}}^{(\boldsymbol{v})}$ the test point with the index $\boldsymbol{v}$. Intuitively, the index $\boldsymbol{v}$ describes which training sets are combined into the current test point. If $\|\boldsymbol{v}\| = 1$ then $\widehat{\boldsymbol{x}}^{(\boldsymbol{v})}$ is the center of one of the training clusters.

We assign the component-wise ordering $\preceq$ to the index set $\mathcal{I}$, i.e., for $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{I}$, we say $\boldsymbol{u} \preceq \boldsymbol{v}$ if and only if $\forall i \in [n], u_i \leq v_i$. It's easy to see that $\preceq$ is a partial-ordering.

Interestingly, in the SIM experiment, the order of the generalization in different test points strictly follow the component-wise order. This finding can be described formally in the following way: the loss function is an order homomorphism between $\preceq$ on the index set, and $\leq$ on the real number. Let $\ell(\boldsymbol{z})$ be the loss function of the test point $\boldsymbol{z}$, then we have the following empirical observation:

$$\forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{I}, \boldsymbol{u} \preceq \boldsymbol{v} \implies \ell\left(\tilde{\boldsymbol{x}}^{(u)}\right) \leq \ell\left(\tilde{\boldsymbol{x}}^{(v)}\right). \tag{B.2}$$

In Fig. 7 we show the loss of each test point in several timepoints, with $\boldsymbol{\mu} = (1,2,3,4)$, $\boldsymbol{\sigma} = \left\{\frac{1}{2}\right\}^4$. There is a clear trend that the test points that are on the right of the graph (larger in the component-wise order) will only be learned after all of its predecessors are all learned. We call this phenomenon the *topological constraint* since the constraint is based on the topology of the graph in Fig. 7.

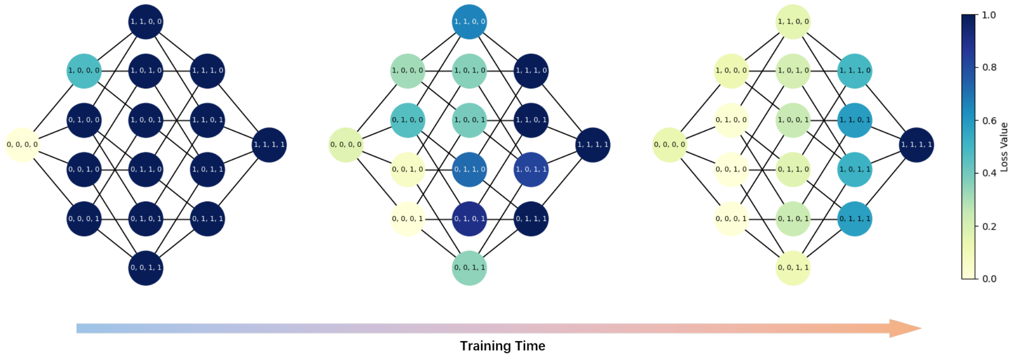

Figure 7: The loss at each test point in different timepoints during training for a 2-layer MLP with ReLU activation. Each graph represents a timepoint. Each node in the graph represents a test point, with index printed on it, and edges connecting nodes with Hamming distance 1. The color of the graph represents the loss of corresponding test point. Notice that we truncate the loss at 1 in order to unify the scale. From lest to right: epoch = $1, 3, 5$.

C Proofs and Calculations

In the main text we have omitted some critical proofs and calculations due to space limitation. In this section we provide the complete derivations. Note that we postpone the proof of related theorems of Sec. 4.2 to App. D because of their length.

C.1 The Loss Function with Linear Model and Infinite Data Limit

In this subsection we derive the transformed loss function eq. (4.1), as well as the expression of the data matrix $\boldsymbol{A}$. For convenience we denote $\boldsymbol{W}_{\boldsymbol{\theta}}$ by $\boldsymbol{W}$. We have

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\theta}) &= \frac{1}{2ns}\sum_{p=1}^{s}\sum_{k=1}^{n}\left\|(\boldsymbol{W}-\boldsymbol{I})\boldsymbol{x}_k^{(p)}\right\|^2 && \text{(C.1)}\\
&= \frac{1}{2ns}\operatorname{Tr}\left[\boldsymbol{x}_k^{(p)\top}(\boldsymbol{W}-\boldsymbol{I})^\top(\boldsymbol{W}-\boldsymbol{I})\boldsymbol{x}_k^{(p)}\right] && \text{(C.2)}\\
&= \frac{1}{2ns}\operatorname{Tr}\left[(\boldsymbol{W}-\boldsymbol{I})^\top(\boldsymbol{W}-\boldsymbol{I})\boldsymbol{x}_k^{(p)}\boldsymbol{x}_k^{(p)\top}\right] && \text{(C.3)}\\
&= \frac{1}{2}\operatorname{Tr}\left[(\boldsymbol{W}-\boldsymbol{I})^\top(\boldsymbol{W}-\boldsymbol{I})\frac{1}{ns}\boldsymbol{x}_k^{(p)}\boldsymbol{x}_k^{(p)\top}\right] && \text{(C.4)}\\
&= \frac{1}{2}\operatorname{Tr}\left[\boldsymbol{A}^{1/2}(\boldsymbol{W}-\boldsymbol{I})^\top(\boldsymbol{W}-\boldsymbol{I})\boldsymbol{A}^{1/2}\right] && \text{(C.5)}\\
&= \frac{1}{2}\left\|(\boldsymbol{W}-\boldsymbol{I})\,\boldsymbol{A}^{1/2}\right\|_{\mathcal{F}}^2 . && \text{(C.6)}
\end{aligned}$$

Let $\mathcal{G}$ be the data generating process. It can be viewed as two components: first assign one of the s clusters, and then draw a Gaussian vector from a Gaussian distribution in that cluster. Specifically, let $\boldsymbol{x}$ be an arbitrary sample from the traning set, then the distribuition of $\boldsymbol{x}$ is equal to

$$\boldsymbol{x} \simeq \boldsymbol{\mu}^{(\eta)} + \operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\xi}, \tag{C.7}$$

where η is a uniform random variable taking values in $[s]$ and $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is a random Gaussian vector that is independent from η. Here $\simeq$ represents having the same distribution.

When $n \to \infty$, the data matrix $\boldsymbol{A}$ converges to the true covariance, which is is

$$\boldsymbol{A} \to \mathbb{E}\left(\boldsymbol{x}\boldsymbol{x}^\top\right) \tag{C.8}$$

$$= \mathbb{E}\left[\left(\boldsymbol{\mu}^{(\eta)} + \operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\xi}\right)\left(\boldsymbol{\mu}^{(\eta)} + \operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\xi}\right)^\top\right] \tag{C.9}$$

$$= \mathbb{E}\left(\boldsymbol{\mu}^{(\eta)}\boldsymbol{\mu}^{(\eta)\top}\right) + \mathbb{E}\operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\xi}\boldsymbol{\xi}^\top\operatorname{diag}(\boldsymbol{\sigma}) \tag{C.10}$$

$$= \frac{1}{s}\sum_{p=1}^{s}\boldsymbol{\mu}^{(p)}\boldsymbol{\mu}^{(p)\top} + \operatorname{diag}(\boldsymbol{\sigma})^2 \tag{C.11}$$

$$= \frac{1}{s}\sum_{p=1}^{s}\mu_p^2\mathbf{1}_p\mathbf{1}_p^\top + \operatorname{diag}(\boldsymbol{\sigma})^2 \tag{C.12}$$

$$= \frac{1}{s}\operatorname{diag}(\boldsymbol{\mu})^2 + \operatorname{diag}(\boldsymbol{\sigma})^2. \tag{C.13}$$

