Proceedings of Machine Learning Research vol 291:1–80, 2025 38th Annual Conference on Learning Theory

Learning Compositional Functions with Transformers from Easy-to-Hard Data

Zixuan Wang* WANGZX@PRINCETON.EDU
Princeton University

Eshaan Nichani* ESHNICH@PRINCETON.EDU
Princeton University

Alberto Bietti ABIETTI@FLATIRONINSTITUTE.ORG
Flatiron Institute

Alex Damian AD27@PRINCETON.EDU
Princeton University

Daniel Hsu DJHSU@CS.COLUMBIA.EDU
Columbia University

Jason D. Lee JASONLEE@PRINCETON.EDU
Princeton University

Denny Wu DENNYWU@NYU.EDU
Flatiron Institute, New York University

Editors: Nika Haghtalab and Ankur Moitra

Abstract

Transformer-based language models have demonstrated impressive capabilities across a range of complex reasoning tasks. Prior theoretical work exploring the expressive power of transformers has shown that they can efficiently perform multi-step reasoning tasks involving parallelizable computations. However, the learnability of such constructions, particularly the conditions on the data distribution that enable efficient learning via SGD, remains an open question. Towards answering this question, we study the learnability of a task called the k-*fold composition*, which requires computing an interleaved composition of k input permutations and k hidden permutations, and can be expressed by a transformer with $O(\log k)$ layers. On the negative front, we provide a Statistical Query lower bound showing that any learner which is trained on samples from the k-fold composition task and makes polynomially many queries must have sample size exponential in k, thus establishing a statistical-computational gap. On the other hand, we show that this function class can be efficiently learned, with runtime and sample complexity polynomial in k, by gradient descent on an $O(\log k)$-depth transformer via two different curriculum learning strategies: one in which data consists of k'-fold composition functions with $k' \leq k$ presented in increasing order of difficulty, and another in which all data is presented simultaneously. Our work sheds light on the necessity and sufficiency of having both easy and hard examples in the data distribution for transformers to learn complex compositional tasks.

Keywords: transformers, training dynamics, multi-step reasoning, curriculum learning

* Equal contribution.

1. Introduction

Large language models based on transformers have demonstrated promising capabilities in complex reasoning tasks that require combining multiple intermediate steps (Nye et al., 2021; Wei et al., 2022; Lewkowycz et al., 2022; Lanchantin et al., 2024; Yao et al., 2024). Recent theoretical works have proven that transformers can *express* various sequential/compositional reasoning algorithms (Liu et al., 2023; Merrill and Sabharwal, 2023a; Li et al., 2024; Feng et al., 2024; Sanford et al., 2024a). However, representational power does not entail statistical or optimization efficiency. In fact, empirical studies have shown that elaborate training procedures – such as curriculum or process supervision (Uesato et al., 2022; Lightman et al., 2023; Dziri et al., 2023; Bachmann and Nagarajan, 2024; Deng et al., 2024) – are often required for models to acquire strong reasoning capabilities. This highlights the need for a deeper theoretical understanding of the optimization and sample efficiency of transformers on compositional reasoning tasks.

Our starting point is the recent work Sanford et al. (2024b), which examined the expressivity of transformers on a specific reasoning task called the "k-hop induction head", which involves composing k steps of pointer-following given in the context to predict the correct answer. This function is a generalization of the *induction head* identified by Olsson et al. (2022) ($k = 1$), and intuitively, the difficulty of computing the function increases with k. Sanford et al. (2024b) showed that a transformer with $\Theta(\log k)$ layers can efficiently represent this task and that this depth is necessary, conditional on a well-known conjecture from the *Massively Parallel Computation* literature (Im et al., 2023). Furthermore, the authors empirically observed that gradient-based learning is challenging unless some form of curriculum (i.e., including lower-hop data during training) is introduced.

The k-hop task (for $k \geq 2$) is closely related to the function composition tasks studied by Peng et al. (2024) and Chen et al. (2024a), which were motivated by certain natural language understanding tasks (e.g., finding an ancestor of a person in a genealogy), as well as "multi-hop" reasoning problems studied extensively in the natural language literature, even before transformers were introduced (e.g., Weston et al., 2014, 2015; Sukhbaatar et al., 2015). An example of a 2-hop reasoning problem due to Weston et al. (2014) is as follows: *John plays football. The football game is on Sunday. On what day does John play?* Here, a model must learn to compose the relationships $(\text{John} \to \text{Football}, \text{Football} \to \text{Sunday})$ present in the context.

More challenging compositional reasoning tasks can also require composing information present in the context (i.e "contextual knowledge") with global "parametric knowledge" not provided in the context (Cheng et al., 2024; Yang et al., 2024). Consider instead the prompt: *John plays quarterback. The football game is on Sunday. On what day does John play?* Now, the model must compose the contextual knowledge $(\text{John} \to \text{Quarterback}, \text{Football} \to \text{Sunday})$ given in the prompt, with the parametric knowledge $(\text{Quarterback} \to \text{Football})$ which cannot be inferred from context alone.

Our contributions. We analyze the complexity of training a (deep) transformer model using SGD to solve a task involving k-hop compositional reasoning, which we refer to as the *k-fold composition* task. The task requires outputting an element of $[N]$ after applying an interleaved product of k *in-context* permutations on N elements and k *hidden parametric* permutations, and can be viewed as an instance of k-hop prediction combining contextual and hidden parametric knowledge. We aim to establish sample complexity upper bounds for gradient-based learning as well as computational lower bounds for this function class. More specifically, our contributions are as follows:

1. **k-fold composition task.** In Section 2.1 we introduce the k-fold composition task, which requires computing an interleaved composition of k input permutations and k hidden permutations on N elements. In Theorem 4 we prove that this task is expressible via an $O(\log k)$ depth transformer with embedding dimension $d = \tilde{O}(Nk)$.