C.2 Proof of Theorem 4.1

In this subsection for the notation-wise convenience we denote $\boldsymbol{W} = \boldsymbol{\theta}$. Since the model is one-layer, the loss function eq. (4.1) becomes

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{2}\left\|(\boldsymbol{W} - \boldsymbol{I})\boldsymbol{A}^{1/2}\right\|_{\mathcal{F}}^2, \tag{C.14}$$

and the gradient is

$$\nabla\mathcal{L}(\boldsymbol{W}) = (\boldsymbol{W} - \boldsymbol{I})\boldsymbol{A} = \boldsymbol{W}\boldsymbol{A} - \boldsymbol{A}. \tag{C.15}$$

We denote the k-th row of $\boldsymbol{W}$ and $\boldsymbol{A}$ by $\boldsymbol{w}_k$ and $\boldsymbol{A}_k$ respectively. Then we have

$$\dot{\boldsymbol{w}}_k = -\boldsymbol{A}\boldsymbol{w}_k + \boldsymbol{a}_k. \tag{C.16}$$

The solution of this differential equation is

$$\boldsymbol{w}_k(t) = \exp\left(-\boldsymbol{A}t\right)\left[w_k(0) - \boldsymbol{A}^{-1}\boldsymbol{a}_k\right] + \boldsymbol{A}^{-1}\boldsymbol{a}_k, \tag{C.17}$$

where we use the convention $0 \times \left(0^{-1}\right) = 0$ to avoid the non-invertible case of $\boldsymbol{A}$.

Thus for any $\boldsymbol{z} \in \mathbb{R}^d$ we have

$$f(\boldsymbol{W}(t); \boldsymbol{z})_k = \langle \boldsymbol{w}_k(t), \boldsymbol{z} \rangle \tag{C.18}$$

$$= \left\langle \left(\boldsymbol{I} - e^{-\boldsymbol{A}t}\right)\boldsymbol{A}^{-1}\boldsymbol{a}_k, \boldsymbol{z} \right\rangle + \left\langle e^{-\boldsymbol{A}t}w_k(0), \boldsymbol{z} \right\rangle \tag{C.19}$$

$$= \sum_{p=1}^{n}\frac{1 - e^{-a_p t}}{a_p}\mathbb{1}_{\{k=p\}}a_p z_p + \sum_{i=1}^{n}e^{-a_i t}w_{k,i}(0)z_i \tag{C.20}$$

$$= \mathbb{1}_{\{k \le s\}}\left(1 - e^{-a_k t}\right)z_k + \sum_{i=1}^{n}e^{-a_i t}w_{k,i}(0)z_i, \tag{C.21}$$

and this proves the claim.

D Theoretical Analysis of the Two Layer Model

In this section we provide a detailed analysis of the symmetric two-layer linear model described in Sec. 4.2.

In this section we assume a finite step size, i.e., $\boldsymbol{W} : \mathbb{N} \to \mathbb{R}^{d \times d}$ is initialized by $\boldsymbol{W}(0)$ and updated by

$$\frac{\boldsymbol{W}(t+1) - \boldsymbol{W}(t)}{\eta} = -\boldsymbol{U}(t)\nabla\mathcal{L}(\boldsymbol{U}(t))^\top - \nabla\mathcal{L}(\boldsymbol{U}(t))\boldsymbol{U}(t)^\top \tag{D.1}$$

$$= \boldsymbol{W}(t)\boldsymbol{A} + \boldsymbol{A}\boldsymbol{W}(t) - \frac{1}{2}\left[\boldsymbol{A}\boldsymbol{W}(t)^2 + \boldsymbol{W}(t)^2\boldsymbol{A} + 2\boldsymbol{W}(t)\boldsymbol{A}\boldsymbol{W}(t)\right]. \tag{D.2}$$

The update of each entry $w_{i,j}(t)$ can be decomposed into three terms, as we described in the main text:

$$\begin{aligned}\frac{w_{i,j}(t+1)-w_{i,j}(t)}{\eta} =& w_{i,j}(t)(a_i+a_j) - \frac{1}{2}\sum_{k=1}^{d} w_{k,i}w_{k,j}(a_i+a_j+2a_k) && \text{(D.3)}\\ =& \underbrace{w_{i,j}(t)(a_i+a_j)}_{G_{i,j}(t)} && \text{(D.4)}\\ & \underbrace{-\frac{1}{2}w_{i,j}\left[w_{i,i}(3a_i+a_j)+\mathbb{1}_{\{i\neq j\}}w_{j,j}(3a_j+a_i)\right]}_{S_{i,j}(t)} && \text{(D.5)}\\ & \underbrace{-\frac{1}{2}\sum_{\substack{k\neq i\\ k\neq j}} w_{k,i}(t)w_{k,j}(t)(a_i+a_j+2a_k)}_{N_{i,j}(t)}\,. && \text{(D.6)}\end{aligned}$$

D.1 Assumptions

We need make several assumptions to prove the results. Below we make several assumptions that all commonly hold in the practice. The first assumption to make is that both the value of a_k and the initialization of $\boldsymbol{W}$ is bounded.

Assumption D.1 (Bounded Initialization and Signal Strength)**.** *There exists $\alpha > 0, \gamma > 1, \beta > 1$ such that*

$$\forall k, \alpha \leq a_k \leq \gamma\alpha, \qquad \text{(D.7)}$$
$$\forall i, j, \omega \leq |w_{i,j}(0)| \leq \beta\omega. \qquad \text{(D.8)}$$

The second assumption is that the step size is small enough.

Assumption D.2 (Small Step Size)**.** *There exists a constant $K \geq 20$, such that $\eta \leq \frac{1}{9K\gamma\alpha}$.*

Next, we define a concept called initial phase. The definition of initial phase is related to a constant $P > 0$.

Definition D.1. *Assume there is a constant $P > 0$. For an entry (i, j) and time t, if $|w_{i,j}(t)| \leq P\beta\omega$, we say this entry is in **initial phase**.*

The next assumption to make the that the boundary of the initial phase should not be too large.

Assumption D.3 (Small Initial Phase)**.** $P\omega\beta \leq 0.4$.

The next assumption to make are that the intialization value (ω) should not be too large.

Assumption D.4 (Small Initialization)**.**

$$\omega \leq \min\left\{\frac{\min\{\kappa-1, 1-\kappa^{-1/2}\}}{PK\gamma d\beta^2}, \frac{1}{\sqrt{2\beta}}\right\} \qquad \text{(D.9)}$$

and $\kappa > 1.1$, and $\kappa \leq 1 + \frac{1}{2}KC^{-1}$, $P \geq 2$.

Finally, we also assume that the signal strength difference is significant enough.

Assumption D.5 (Significant Signal Strength Difference)**.** *For any $i > j$, we have*

$$\frac{a_i + a_i}{2a_i} \leq \frac{\log P}{10\kappa^2 \log\frac{1}{P\beta\omega} + \log P\beta}. \qquad \text{(D.10)}$$

and there exists a constant $C > 1$ such that $a_i - 3a_j \geq C^{-1}\alpha$.

D.2 The Characterization of the Evolution of the Jacobian

In this subsection, we provide a series of lemmas that characterize each stage the evolution of the Jacobian matrix $\boldsymbol{W}$.

The whole proof is based on induction, and in order to avoid a too complicated induction, we make the following assertion, which obviously holds at initialization.

Assertion D.1. *For all $t \in \mathbb{N}$, if $i \neq j$, then the entry (i, j) stays in the initial phase for all time.*

We will use Assertion D.1 as an assumption throughout the proves and prove it at the end. This is essentially another way of writing inductions.

We have the following corollary that directly followed by Assertion D.1.

Corollary D.1. *For all $t \in \mathbb{N}$ and all i, j, $|N_{i,j}(t)| \leq 2P\gamma\alpha d\beta^2\omega^2$.*

Now, we are ready to present and prove the major lemmas. The first lemma is to post a (rather loose) upper bound of the value of the entries.

Lemma D.1 (Upper Bounded Growth)**.** *Consider entry (i, j). We have for all $t \in \mathbb{N}$, at timepoint t the absolute value of the (i, j)-th entry satisfies*

$$|w_{i,j}(t)| \leq |w_{i,j}(0)| \exp\left[\eta t(a_i + a_j)\kappa\right]. \tag{D.11}$$

Proof. Since of the $N_{i,j}$ term we only use its absolute value, the positive case and negative case are symmetric. WLOG we only consider the case where $w_{i,j}(0) > 0$ here.

The claim is obviously satisfied at initialization. We use it as the inductive hypothesis. Suppose at timepoint $t \leq T - 1$ the claim is satisfied, we consider the time step $t + 1$.

Since Assertion D.1 guaranteed that every non-diagonal entry is in the initial phase, and the $S_{i,j}$ term has different symbol with $w_{i,j}(0)$, we have

$$S_{i,j}(t) + N_{i,j}(t) \leq 2P\gamma\alpha d\beta^2\omega^2. \tag{D.12}$$

We have

$$\begin{aligned}
w_{i,j}(t+1) - w_{i,j}(t) &\leq \eta w_{i,j}(t)(a_i + a_j) + 4\eta\gamma\alpha d\beta_0\omega^2 && \text{(D.13)}\\
&\leq \eta(a_i + a_j)w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa\right] + 2P\eta\gamma\alpha d\beta^2\omega^2 && \text{(D.14)}\\
&= w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa\right]\left[\eta(a_i + a_j) + \frac{2P\gamma\alpha d\beta^2\omega^2}{w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa\right]}\right] && \text{(D.15)}
\end{aligned}$$

From Assumption D.4, we have

$$\begin{aligned}
\eta(a_i + a_j) + \frac{2P\gamma\alpha d\beta^2\omega^2}{w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa\right]} &\leq \eta(a_i + a_j) + 2P\gamma\alpha d\beta^2\omega && \text{(D.16)}\\
&\leq \eta(a_i + a_j) + 2(\kappa - 1)\eta\alpha && \text{(D.17)}\\
&\leq \kappa\eta(a_i + a_j) && \text{(D.18)}\\
&\leq \exp(\kappa\eta[a_i + a_j]) - 1, && \text{(D.19)}
\end{aligned}$$

thus we have

$$\begin{aligned}
w_{i,j}(t+1) \leq& w_{i,j}(t) + \left[\exp(\kappa\eta[a_i + a_j]) - 1\right] w_{i,j}(t) && \text{(D.20)}\\
&\leq w_{i,j}(0)\exp\left[\eta(t+1)(a_i + a_j)\kappa\right]. && \text{(D.21)}
\end{aligned}$$

Finally, notice that since $T_1 = \frac{\kappa \log P}{2\eta\gamma\alpha} \leq \frac{\kappa \log 2}{\eta(a_i + a_j)}$, we have

$$\exp\left[\eta T(a_i + a_j)\kappa^{-1}\right] \leq P. \tag{D.22}$$

□

Next, we prove that Lemma D.1 is tight in the initial stage of the training, up to a constant κ in the exponential term.

Lemma D.2 (Lower Bounded Initial Growth)**.** *Let $T_1 = \frac{\log P}{2\eta\gamma\alpha\kappa}$. We have for all $t \in [T_1]$, at timepoint t every entry (i,j) is in the initial phase, and the absolute value of the (i,j)-th entry satisfies*

$$|w_{i,j}(t)| \geq |w_{i,j}(0)| \exp\left[\eta t(a_i + a_j)\kappa^{-1}\right] \tag{D.23}$$

and $w_{i,j}(t)w_{i,j}(0) > 0$.

Proof. Similar to the proof of Lemma D.1, we may just assume $w_{i,j}(0) > 0$.

Moreover, we also use the claim as an inductive hypothesis and prove it by induction. Since here the inductive hypothesis states that every entry is in the initial phase, we have

$$|S_{i,j}(t) + N_{i,j}(t)| \leq 4\gamma\alpha d\beta^2\omega^2. \tag{D.24}$$

We have

$$w_{i,j}(t+1) - w_{i,j}(t) \geq \eta(a_i + a_j)w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa^{-1}\right] - 2P\eta\gamma\alpha d\beta^2\omega^2 \tag{D.25}$$

$$= w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa^{-1}\right]\left[\eta(a_i + a_j) - \frac{2P\eta\gamma\alpha d\beta^2\omega^2}{w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa^{-1}\right]}\right] \tag{D.26}$$

From Assumption D.4, we have

$$\frac{2P\eta\gamma\alpha d\beta^2\omega^2}{w_{i,j}(0)\exp\left[\eta t(a_i + a_j)\kappa^{-1}\right]} \leq 2P\eta\gamma\alpha d\beta^2\omega \tag{D.27}$$

$$\leq \left(1 - \kappa^{-1/2}\right)\eta(a_i + a_j). \tag{D.28}$$

Moreover, notice that when $\kappa > 1.1$, for any $x < 0.1$, we have $\kappa^{-1/2}x + 1 \geq e^{\kappa^{-1}x}$. Since Assumption D.2 ensured that $\eta \leq \frac{1}{10(a_i + a_j)}$, we have

$$w_{i,j}(t+1) \geq w_{i,j}(t) + w_{i,j}(t)\left[\kappa^{-1/2}\eta(a_i + a_j)\right] \tag{D.29}$$

$$\geq w_{i,j}(t)\exp\left(\eta(a_i + a_j)\kappa^{-1}\right) \tag{D.30}$$

$$\geq w_{i,j}(0)\exp\left[\eta(t+1)(a_i + a_j)\kappa^{-1}\right]. \tag{D.31}$$

Finally, from Lemma D.1, we have when

$$w_{i,j}(t) \leq |w_{i,j}(0)|\exp\left(\eta t(a_i + a_j)\kappa\right) \tag{D.32}$$

$$\leq \beta\omega\exp\left(2\eta T_1\gamma\alpha\kappa\right) \tag{D.33}$$

$$\leq P\beta\omega, \tag{D.34}$$

which confirms that every entry (i,j) stays in the initial phase before time T_1.

□

Notice that the time bound in Lemma D.2 is a uniform one which applies to all entries. For the major entries, we might want to consider a finer bound of the time that it leaves the initial phase. This can be proved by essentially repeating the same proof idea of Lemma D.2.

Lemma D.3 (Lower Bounded Initial Growth for Diagonal Entries)**.** *Consider an diagonal entry (i,i). Let $T_1^{(i)} = \frac{\log\frac{P\beta\omega}{w_{i,i}(0)}}{2\eta a_i\kappa}$. We have for all $t \in \left[T_1^{(i)}\right]$, at timepoint t the entry (i,i) is in the initial phase, and the absolute value of the (i,i)-th entry satisfies*

$$w_{i,i}(t) \geq w_{i,i}(0)\exp\left[2\eta t a_i\kappa^{-1}\right]. \tag{D.35}$$

We omit the proof of Lemma D.3 since it is almost identical to the proof of Lemma D.2, only with replacing $\gamma\alpha$ by a_i and $\beta\omega$ by $w_{i,i}(0)$.