2. **Lower bound.** In Section 4, we establish a statistical query (SQ) lower bound showing that either $N^{\Omega(k)}$ queries or a tolerance of $\tau \leq N^{-\Omega(k)}$ is required to learn the k-fold composition task when trained on samples only from the k-fold functions. Using the standard $\tau \approx n^{-1/2}$ heuristic, this implies that any SQ learner (which can model gradient descent on neural networks) must either have sample size or runtime exponential in k.

3. **Gradient-based learning.** On the other hand, in Section 5 we show that a transformer *can* efficiently learn the task when training on easy-to-hard data consisting of k'-fold functions for $k' \leq k$. In Theorem 7, we show that if the transformer is presented with a curriculum of 2^ℓ-fold data for increasing values of ℓ, then gradient descent can learn the k-fold task with $\mathrm{poly}(N, k)$ samples, which removes the exponential dependence on k in the SQ lower bound. In Theorem 8, we show that this efficient learning guarantee also applies to simultaneously training on a mixture of the 2^ℓ-fold data.

1.1. Related Work: Transformer Theory

Expressivity of transformers. As already alluded to previously, many prior works have connected the computational power of transformers to models of parallel computation, including circuit models (Liu et al., 2023; Merrill and Sabharwal, 2023b) and massively parallel computation (Sanford et al., 2024b,a). This stands in contrast to sequential neural architectures such as recurrent neural networks, which are unable to efficiently represent certain parallel computations that transformers can (Sanford et al., 2023; Bhattamishra et al., 2024; Jelassi et al., 2024). As for negative results on the expressivity of compositional tasks, Peng et al. (2024) showed the composition of two functions cannot be efficiently represented by one-layer transformers (even in an average-case sense), and Chen et al. (2024a) showed that, for any constant L, compositions of L functions cannot be efficiently represented by L-layer decoder-only transformers. Since we consider encoder, and not decoder-only, transformers, the lower bound of Chen et al. (2024a) does not apply to our setting.

Optimization guarantees for transformers. A number of prior works have studied the gradient descent dynamics of transformers for various synthetic tasks (Jelassi et al., 2022; Li et al., 2023; Bietti et al., 2023; Tian et al., 2023; Zhang et al., 2023; Huang et al., 2023; Nichani et al., 2024a,b; Ren et al., 2024; Wang et al., 2024; Chen et al., 2024c,b; Huang et al., 2025). These works, however, only focus on one or two-layer transformers. While existing gradient flow or landscape results do exist for deeper transformers (Ahn et al., 2024; Gao et al., 2024), we are the first to provide an end-to-end optimization and statistical guarantee for a deep transformer.

Among existing optimization guarantees for transformers, to our knowledge, the only setting that considers a compositional structure is the parity problem (Kim and Suzuki, 2024; Wen et al., 2024). That said, while the task decomposition (i.e., curriculum) for parity exhibits a hierarchical structure, the target function itself is not inherently compositional and can be represented by a shallow network; Abbe et al. (2023); Panigrahi et al. (2024) have theoretically studied the benefit of curriculum learning for learning parities with shallow neural networks. Our goal is to analyze the gradient-based

learning of a more challenging compositional task that fundamentally requires a deeper transformer architecture.

1.2. Related Work: Compositional Tasks

Our k-fold composition task is most similar to the k-hop induction head task in Sanford et al. (2024b), which takes as input a sequence of T tokens $X \in \Sigma^T$, and requires outputting a k-fold composition of a certain "hop" function defined as an in-context map from $[T] \to [T]$ by a similar mechanism to the induction head (Olsson et al., 2022). The permutation composition task also computes such a composition, but where each hop is specified explicitly (from $(i, j) \in [k] \times [N]$ to $(i-1, \sigma_i(j))$) rather than needing to be learned in context. The k-hop induction head task is closely related to the well-studied pointer-chasing problem (Papadimitriou and Sipser, 1984; Nisan and Wigderson, 1993), for which communication complexity lower bounds have been established in various settings (e.g., Yehudayoff, 2020; Assadi and N, 2021). Furthermore, our k-fold composition task is an instance of computing an interleaved group product, a problem also studied in communication complexity (Gowers and Viola, 2019). However, in neither setting has the question of learnability been investigated.

Our task is also related to the semiautomaton simulation task of Liu et al. (2023), which takes as input a sequence of elements $(g_1, \ldots, g_T)$ from some semigroup, and outputs their product $g_T g_{T-1} \cdots g_2 g_1$. Indeed, Liu et al. (2023) showed that there exists an $O(\log T)$ depth transformer which can solve the task. Our k-fold composition task can be viewed as a special case of this task for the symmetric group $G = S_N$. However, our approach differs in that we encode each permutation σ_i as N tokens $(\sigma_i(1), \ldots, \sigma_i(N))$ rather than a single token, and consider the learning of a function class formed by interleaving the hidden permutations $(\pi_1, \ldots, \pi_k)$ into the product.

Another relevant compositional problem is "planning in path-star graphs" introduced by Bachmann and Nagarajan (2024) to show the limitation of learning with next-token prediction. Their task involves finding a path to a final node in a star graph, when the edges of the graph are given in the context. The first node in the path is then essentially a k-hop prediction starting from the final node, where k is the length of the paths. The authors discussed the difficulty of this k-hop prediction, and the benefits of first predicting intermediate steps in the path, which is related to our curriculum and mixture strategies, but no explicit expressivity or optimization guarantees were given. Finally, the hardness of learning the k-fold function class is related to the locality barrier studied in Abbe et al. (2024), where the authors established a computational lower bound for a particular cycle task that exhibits compositional structure, using the permutation invariance of the learning algorithm.