Next, we characterize the behavior of one diagonal entry after it leaves the initial phase.

Lemma D.4 (Lower Bounded After-Initial Growth for Diagonal Entries)**.** *Consider a diagonal entry (i, i). If at time t_0 we have $|w_{i,i}(t_0)| \geq P\beta\omega$, and for a $\lambda \in (P\beta\omega, 1 - K^{-1})$, before time $T^{(\lambda)}$ we have $w_{i,i}(t + t_0) < \lambda$ for all $t \in [T^{(\lambda)}]$, then we have*

$$w_{i,i}(t + t_0) \geq w_{i,i}(t_0) \exp\left[2\eta t a_i (1 - \lambda)\kappa^{-1}\right]. \tag{D.36}$$

Moreover, $w_{i,i}(0), w_{i,i}(t_0), w_{i,i}(t_0 + t) \geq 0$.

Proof. Notice that since $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{U}^\top$ is a PSD matrix, its diagonal entries are always non-negative, this ensures that $w_{i,i}(0), w_{i,i}(t_0), w_{i,i}(t_0 + t) \geq 0$.

For the time after t_0 and before $t_0 + T^{(\lambda)}$, we use an induction to prove the claim, with the claim itself as the inductive hypothesis. It clearly holds when $t = 1$.

Notice that when $w_{i,j}(t') < \lambda$, we have

$$G_{i,j}(t') - S_{i,j}(t') = 2a_i w_{i,i}(t')\left[1 - w_{i,i}(t')\right] \geq 2a_i w_{i,i}(t')(1 - \lambda). \tag{D.37}$$

Thus we have

$$\begin{aligned}
&w_{i,i}(t_0 + t + 1) - w_{i,i}(t_0 + t) && \text{(D.38)}\\
&\geq 2\eta a_i(1 - \lambda) w_i(t_0) \exp\left[\eta t(a_i + a_j)(1 - \lambda)\kappa^{-1}\right] - 2P\eta\gamma\alpha d\beta^2\omega^2 && \text{(D.39)}\\
&= w_{i,i}(t_0) \exp\left[2\eta t a_i(1 - \lambda)\kappa^{-1}\right]\left[2\eta a_i(1 - \lambda) - \frac{2P\eta\gamma\alpha d\beta^2\omega^2}{w_{i,i}(t_0)\exp\left[2\eta t a_i(1 - \lambda)\kappa^{-1}\right]}\right] && \text{(D.40)}
\end{aligned}$$

Since $\lambda < 1 - K^{-1}$, and $w_{i,i}(t_0) \geq 2\beta\omega \geq \omega$, from Assumption D.4, we have

$$\begin{aligned}
\frac{2P\eta\gamma\alpha d\beta^2\omega^2}{w_{i,i}(t_0)\exp\left[2\eta t a_i(1 - \lambda)\kappa^{-1}\right]} &\leq 2P\eta\gamma\alpha d\beta^2\omega && \text{(D.41)}\\
&\leq 2K^{-1}\left(1 - \kappa^{-1/2}\right)\eta\alpha && \text{(D.42)}\\
&\leq 2\left(1 - \kappa^{-1/2}\right)\eta a_i(1 - \lambda). && \text{(D.43)}
\end{aligned}$$

Moreover, since Assumption D.2 ensured that $\eta \leq \frac{1}{2Ka_i(1-\lambda)} \leq \frac{1}{20a_i(1-\lambda)}$, using the fact that if $\kappa > 1.1$ then $\kappa^{-1/2}x + 1 \geq e^{\kappa^{-1}x}$ for any $x < 0.1$, we can get

$$\begin{aligned}
w_{i,i}(t + 1) &\geq w_{i,i}(t) + w_{i,i}(t)\left[\kappa^{-1/2} 2\eta a_i(1 - \lambda)\right] && \text{(D.44)}\\
&\geq w_{i,i}(t)\exp\left(2\eta a_i \kappa^{-1}(1 - \lambda)\right) && \text{(D.45)}\\
&\geq w_{i,i}(t_0)\exp\left[2\eta(t + 1)\kappa^{-1}(1 - \lambda)\right]. && \text{(D.46)}
\end{aligned}$$

□

Next, we provide an uniform upper bound (over time) of the diagonal entries. Remember that we mentioned in the gradient flow case, the diagonal term stops evolving when it reaches 1. In the discrete case, since the step size is not infinitesimal, Lemma D.5 shows that it can actually exceed 1 a little bit but not too much since the step size is small.

Lemma D.5 (Upper Bounded Diagonal Entry)**.** *For any diagonal entry (i, i) and any time t, $0 \leq w_{i,i}(t) \leq 1 + 2K^{-1}$.*

Proof. First notice that since $\boldsymbol{W}(t)$ is PSD, its diagonal entry $w_{i,i}(t)$ should always be non-negative, thus $w_{i,i}(t) \geq 0$ is always satisfied. In the following we prove $w_{i,i}(t) \leq 1 + 2K^{-1}$.

We use induction to prove this claim. The inductive hypothesis is the claim it self. It is obviously satisfied at initialization. In the following we assume the claim is satisfied at timepoint t and prove it for timepoint $t + 1$. Notice that since $K \leq 10$, we have $1 + K^{-1} \leq 2$.

Notice that by Assertion D.1 and Assumption D.4,

$$|N_{i,i}(t)| \leq 2P\gamma\alpha d\beta^2\omega^2 \leq \frac{(\kappa - 1)^2}{K^2\gamma d\beta^2}\alpha \leq K^{-1}a_i. \tag{D.47}$$

If $w_{i,i}(t) \geq 1 + K^{-1}$, we have

$$G_{i,i}(t) - S_{i,i}(t) = 2a_i w_{i,i}(1 - w_{i,i}) \leq -4a_i K^{-1}. \tag{D.48}$$

Therefore,

$$\begin{aligned} w_{i,i}(t+1) &= w_{i,i}(t) + \eta \left[G_{i,i}(t) - S_{i,i}(t) - N_{i,i}(t)\right] && \text{(D.49)} \\ &\leq w_{i,i}(t) - 3a_i K^{-1}\eta && \text{(D.50)} \\ &\leq w_{i,i}(t) && \text{(D.51)} \\ &\leq 1 + 2K^{-1}. && \text{(D.52)} \end{aligned}$$

Moreover, since $w_{i,i}(t) \leq 1 + 2K^{-1} \leq 2$, we have

$$|G_{i,i}(t)| + |S_{i,i}(t)| + |N_{i,i}(t)| \leq 4a_i + 4a_i + K^{-1}a_i \leq 9\gamma\alpha \leq \frac{1}{K\eta}. \tag{D.53}$$

When $w_{i,i}(t) \leq 1 + K^{-1}$, we have

$$w_{i,i}(t+1) \leq w_{i,i}(t) + \eta\left(|G_{i,i}(t)| + |S_{i,i}(t)| + |N_{i,i}(t)|\right) \leq 1 + 2K^{-1}. \tag{D.54}$$

The above results together shows that $w_{i,i}(t+1) \leq 1 + 2K^{-1}$.

□

Corollary D.2 (Upper Bounded Diagonal Update)**.** *For any diagonal entry* (i,i) *and any time* t*,* $|w_{i,i}(t+1) - w_{i,i}(t)| \leq K^{-1}$.

Corollary D.2 is a direct consequence of Lemma D.5 (and we actually proved Corollary D.2 in the proof of Lemma D.5).

The next lemma lower bounds the final value of diagonal entries. Together with Lemma D.5 we show that in the terminal stage of training the diagonal entries oscillate around 1 by the amplitude not exceeding $2K^{-1}$.

Lemma D.6. *Consider a diagonal entry* (i,i)*. If at time* t_0 *we have* $w_{i,i}(t_0) \geq 1 - 2K^{-1}$*, then for all* $t' \geq t_0$ *we have* $w_{i,i}(t') \geq 1 - 2K^{-1}$.

Proof. we use an induction. The inductive hypothesis the claim itself. This obviously holds when $t' = t_0$. We assume $w_{i,i}(t') \geq 1 - 2K^{-1}$ at timepoint t' and prove the claim for $t' + 1$.

If $w_{i,i}(t') < 1 - K^{-1}$, then from Lemma D.4 we know

$$w_{i,i}(t'+1) \geq w_{i,i}(t') \geq 1 - 2K^{-1}. \tag{D.55}$$

If $w_{i,i}(t') > 1 - K^{-1}$, then from Corollary D.2 we have

$$w_{i,i}(t'+1) \geq w_{i,i}(t') - K^{-1} \geq 1 - 2K^{-1}. \tag{D.56}$$

□

Now, we are ready to prove Assertion D.1 by considering the suppression. We first prove a lemma that upper bounds the absolute value of the minor entries after its corresponding major entry becomes significant.

Lemma D.7 (Suppression)**.** *Consider an off-diagonal entry* (i,j) *where* $i > j$*. If there exists a time* t_0 *such that* $w_{i,i}(t_0) > 0.8$*, then for any* $t' \geq t_0$ *we have*

$$|w_{i,j}(t')| \leq \max\left\{|w_{i,j}(t_0)|, \omega\right\}. \tag{D.57}$$

Proof. Since $K > 10$, from Lemma D.6 and Lemma D.4 we know $w_{i,i}(t') > 0.8$ for all $t' \geq t_0$.

In this proof, we use an induction with the inductive hypothesis being the claim itself, i.e., we assume the claim is true at timepoint t' and prove it for $t' + 1$. The claim obviously holds for $t' = t_0$.

Since in this proof we only use the absolute value of $N_{i,j}$, WLOG we may assume that $w_{i,j}(t') > 0$.

If $w_{i,j}(t') < \omega$ then we have proved the claim. In the following we may assume $w_{i,j}(t') \geq \omega$.

We have

$$\begin{aligned} G_{i,j}(t') - S_{i,j}(t') &\leq w_{i,j}(t')(a_i + a_j) - \frac{1}{2} w_{i,j}(t') w_{i,i}(3a_i + a_j) && \text{(D.58)} \\ &\leq w_{i,j}(t')(a_i + a_j) - w_{i,j}(t') \left[0.4(3a_i + a_j)\right] && \text{(D.59)} \\ &= -\frac{1}{5} w_{i,j}(t') a_i + \frac{3}{5} w_{i,j}(t') a_j && \text{(D.60)} \\ &\overset{\text{(i)}}{\leq} -C^{-1}\omega\alpha, && \text{(D.61)} \end{aligned}$$

where in (i) we use Assumption D.5.

Thus we have

$$\begin{aligned} G_{i,j}(t') - S_{i,j}(t') - N_{i,j}(t') &\leq G_{i,j}(t') - S_{i,j}(t') + |N_{i,j}(t')| && \text{(D.62)} \\ &\leq -C^{-1}\omega\alpha + 2P\gamma\alpha d\beta^2\omega^2 && \text{(D.63)} \\ &\overset{(i)}{<} 0, && \text{(D.64)} \end{aligned}$$

where (i) is from Assumption D.4 and Assumption D.5. This confirms that $w_{i,j}(t'+1) < w_{i,j}(t') \leq \max\{|w_{i,j}(t_0), \omega\}$.

Next, we prove $w_{i,j}(t'+1) \geq -\max\{|w_{i,j}(t_0)|, \omega\}$. Notice that Lemma D.5 stated that $|w_{i,i}| \leq 2$. Notice that we also have $w_{i,j}(t') \leq K^{-1}$, thus

$$\begin{aligned} |G_{i,j}(t')| + |S_{i,j}(t')| + |N_{i,j}(t')| &\leq 10\gamma\alpha |w_{i,j}(t')| + 2P\gamma\alpha d\beta^2\omega^2 && \text{(D.65)} \\ &\leq \frac{10|w_{i,j}(t')| + 2Pd\beta^2\omega^2}{9K\eta} && \text{(D.66)} \\ &\leq \frac{10|w_{i,j}(t')| + 2\omega}{9K\eta} && \text{(D.67)} \\ &\leq \frac{|w_{i,j}(t')| + \omega}{2\eta}. && \text{(D.68)} \end{aligned}$$

We have

$$\begin{aligned} w_{i,j}(t'+1) &\geq w_{i,j}(t') - \eta(|G_{i,j}(t')| + |S_{i,j}(t')| + |N_{i,j}(t)'|) && \text{(D.69)} \\ &\geq -\eta(|G_{i,j}(t')| + |S_{i,j}(t')| + |N_{i,j}(t')|) && \text{(D.70)} \\ &\geq -\frac{1}{2}(|w_{i,j}(t')| + \omega) && \text{(D.71)} \\ &\geq -\max\{|w_{i,j}(t')|, \omega\}. && \text{(D.72)} \end{aligned}$$

□

With all the lemmas proved above, we are now ready to prove Assertion D.1.

Lemma D.8 (Assertion D.1)**.** *For all $t \in \mathbb{N}$, if $i \neq j$, then the entry (i, j) stays in the initial phase for all time.*

Proof. Notice that since $\boldsymbol{W}$ is symmetric, we only need to prove the claim for $i > j$. Moreover, From Lemma D.7, we only need to prove that there exists a timepoint t^*, such that $w_{i,i}(t^*) \geq 0.8$, and $|w_{i,j}(t^*)| \leq P\beta\omega$.

Let $t_0 = \frac{\log \frac{P\beta\omega}{w_{i,i}(0)}}{2\eta a_i \kappa}$, by Lemma D.3, we have $w_{i,i}(t_0) \geq P\beta\omega$. By Lemma D.3 and Lemma D.4, we have for any $t \geq t_0$ such that $w_{i,i}(t) \leq \lambda$, where $\lambda = 0.85$,

$$\begin{aligned} w_{i,i}(t) &\geq w_{i,i}(t_0) \exp\left[0.3\eta(t - t_0) a_i \kappa^{-1}\right] && \text{(D.73)} \\ &\geq P\beta\omega \exp\left[0.3\eta(t - t_0) a_i \kappa^{-1}\right] && \text{(D.74)} \end{aligned}$$

Let t' be the first time that $w_{i,i}(t')$ arrives above 0.8. Let $t^* = \min\left\{\frac{\kappa \log \frac{0.8}{P\beta\omega}}{0.3\eta a_i} + t_0, t'\right\} \geq t_0$. If $t^* = t'$, we have $w_{i,i}(t^*) \geq 0.8$. If $t^* = \frac{\log \frac{P\beta\omega}{w_{i,i}(0)}}{2\eta a_i \kappa} + t_0$, we have

$$\begin{aligned} w_{i,i}(t^*) &\geq w_{i,i}(0) \exp\left(0.3\eta t^* a_i \kappa^{-1}\right) && \text{(D.75)} \\ &\geq P\beta\omega \exp\left(\log \frac{0.8}{P\beta\omega}\right) && \text{(D.76)} \\ &\geq 0.8. && \text{(D.77)} \end{aligned}$$