2. Preliminaries

Notation. Let S_N denote the set of permutations on N elements, and for two permutations $\pi, \sigma \in S_N$, let $\pi \circ \sigma$ denote their composition. For integer d and index $i \in [d]$, let $e_{d,i} \in \mathbb{R}^d$ denote the d-dimensional one-hot vector with a 1 in the ith coordinate. We use $\tilde{O}, \tilde{\Omega}$ to hide $\operatorname{poly}\log(kN)$ factors, and we use $f \lesssim g$ (or $f = O(g)$, $g = \Omega(f)$) when $f \leq Cg$ for an absolute constant $C > 0$.

2.1. The k-fold Composition Task

Let $\pi := (\pi_1, \ldots, \pi_k) \in (S_N)^k$ be a tuple of k hidden permutations. The *k-fold composition* task takes as input $(\sigma, x) \in \mathcal{X} := (S_N)^k \times [N]$, where $\sigma = (\sigma_1, \ldots, \sigma_k)$ is a tuple of k permutations

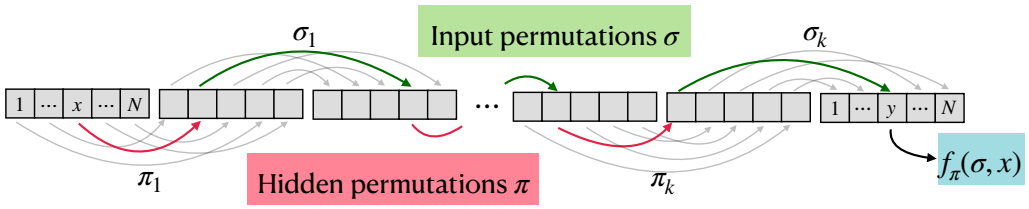

Figure 1: **k-fold composition task** – red arrows represent hidden permutations π_i and green arrows denote input permutations σ_i. Given input (σ, x), $f_\pi(\cdot, \cdot)$ composes $2k$ permutations to output $f_\pi(\sigma, x)$.

and x is an index in $[N]$. The task $f_\pi : \mathcal{X} \to [N]$ is defined as

$$f_\pi(\sigma, x) := (\sigma_k \circ \pi_k \circ \sigma_{k-1} \circ \pi_{k-1} \circ \cdots \circ \sigma_1 \circ \pi_1)(x).$$

Our target function class is $\mathcal{F} = \{f_\pi : \pi \in (S_N)^k\}$, and the goal is to learn $\mathcal{F}$ with respect to the uniform distribution over $\mathcal{X}$. See Figure 1 for illustration.

We will also consider an extension of the task called the *cyclic* k-fold composition task. Here, the input space is $\mathcal{X}^{\text{cyc}} := (S_N)^k \times [k] \times [N]$, and the target $f_\pi^{\text{cyc}} : \mathcal{X}^{\text{cyc}} \to [N]$ is defined by

$$f_\pi^{\text{cyc}}(\sigma, i, x) := (\sigma_{i+k-1} \circ \pi_{i+k-1} \circ \sigma_{i+k-2} \circ \pi_{i+k-2} \circ \cdots \circ \sigma_{i+1} \circ \pi_{i+1} \circ \sigma_i \circ \pi_i)(x),$$

where the indices of permutation are taken modulo k. We define $\mathcal{F}^{\text{cyc}} = \{f_\pi^{\text{cyc}} : \pi \in (S_N)^k\}$.

Remark 1 *As mentioned in the Introduction, the target function is defined as an interleaved composition of k contextual permutations σ (e.g., John $\to$ Quarterback) and k hidden permutations π not given in the input (e.g., Quarterback $\to$ Football) which may represent "parametric" or "in-weights" knowledge Chan et al. (2022); Yang et al. (2024); Cheng et al. (2024). The contextual permutation is a standard feature in the k-hop reasoning task Sanford et al. (2024b), whereas the parametric permutation enables us to define a function class to study the statistical hardness Kearns (1998) of compositional tasks.*

2.2. Transformer Architecture

Our learner is an L-layer transformer. Transformers take as input a length T sequence of d dimensional embedding vectors $X = [x_1, \ldots, x_T] \in \mathbb{R}^{d \times T}$. We will restrict ourselves to attention-only transformers, where each layer is a *self-attention head*, defined as follows:

Definition 2 (Self-attention head) *For a vector $v \in \mathbb{R}^k$, define the element-wise softmax operator $\mathcal{S} : \mathbb{R}^k \to \mathbb{R}^k$ by $\mathcal{S}(v)_i = \exp(v_i)/\sum_{j=1}^k \exp(v_j)$. The self-attention head is a mapping $\text{attn}(\,\cdot\,; W_{KQ}, W_{OV}) : \mathbb{R}^{d \times T} \to \mathbb{R}^{d \times T}$ parameterized by the key-query matrix $W_{KQ} \in \mathbb{R}^{d \times d}$ and output-value matrix $W_{OV} \in \mathbb{R}^{d \times d}$, which operates on a sequence of embeddings $X \in \mathbb{R}^{d \times T}$ as*

$$\text{attn}(X; W_{KQ}, W_{OV}) = X + W_{OV} X \mathcal{S}(X^\top W_{KQ} X),$$

where the softmax function $\mathcal{S}$ is applied column-wise.

A multi-layer transformer composes multiple self-attention heads in series. For simplicity, we consider transformers with a single head per layer.