Moreover, from Lemma D.1 and Assumption D.5, we have

$$\begin{aligned} |w_{i,j}|(t^*) &\leq |w_{i,j}(0)| \exp\left[\eta t^* \kappa (a_i + a_j)\right] && \text{(D.78)} \\ &\leq \beta\omega \exp\left[\left(\frac{\kappa^2 \log \frac{0.8}{P\beta\omega}}{0.15} + \log \frac{P\beta\omega}{w_{i,i}(0)}\right) \times \frac{a_i + a_j}{2a_i}\right] && \text{(D.79)} \\ &\leq \beta\omega \exp\left[\left(10\kappa^2 \log \frac{1}{P\beta\omega} + \log P\beta\right) \times \frac{a_i + a_j}{2a_i}\right] && \text{(D.80)} \\ &\leq \beta\omega \exp\left[\log(P)\right] && \text{(D.81)} \\ &\leq P\omega\beta. && \text{(D.82)} \end{aligned}$$

The claim is thus proved by combining the above bounds on $|w_{i,j}(t^*)|$ and $w_{i,i}(t^*)$ with Lemma D.7. □

E Additional Discussions

In this section, we further discuss the findings and theoretical predictions presented in this paper.

E.1 Multiple Descents

In Fig. 5, we verified our theoretical predictions of the Transient Memorization through an experiment of an $s = 2$ example. However, in our theory, there can be multiple growth / suppression stages when $s > 2$, which should give us a multiple descent-like curve. We note here that based on the conditions given in App. D.1, it is indeed possible to see multiple descent but only with a subtle choice of the signal strengths ($\boldsymbol{\mu}$) and under specific initialization conditions.

In Fig. 8 and 9, we illustrate two settings where the loss curves exhibit epochwise triple and quadruple descent. In both settings we use symmetric 2-layer linear model, same as the model used in Sec. 4.2. Note that we tuned initialization random seed to generate these results. Moreover, since the time scale of each descent vary, we use a log scale for the number of epochs to make the results more apparent.

It is worth noting that in Fig. 8 and 9, each major entry starts to grow only after the corresponding minor entry is suppressed (for example in Fig. 8, $w_{2,2}$ starts to grow after $w_{2,3}$ is suppressed, $w_{1,2}$ starts to decay after $w_{2,2}$ is close to 1, and $w_{1,1}$ starts to grow after $w_{1,2}$ is suppressed), and each ascending / descending stage of loss curve aligns well with a stage of growth / suppression of the minor and major entries. These correspondence match exactly with our theoretical prediction and shows the correctness and preciseness of our theory.

E.2 The Breakdown of the Initialization Assumption

In Sec. 3.3, we mentioned that Transient Memorization seems to be more significant when the dimensionality of the dataset is low, and in Sec. 4.2.2, we attributed the reason of it to the fact that when the dimensionality of the dataset is small, it's easier to have more minor entries initialized positive, which lead to an illusion of learning in the minor entry growth stage, whose later suppression leads to the non-monotonic output trajectory behavior of Transient Memorization.

In this section, we note that, another reason for the Transient Memorization to be less significant is that Assumption D.4 breaks down when the dimension is high, if we use standard Gaussian initialization to initialize the model weights. Specifically, in Assumption D.4, we require that all entries of

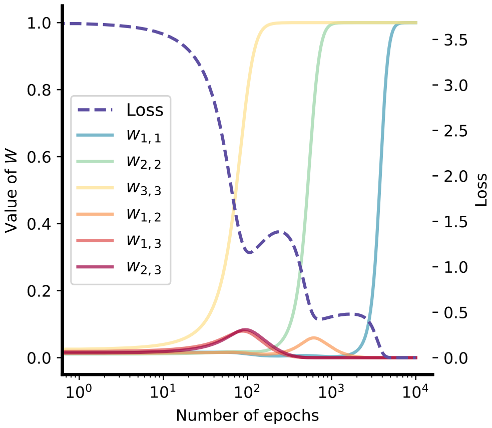

Figure 8: **An illustration of epochwise triple descent of the symmetric 2-layer linear model.** The dataset has dimensionality and number of informative directions $d = s = 3$, signal strength values $\boldsymbol{\mu} = (1.0, 1.5, 2.2)$, and noise values $\boldsymbol{\sigma} = (0.05, 0.05, 0.05)$.

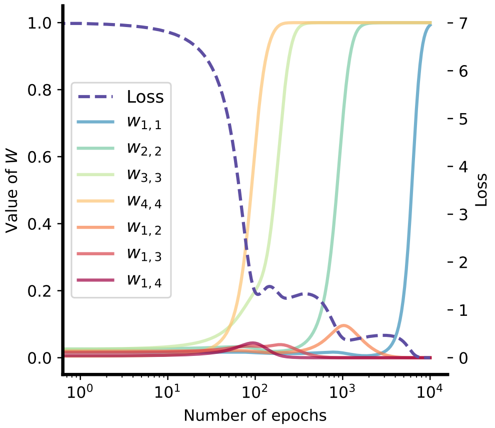

Figure 9: **An illustration of epochwise quadruple descent with the symmetric 2-layer linear model.** The dataset has dimensionality and number of informative directions $d = s = 4$, signal strength values $\boldsymbol{\mu} = (1.0, 1.5, 2.2, 2.7)$, and noise values $\boldsymbol{\sigma} = (0.05, 0.05, 0.05, 0.5)$.

$\boldsymbol{W}$ are initialized around a relatively small value ω, which indicates that there is no huge difference between the magnitude of the initialization of major entries and minor entries.

However, notice that $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{U}^\top$ and thus $w_{i,j} = \langle \boldsymbol{u} - i, \boldsymbol{u}_j \rangle$, where $\boldsymbol{u}_i \in \mathbb{R}^{d'}$ is the i-th row of $\boldsymbol{U}$. If we use Gaussian distribution to initialize $\boldsymbol{U}$, i.e. $\boldsymbol{u}_i \sim \mathcal{N}\left(\boldsymbol{0}, \tau^2 \boldsymbol{I}\right)$, where τ is a small real number, then we have the expectation of $w_{i,j}$ be

$$\mathbb{E} w_{i,j} = \begin{cases} 0 & i \neq j \\ d'\tau^2 & i = j, \end{cases} \tag{E.1}$$

which highlights the different between major entries and minor entries in initialization when d' is large (and notice that d' is lower bounded by d, which is the dataset dimensionality). Moreover, when d' is small, the variance of $w_{i,j}$ will be large, so there is a greater chance for them to be away from 0.

E.3 Failure Modes

A breakdown in the assumptions in App. D.1 can also lead to the model converging to “wrong” solutions that do not fully generalize OOD. For example, if a minor entry happens to be initialized too large (breaking the Assumption D.4), and / or the corresponding signal strength distinction is not large enough (breaking the Assumption D.5), then it is possible that the minor entry is not suppressed until it grows to a significant value, which can, in turn, lead to a too strong suppression on the corresponding major entry. In this case, a major entry might be suppressed to 0 (or at least, leave the initial phase from below) before it starts to grow, and thus never has chance to grow. This case corresponds to the model being “trapped” in a state that it only learns to compositionally generalize to a combination of certain (but not all) concepts.

In Fig. 10, we exhibit a case of such failure mode where the model fails to fully achieve OOD generalization. Notice how the loss value converges to a non-zero value and the major entry $w_{1,1}$ is suppressed at the very beginning and never grows. Additionally, the output trajectory is trapped at a point that combines only two directions, missing the third direction.

E.4 Future Directions

We note that, current characterization of the model learning dynamics relies on the critical assumptions in App. D.1. Although those assumptions are reasonable and common in practice, the model

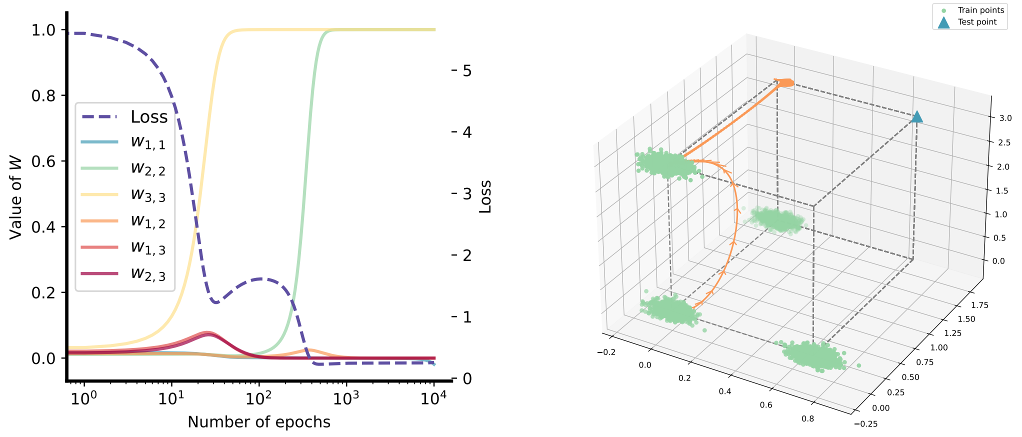

Figure 10: **An illustration of a failure case where the model doesn't successfully generalize OOD.** The dataset has $d = s = 3$, $\boldsymbol{\mu} = (0.7, 1.7, 3)$ and $\boldsymbol{\sigma} = (0.05, 0.05, 0.05)$. Left: The evolution of values of Jacobian and the OOD loss evaluated at test point $\hat{\boldsymbol{x}}$; Right: The output trajectory (orange curve).

behavior still shows some regularity when those assumptions breakdown. We have discussed some of the possible consequences when one of those assumptions breakdown above, but more in an intuitive way, instead of a systematic way. Therefore, one important future direction is to systematically characterize what will happen beyond the assumptions given in App. D.1. Among which, one specific and very important topic is the failure modes, i.e. under what conditions the model fails to generalize OOD.

Another important direction is to generalize current analysis to more complex models, such as deep linear networks or two-layer ReLU networks. The key point of current analysis of the 2-layer symmetric model is to correctly slice the learning dynamics of each entry of the Jacobian into multiple stages, such that in each stage, the learning dynamics is dominated by a rather simple dynamics.

Currently, since the model is 2-layer and without bias terms, there are only first-order and second-order terms in the learning dynamics. However, if we consider deeper models, there might be higher-order terms in the dynamics, and it is important to identify and simplify the effect these higher-order interactions in order to make the problem tractable.

For ReLU networks, it is know that there will be an "early-alignment" stage when trained on linear-separable data [48, 50], where each neuron converge to a fixed direction, and make the model behaves like a linear model. We claim that investigating the early-alignment of 2-layer ReLU networks on the SIM task can be the starting point of theoretically characterizing the behavior of ReLU networks on the SIM task.

F Additional SIM Experiment Details and Results

In this section, we present the results of SIM experiments under different settings, including linear and non-linear models. The consistent behavior observed across these settings confirms the universality of our findings and explanations.

F.1 Experiment Details

In all SIM experiments, including those presented in main paper and in appendix, the number of training samples in each Gaussian cluster is 5000. We use MLP models with either linear activations or ReLU activations, and all the models are trained using stochastic gradient descent with a batch size of 128 and a learning rate of 0.1 for 40 epochs. Unless otherwise specified, the dimensionality of all data points is $d = 64$, and the hidden layer dimensionality of the models is also 64 by default.

It is important to note that in our theory, we assumed that all training clusters and the test point are aligned with the standard coordinate. However, in our experiments, in order to make the results more universal and general, we add a random rotation to all the train / test points.

F.2 Additional Experiment Results

Fig. 11 and Fig. 12 repeat the learning order experiments described in Sec. 3.1, using a 2-layer model with and without ReLU activation, respectively. It is easy to see that despite showing more non-regular curves, in multi-layer models the overall trends described in Sec. 3.1 and Sec. 3.2 are preserved.

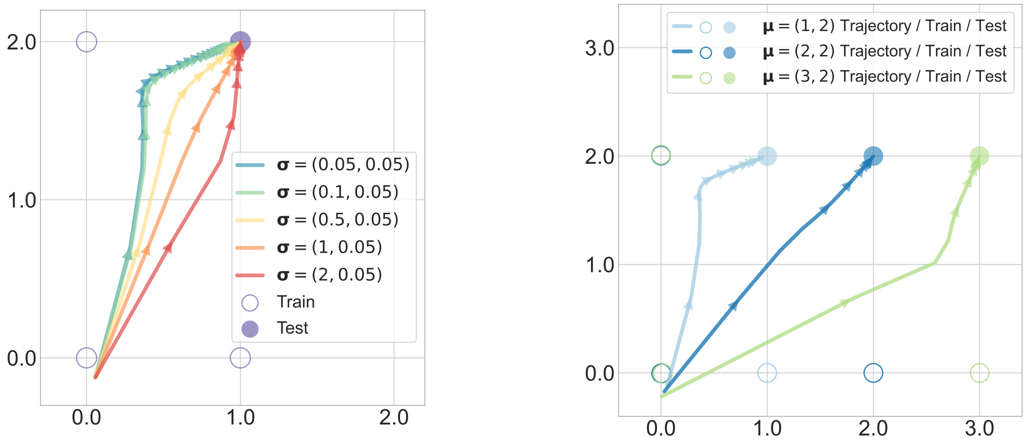

Figure 11: **Output trajectory of 2-layer models with linear activations**. The number of informative directions in the dataset is $s = 2$. Left: $\boldsymbol{\mu}_{:2} = (1, 2)$ with varied $\boldsymbol{\sigma}$'s; Right: $\boldsymbol{\sigma}_{:2} = (0.05, 0.05)$ with varied $\boldsymbol{\mu}$'s.

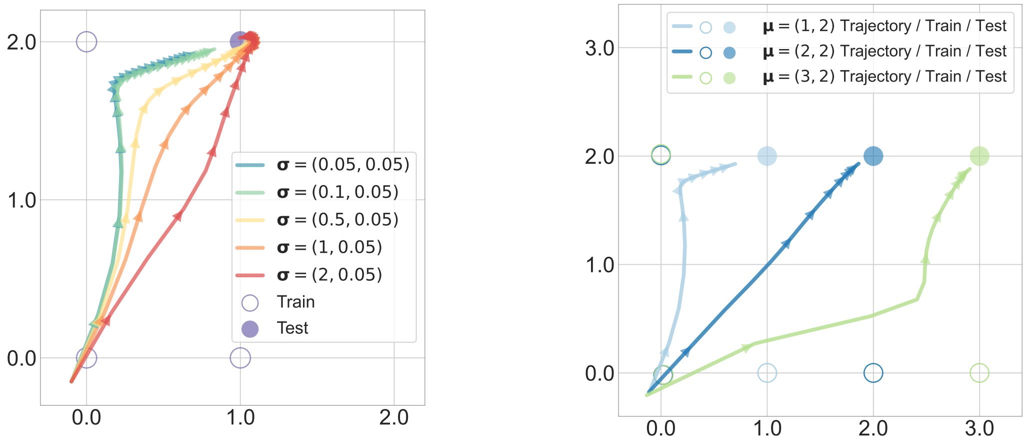

Figure 12: **Output trajectory of 2-layer models with ReLU activations**. The number of informative directions in the dataset is $s = 2$. Left: $\boldsymbol{\mu}_{:2} = (1, 2)$ with varied $\boldsymbol{\sigma}$ values; Right: $\boldsymbol{\sigma}_{:2} = (0.05, 0.05)$ with varied $\boldsymbol{\mu}$ values.