Definition 3 (Attention-only transformer) *Let L be the depth and d be the embedding dimension. For $\ell \in [L]$, let $W_{KQ}^{(\ell)}, W_{OV}^{(\ell)}$ be the key-query and output-value matrices in the ℓ-th layer, respectively. Let $\theta := \{(W_{KQ}^{(\ell)}, W_{OV}^{(\ell)})\}_{\ell \in [L]}$ denote the aggregate parameter vector. A transformer $\mathrm{TF}_\theta : \mathbb{R}^{d \times T} \to \mathbb{R}^{d \times T}$ operates on an input sequence of embeddings $X \in \mathbb{R}^{d \times T}$ as follows:*

$$\begin{aligned} X^{(0)} &= X, \\ X^{(\ell)} &= \mathrm{attn}(X^{(\ell-1)}; W_{KQ}^{(\ell)}, W_{OV}^{(\ell)}), \quad i \in [L] \\ \mathrm{TF}_\theta(X) &= X^{(L)}. \end{aligned}$$

Embedding and Decoding. We embed the input σ as follows. Let $\phi : [k] \times [N] \times [N] \to \mathbb{R}^d$ be some embedding function; then, the input to the transformer is the length $T = kN$ sequence

$$X(\sigma) = \{\phi(i, j, \sigma_i(j))\}_{i \in [k], j \in [N]},$$

where the columns of $X(\sigma)$ are indexed by the tuples $(i, j) \in [k] \times [N]$.

The output of $\mathrm{TF}_\theta(X(\sigma))$ must also be decoded to a prediction in $[N]$ as follows. Let $\Psi \in \mathbb{R}^{d \times N}$, be the readout layer. The predictions of the learner for the permutation composition and the cyclic permutation tasks are given by

$$\hat{f}(\sigma, x) = (\Psi^\top \mathrm{TF}_\theta(X(\sigma))_{(1,x)} \in \mathbb{R}^N, \quad \text{and} \quad \hat{f}(\sigma, i, x) = (\Psi^\top \mathrm{TF}_\theta(X(\sigma))_{(i,x)} \in \mathbb{R}^N$$

respectively.

3. Transformer Construction

While the k-fold composition task requires composing the $2k$ permutations, we show that there exists a transformer with $O(\log k)$ layers that solves the task, given by the following theorem.

Theorem 4 *Assume that k is a power of two. There exists an embedding function ϕ with $d = kN(3 + \log_2 k)$ such that, for any $\pi \in (S_N)^k$, there exists an $L = \log_2 k + 1$ layer transformer which can exactly express the k-fold composition task, i.e*

$$(\Psi^\top \mathrm{TF}_\theta(X(\sigma)))_{(1,j)} = e_{N, f_\pi(\sigma, x)} \quad \textit{for all } (\sigma, x) \in \mathcal{X}.$$

Proof Sketch. The proof proceeds similarly to the constructions in Sanford et al. (2024b); Liu et al. (2023). For notational convenience, define the permutation $\mathrm{hop}_i^r(\sigma, \cdot) \in S_N$ by

$$\mathrm{hop}_i^r(\sigma, \cdot) := \sigma_{i+r-1} \circ \pi_{i+r-1} \circ \cdots \circ \sigma_{i+1} \circ \pi_{i+1} \circ \sigma_i \circ \pi_i.$$

We will consider the embedding

$$\phi(i, j, \sigma_i(j)) = \begin{bmatrix} E(i, j) \\ E(i, \sigma_i(j)) \\ 0_{kNL} \end{bmatrix} \in \mathbb{R}^{kN(L+2)}, \quad \text{where} \quad E(i, j) := e_{k,i} \otimes e_{N,j} \in \mathbb{R}^{kN}.$$

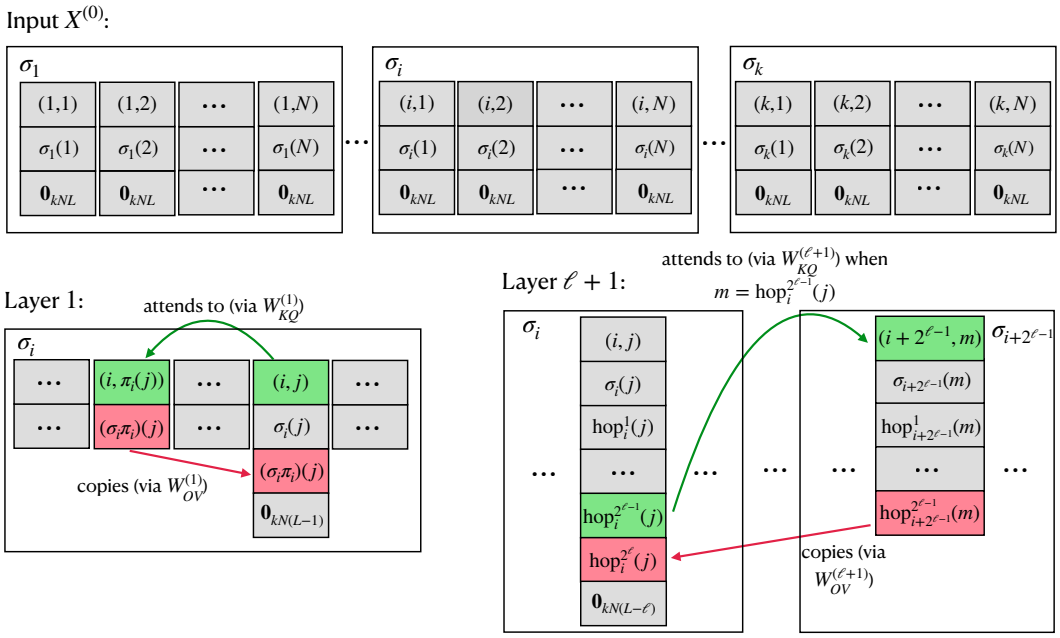

Figure 2: Illustration the format of input $X^{(0)}$ and the attention pattern in Theorem 4.

The first layer of the transformer encodes the hidden permutation π. The key-query matrix $W_{KQ}^{(1)}$ is set so that the (i,j) position attends to the $(i,\pi_i(j))$ position. The value matrix $W_{OV}^{(1)}$ then copies the second block of $X^{(0)}_{(i,\pi_i(j))}$, which encodes $\mathrm{hop}_i^1(\sigma,j)$, to the residual stream of (i,j) (see Figure 2, left). As such, $X^{(1)}_{(i,j)}$ now contains $\mathrm{hop}_i^1(\sigma,j)$.

The remainder of the construction proceeds recursively. Let us assume that the output of the ℓ-th layer has computed $\mathrm{hop}_i^{2^{\ell-1}}(\sigma,\cdot)$; in particular that $X^{(\ell)}$ is of the form

$$(X^{(\ell)})_{(i,j)} = \Big[E(i,j)^\top,\ E(i,\sigma_i(j))^\top,\ E(i,\mathrm{hop}_i^1(j))^\top,\ E(i+1,\mathrm{hop}_i^2(j))^\top,\ E(i+3,\mathrm{hop}_i^4(j))^\top,$$
$$\cdots\ E(i+2^{\ell-1}-1,\mathrm{hop}_i^{2^{\ell-1}}(j))^\top,\ 0_{kN(L-\ell)}^\top\Big]^\top.$$

The $(\ell+1)$th layer composes the quantities $\mathrm{hop}_i^{2^{\ell-1}}(\sigma,\cdot)$ and $\mathrm{hop}_{i+2^{\ell-1}}^{2^{\ell-1}}(\sigma,\cdot)$ to obtain $\mathrm{hop}_i^{2^\ell}(\sigma,\cdot)$. To do so, $W_{KQ}^{(\ell+1)}$ is first set so that the (i,j) position attends to the $(i+2^{\ell-1},\mathrm{hop}_i^{2^{\ell-1}}(j))$ position (by comparing the green blocks in Figure 2, right). Then, the value matrix $W_{OV}^{(\ell+1)}$ copies the last nonzero block (the red block in Figure 2, right) from $X^{(\ell)}_{(i,\mathrm{hop}_i^{2^{\ell-1}}(j))}$ to the residual stream of (i,j). The $(\ell+1)$th layer thus computes $\mathrm{hop}_{i+2^{\ell-1}}^{2^{\ell-1}}(\sigma,\mathrm{hop}_i^{2^{\ell-1}}(\sigma,j)) = \mathrm{hop}_i^{2^\ell}(\sigma,j)$ as desired.

Altogether $\log_2 k + 1$ layers suffice to compute $f_\pi(\sigma,x) = \mathrm{hop}_1^k(\sigma,x)$. The complete proof of Theorem 4 is contained in Appendix A. In Section A.2, we consider a modification of the k-fold composition task with $m \ll k$ hidden permutations, and show that an embedding dimension of $d = \tilde{\Theta}(mN)$ suffices.

4. Statistical Query Lower Bound

We have shown that for any $f_\pi \in \mathcal{F}$, there exists a transformer with $O(\log k)$ layers and embedding dimension $\mathrm{poly}(Nk)$ which can exactly express f_π. On the contrary, we will now show that in order to learn f_π, a learner must use either compute or sample size exponential in k. Formally, we prove a statistical query (SQ) lower bound for learning $\mathcal{F}$ (Kearns, 1998). Many learning algorithms, including gradient descent, can be understood in the SQ model, and thus SQ complexity is a useful proxy for the complexity of learning via gradient descent; we discuss this further in Appendix B.

Under the SQ framework, the learner can interact with the target function f_π by specifying a query $g : \mathcal{X} \times [N] \to \mathbb{R}$ and tolerance level τ; the SQ oracle then returns any response $\hat{q}$ satisfying[1] $|\hat{q} - \mathbb{E}[g(\sigma, x, f_\pi(\sigma, x))]| \le \tau$. We further assume without loss of generality that g satisfies the normalization $\sum_{y\in[N]} \mathbb{E}_{\sigma,x}[g(\sigma, x, y)^2] = 1$ and $\mathbb{E}_\sigma[g(\sigma, x, y)] = 0$ (see Appendix B for discussion on this choice of normalization). After making some number of queries, the learner outputs a predictor $\hat{f} : \mathcal{X} \to [N]$, which incurs the 0-1 loss $L(\hat{f}) := \mathbb{P}_{\sigma,x}\left(\hat{f}(\sigma, x) \neq f_\pi(\sigma, x)\right)$. Our lower bound against SQ learners is given by the following theorem:

Theorem 5 *Any SQ learner for the function class $\mathcal{F}$ or $\mathcal{F}^{\mathrm{cyc}}$ must either make $q \ge N^{\Omega(k)}$ queries or use a tolerance $\tau \le N^{-\Omega(k)}$ to output a predictor $\hat{f}$ with loss $L(\hat{f}) = O(1)$.*

Remark 6 *Using the standard $\tau \approx n^{-1/2}$ concentration heuristic, where n is the sample size, Theorem 5 implies that either runtime (at least the number of queries) or sample size n must be $\ge N^{\Omega(k)}$; this exponential dependence on k implies a large statistical-computational gap under the SQ class, since information theoretically $\tilde{\Theta}(Nk)$ samples are sufficient to learn this target function.*

The proof of Theorem 5 requires constructing a subset of $\mathcal{F}$ of nearly orthogonal functions. It turns out that the SQ model induces the following inner product between functions $f_\pi, f_\rho \in \mathcal{F}$.

$$\langle f_\pi, f_\rho \rangle := \mathbb{P}_{\sigma,x}[f_\pi(\sigma, x) = f_\rho(\sigma, x)] - 1/N. \tag{1}$$

One can interpret this inner product as a covariance between the random variables $f_\pi(\sigma, x)$ and $f_\rho(\sigma, x)$. An intermediate step towards the proof is Lemma 12, where we construct a large set of functions with a small pairwise correlation under the inner product (1). The complete proof of Theorem 5 is presented in Appendix B.