In Fig. 14, we present the output trajectory for two settings that exhibit significant Transient Memorization and Fig. 14 the corresponding loss curve. Specifically, the dataset has a dimensionality of $d = 3$, and is not randomly rotated. The models used have 3 layers, 3 hidden dimensions and linear activations. Comparing these results with the curve presented in Fig. 11, Fig. 12 and in Fig. 2, it is evident that models with more layers and fewer input dimensions are easier to have Transient Memorization, which confirms our theoretical prediction in Sec. 4.2.2.

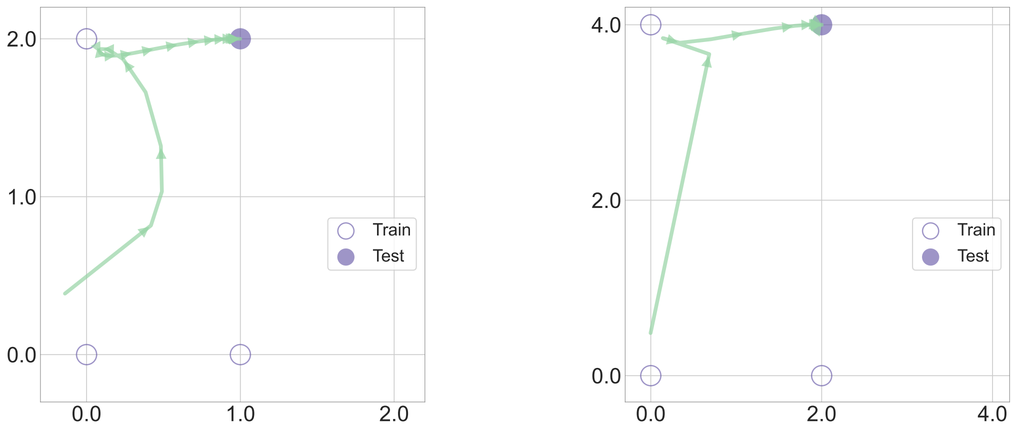

Figure 13: **Output trajectory of 3-layer models with linear activations and 3 hidden dimensions.** The dataset has dimensionality $d = 3$, number of informative directions $s = 2$ and variance $\boldsymbol{\sigma}_{:2} = (0.05, 0.05)$. Left: $\boldsymbol{\mu}_{:2} = (1, 2)$; Right: $\boldsymbol{\mu}_{:2} = (2, 4)$.

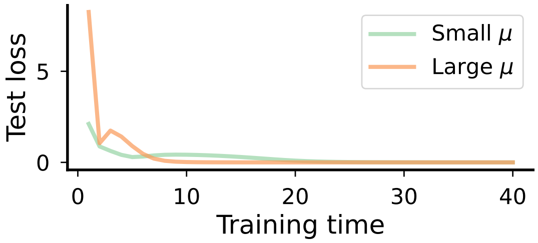

Figure 14: **The loss curve of corresponding models in Fig. 13.** Small μ: $\boldsymbol{\mu}_{:2} = (1, 2)$; Large μ: $\boldsymbol{\mu}_{:2} = (2, 4)$.

G Diffusion Model Experiments

We describe experimental details for the diffusion model experiments. We largely follow [54] in these experiments.

G.1 Synthetic Data

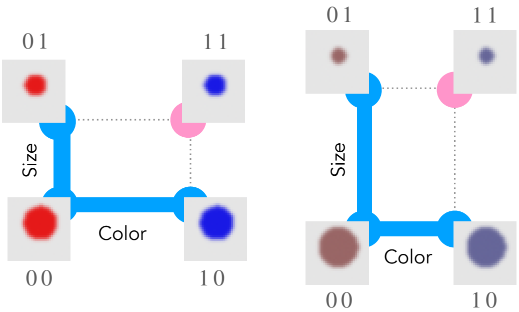

Figure 15: **Data Generation process with different concept signals.** Figure from [54]. The data generating processed used to train the diffusion model, where we can control the strength of the two concept signals independently. Left: A data distribution with a stronger concept signal in the `color` dimension. Right: A data distribution with a stronger concept signal in `size`.

Fig. 15 illustrates the DGP. We borrow part of the compositional data generating process (DGP) introduced by in [54]. The DGP generates a set of images of circles based on the *concept vari-*

ables color={red, blue} and size={big, small}. Each concept variable can be selected as composed to yield four classes 00, 01, 10, 11 respectively corresponding to (red, big), (red, small), (blue, big), (blue, small). Here, class average pixels values of red and blue colors will control the concept signal for color and the difference between small and big. In Fig. 6, we fix the big circle's diameter to 70% of the image and the small circle's diameter to 30% of the image. We then adjust the absolute difference between the blue color and red color from 0.2 (very similar colors) to 0.7 (very different colors). The DGP randomizes the location of the circle, the background color and adds some noise to avoid having a very narrow data distribution. Please refer to [54] for further detail.

In Fig. 15, we show two different data distributions, one with a big color concept signal and one with a big size concept signal.

G.2 Model & Training

We train a conditional diffusion model on the synthetic data defined above. In specific, we train a variational diffusion model [35] to generate $3 \times 32 \times 32$ images conditioned on a 4-dimensional vector where the first element of the vector specifies the size of the circle and the 3 others specifies the RGB colors.

Model Architecture We use a conditional U-Net [63] with hidden dimensions $[64, 128, 256]$ before each downsampling layer and two ResNet [25] layers in each level. The conditioning vector is first transformed into the same dimensions as the hidden dimensions using a 2-layer MLP and are added to the representation after every downsampling layer. The U-Net has a self attention layer [11] in its bottleneck. We used LayerNorm [6] for normalization layers and GELU [27] activations.

Diffusion We use a learned linear noise schedule for the diffusion process as defined in [35], initialized with $\gamma_{\max} = 10,\ \gamma_{\min} = -5$. We assume a data noise of 1×10^{-3}. Variational diffusion models do not require fixing the number of diffusion steps at training time, but we use 100 steps for generation at inference time.

Training We train our model with the AdamW optimizer [46] with learning rate 1×10^{-3} and weight decay 0.01. We use a batch size of 128 and train for 20k steps.

G.3 Evaluation

We evaluate the concept space representation of the generated output image using a trained classifier. Since we have the ground truth DGP, we used a large amount of data to train a perfect classifier. We used a U-Net backbone followed by a max pooling layer and a MLP classifier to classify each concept variable color and size. We train this classifier for 10k steps and achieve a 100% accuracy on a held out test set. We average over 32 generated images and 5 model run seeds to get the ensemble average concept space representation.

The concept space MSE in Fig. 6 (b) is simply calculated as the MSE distance in the concept space defined in [54]. The concept learning speed $|\mathrm{d}C/\mathrm{d}t|$ is quantified by estimating the movement speed in the same concept space by a finite difference method.