5. Upper Bound for Gradient-based Learning

Theorem 5 implies that any SQ learner with polynomial compute and access only to the target labels must use $N^{\Omega(k)}$ samples in order to learn a k-fold composition function f_π. We now show that with the aid of a specifically chosen *curriculum*, it is possible to train a $O(\log k)$ depth transformer to learn f_π with only $\mathrm{poly}(Nk)$ samples. Curriculum learning (Elman, 1993; Bengio et al., 2009) is the process of training a model on data with increasing difficulty, the benefit of which has been demonstrated in various empirical and theoretical works; see Section 7 for further discussion.

Our curriculum learning strategy is summarized as follows. Recall that for $i, r \in [k]$, we have defined the permutation $\mathrm{hop}_i^r(\sigma, \cdot) \in S_N$ by $\mathrm{hop}_i^r(\sigma, \cdot) = \sigma_{i+r-1} \circ \pi_{i+r-1} \circ \cdots \circ \sigma_{i+1} \circ \pi_{i+1} \circ \sigma_i \circ \pi_i$.

1. For ease of exposition, we focus here on the regular k-fold composition task and function class $\mathcal{F}$, and defer the cyclic variant to the appendix.

Algorithm 1 Training Algorithm (Curriculum)

Input: initialization size β_0; learning rates η

Initialize $W_{KQ}^{(\ell)}(0) = 0_{d\times d}$, $\Psi^{(\ell)}(0) = \beta_0 e_{L+2,\ell+2} \otimes \mathbf{1}_k \otimes I_{N\times N}$, $\ell \in [L]$

for $t = 1, \cdots, L$ **do**

- Use loss $\mathcal{L}^{(t)}(\theta(t-1))$. ▷ Stage t: train on $\mathrm{hop}^{2^{t-1}}$
- $\theta_{KQ}(t) \leftarrow \theta_{KQ}(t-1) - \eta \nabla_{\theta_{KQ}} \mathcal{L}^{(t)}(\theta(t-1))$ ▷ Train the key-query matrices θ_{KQ}
- $\theta' \leftarrow (\theta_{KQ}(t), \theta_\Psi(t-1))$
- $\theta_\Psi(t) \leftarrow \theta_\Psi(t-1) - \eta \nabla_{\theta_\Psi} \mathcal{L}^{(t)}(\theta')$ ▷ Train the readout layer θ_Ψ
- $\theta(t) \leftarrow (\theta_{KQ}(t), \theta_\Psi(t))$

end

Output: $\hat{\theta} = \theta(L)$.

Intuitively, the difficulty of the task scales with k, and thus we expect it to be much easier for a transformer to learn from samples of $\mathrm{hop}_i^r(\sigma, \cdot)$, for much smaller $r \le k$. Our construction in Theorem 4 also motivates a natural curriculum, as the model computes f_π recursively with the output of the ℓ-th layer being $\mathrm{hop}_i^{2^\ell-1}(\sigma, \cdot)$. We thus consider a curriculum where in the ℓ-th stage the model receives samples of $\mathrm{hop}_i^{2^{\ell-1}}(\sigma, \cdot)$. We begin by describing this strategy and the gradient-based training algorithm in detail and show that it can learn f_π in $\mathrm{poly}(N, k)$ samples.

5.1. Training Procedure

Our training algorithm is L-stage gradient descent with $L = \log k + 1$ on the cross entropy loss using the transformer architecture defined in Definition 3. We apply an easy-to-hard curriculum for training: for each stage ℓ, we sample M input sequences $\{\sigma^{(m)}\}_{m\in[M]}$, and for each sequence sample a query position (i_m, j_m) and receive as label its $2^{\ell-1}$th hop $\mathrm{hop}_{i_m}^{2^{\ell-1}}(\sigma^{(m)}, j_m)$. The empirical training loss of stage ℓ is thus

$$\mathcal{L}^{(\ell)}(\theta) = -\frac{1}{M}\sum_{m=1}^{M}\left[\sum_{s'\in[N]} \mathbf{1}\{s' = \mathrm{hop}_{i_m}^{2^{\ell-1}}(\sigma^{(m)}, j_m)\} \log\Big(\mathcal{S}(\Psi_\ell^\top \mathrm{TF}_\theta(X_m)_{(i_m,j_m)})_{s'}\Big)\right], \quad (2)$$

where we denote $X_m = X(\sigma^{(m)})$. The key-query matrices are initialized at $W_{KQ}^{(\ell)}(0) = 0_{d\times d}$, and the readout/unembedding layers are initialized at $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2} \otimes \mathbf{1}_k \otimes I_{N\times N}$ for initialization scale $\beta_0 > 0$. We fix the value matrix for each layer as $W_{OV}^{(\ell)} = e_{L+2,\ell+2} e_{L+2,\ell+1}^\top \otimes I_{kN\times kN}$ to match the sparsity pattern in the construction. See Section 7 for a discussion on learning value matrices.

We define the parameter $\theta := (\theta_{KQ}, \theta_\Psi)$, where $\theta_{KQ} = (W_{KQ}^{(1)}, \cdots, W_{KQ}^{(L)})$ and $\theta_\Psi = (\Psi_1, \cdots, \Psi_L)$. Within each stage, our algorithm first takes one step of gradient descent on all key-query matrices θ_{KQ}. We then take one gradient step on the readout layer θ_Ψ. Pseudocode is given in Algorithm 1.

5.2. Main Theorem

Our main theorem is that $\hat{\theta}$, the output of the L stage curriculum training in Algorithm 1, successfully learns all the 2^ℓ-hop functions for $\ell \le L$.

Theorem 7 (Guarantee for Algorithm 1) *Assume $k = 2^{L-1}$, $M \geq \tilde{\Omega}(k^4 N^6)$, $0 < \epsilon \leq \tilde{O}(\frac{1}{k^2 N^3})$, and $\eta \geq \tilde{\Omega}(\frac{k^2 N^3}{\beta_0} \log \frac{1}{\epsilon})$.Then, with high probability the final output $\hat{\theta}$ of Algorithm 1 satisfies that, over any draw of the input permutation σ and the query index (i, j), for any $\ell \in [L]$,*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_\ell^\top \mathrm{TF}_{\hat{\theta}}(X(\sigma))_{(i,j)}) - e_{\mathrm{hop}_i^{2^{\ell-1}}(j)} \right\|_\infty \leq \epsilon.$$

In particular, for $\ell = L$, the output of the transformer $\hat{f}(\sigma, i, j) = \Psi_L^\top \mathrm{TF}_{\hat{\theta}}(X(\sigma))_{(i,j)}$ approximates the k-fold composition task $f_\pi^{\mathrm{cyc}}(\sigma, i, j)$.

Note that the $\mathrm{poly}(N, k)$ sample complexity with curriculum learning represents an significant improvement over the exponential dependence on k in the SQ lower bound (Theorem 5). We provide a proof sketch in the next section with the complete proof deferred to Appendix C. For clarity, we denote $X := X(\sigma), \mathrm{hop}_i^r(\sigma, j) := \mathrm{hop}_i^r(j)$ when σ is clear from context.

5.3. Proof Sketch

5.3.1. STAGE 1: LEARNING THE HIDDEN PERMUTATION (1-HOP)

The first step of the proof is to show that during the first stage of training, the first attention layer $W_{KQ}^{(1)}$ learns the hidden permutations π_i for all $i \in [k]$. The proof strategy is to first analyze the population dynamics, and then upper bound the sample noise by concentration.

We begin by decomposing the transformer output in the following summation:

$$\mathrm{TF}_\theta(X) = X^{(0)} + \sum_{\ell=1}^{L} W_{OV}^{(\ell)} X^{(\ell-1)} \mathcal{S}({X^{(\ell-1)}}^\top W_{KQ}^{(\ell)} X^{(\ell-1)}).$$

By our choice of initialization of the ℓ-th readout layer Ψ_ℓ and value matrix $W_{OV}^{(\ell)}$, we observe that $\Psi_{\ell'}^\top W_{OV}^{(\ell)}$ is non-zero if and only if $\ell = \ell'$. This means the final relevant output for stage ℓ is the ℓ-th layer $W_{OV}^{(\ell)} X^{(\ell-1)} \mathcal{S}({X^{(\ell-1)}}^\top W_{KQ}^{(\ell)} X^{(\ell-1)})$, the gradient for the ℓ-th layer $W_{KQ}^{(\ell)}$ is non-zero only in stage $\geq \ell$, and the gradient for a layer $W_{KQ}^{(\ell)}$ is also close to zero after being trained. We thus only consider the gradient of $W_{KQ}^{(\ell)}$ with respect to the loss $\mathcal{L}^{(\ell)}$ and upper bound the error term after each stage. In the first stage, the only updated key-query matrix is $W_{KQ}^{(1)}$.

Now we show that after a large step of gradient descent, the key-query matrix $W_{KQ}^{(1)}$ encodes the hidden permutations π, and the token in the (i, j) position attends to the token in the $(i, \pi_i(j))$ position. Define $\mathcal{L}_{\mathcal{D}}^{(\ell)} := \mathbb{E}[\mathcal{L}^{(\ell)}]$ as the population loss. The population gradient for $W_{KQ}^{(1)}$ can be computed as follows:

$$\nabla_{W_{KQ}^{(1)}} \mathcal{L}_{\mathcal{D}}^{(1)} = -\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \left(\mathbf{1}\{s' = \mathrm{hop}_i^1(j)\} - \mathcal{S}(\Psi_1^\top \mathrm{TF}_{(i,j)})_{s'} \right) X J^{(1)} X^\top (\Psi_1^\top W_{OV}^{(1)})_{s'}^\top X_{(i,j)}^\top \right],$$

where we know that since $W_{KQ}^{(1)}$ is zero-initialized, $\mathcal{S}(\Psi_1^\top \mathrm{TF}_{(i,j)})_{s'} = \frac{1}{N}$ and the Jacobian term is $J^{(1)} = \frac{1}{kN}\left(I_{kN} - \frac{1}{kN} \mathbf{1}_{kN} \mathbf{1}_{kN}^\top\right)$. Recall that the input embedding is $X_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ 0_{knL} \end{bmatrix}$. Since

the readout layer is $(\Psi_1^\top W_{OV}^{(1)})_{s'}^\top = \beta_0 e_{L+2,2} \otimes \mathbf{1}_k \otimes e_{N,s'}$, one observes that $XX^\top (\Psi_1^\top W_{OV}^{(1)})_{s'}^\top = \beta_0 \sum_{p=1}^k X_{(p,s')}$. Finally, substituting $s' = \mathrm{hop}_i^1(j) = \sigma_i \pi_i(j)$, and dealing with cancellation in the mean centering term, one can simplify and expand the gradient into block matrices with dimension $\mathbb{R}^{kN \times kN}$ as follows:

$$\nabla_{W_{KQ}^{(1)}} \mathcal{L}_{\mathcal{D}}^{(1)} = -\frac{\beta_0}{kN} \mathbb{E} \left[\left(\begin{bmatrix} \sum_{p=1}^k e_{k,p} \otimes e_{N,\sigma_p^{-1}\sigma_i\pi_i(j)} \\ \sum_{p=1}^k e_{k,p} \otimes e_{N,\sigma_i\pi_i(j)} \\ 0_{knL} \end{bmatrix} - \frac{1}{N} \begin{bmatrix} \mathbf{1}_{kN} \\ \mathbf{1}_{kN} \\ 0_{knL} \end{bmatrix} \right) \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ 0_{knL} \end{bmatrix}^\top \right]$$

Let A_1 denote the top left $kN \times kN$ block of $W_{KQ}^{(1)}$. To attend to $(i, \pi_i(j))$, our construction in Theorem 4 uses A_1 to map $X_{(i,j)}$ to $X_{(i,\pi_i(j))}$; in particular, A_1 takes the first block in $X_{(i,j)}$, i.e. $e_{k,i} \otimes e_{N,j}$, and maps it to $e_{k,i} \otimes e_{N,\pi_i(j)}$. Computing the gradient of the population loss with respect to A_1, we observe that

$$\nabla_{A_1} \mathcal{L}_{\mathcal{D}}^{(1)} = -\frac{\beta_0}{kN} \mathbb{E}_{\sigma,(i,j)} \left[\left(\sum_{p=1}^k e_{k,p} \otimes e_{N,\sigma_p^{-1}\sigma_i\pi_i(j)} - \frac{1}{N} \mathbf{1}_{kN} \right) (e_{k,i} \otimes e_{N,j})^\top \right].$$

Conditioning on the query position (i, j), we observe that σ_p is independent of the query embedding $X_{(i,j)}$ when $p \neq i$. The terms with independent permutation σ_p^{-1} will have expectation $\frac{1}{N}\mathbf{1}_{kN}$ and cancel with the mean-centering term. Therefore, the only remaining term is $e_{k,i} \otimes e_{N,\pi_i(j)}$, Taking expectation over all (i, j), the gradient with respect to A_1 is thus

$$\nabla_{A_1} \mathcal{L}^{(1)} = -\frac{\beta_0}{k^2 N^2} \sum_{i=1}^k \sum_{j=1}^N \underbrace{(e_{k,i} \otimes e_{N,\pi_i(j)})(e_{k,i} \otimes e_{N,j})^\top}_{\text{Maps } (i,j) \to (i,\pi_i(j))} + \frac{\beta_0}{k^2 N^3} \sum_{i=1}^k \underbrace{(e_{k,i} \otimes \mathbf{1}_N)(e_{k,i} \otimes \mathbf{1}_N)^\top}_{\text{Mean-centering term}}.$$

We thus see that A_1 encodes all the hidden permutations π_i for all $i \in [k]$,

Moreover, we show that the gradient of the A_1 block dominates the gradient of the other blocks of $W_{KQ}^{(1)}$. Altogether, after concentrating the finite-sample gradient, we see that with high probability $\mathcal{S}(X^\top W_{KQ}^{(1)} X_{(i,j)})_{(i,\pi_i(j))} \approx 1$ and thus the first layer approximately outputs the 1-hop embedding

$$W_{OV}^{(1)} X^{(0)} \mathcal{S}({X^{(0)}}^\top W_{KQ}^{(1)} X^{(0)})_{(i,j)} \approx e_{L+2,3} \otimes e_{k,i} \otimes e_{N,\sigma_i\pi_i(j)}.$$

Next, after the second gradient step on the readout layer Ψ_1, the readout layer grows large in the direction $I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top$ and approximately outputs the one-hot vector of the correct hop $e_{N,\mathrm{hop}_i^1(j)} = e_{N,\sigma_i\pi_i(j)}$ after the output softmax layer.

5.3.2. STAGE ℓ ($2 \le \ell \le 1 + \log_2 k$): LEARNING THE $2^{\ell-1}$-HOP

Next, we inductively show that in the ℓ-th stage, the ℓ-th layer learns to compute the $2^{\ell-1}$-hop. In particular, we show that for a query $(i, j) \in [k] \times [N]$, the key-query matrix learns to compose $\mathrm{hop}_i^{2^{\ell-2}}$ and $\mathrm{hop}_{i+2^{\ell-2}}^{2^{\ell-2}}$ computed in the previous layer.

In Lemma 19 ($\ell = 2$) and Lemma 24 ($\ell \ge 3$), we show that a one-step gradient update on $W_{KQ}^{(\ell)}$ approximately matches the constructed ℓ-th layer key-query matrix in Theorem 4. After the gradient

descent step, in the ℓ-th layer the (i, j) position correctly attends to the $(i + 2^{\ell-2}, \text{hop}_i^{2^{\ell-2}}(j))$ position and extracts the $2^{\ell-2}$-hop, thus computing the desired output $e_{N,\text{hop}_i^{2^{\ell-1}}(j)}$.

Finally, we upper bound the total error from all ℓ stages altogether. We show inductively that the accumulation of errors in the actual intermediate sequences $\hat{X}^{(\ell-1)}$ (resulting from the finite-sample gradient and finite learning rate) grows linearly with depth, i.e. $\|\hat{X}^{(\ell)} - X^{(\ell)}\|_\infty \le \ell\epsilon'$, where ϵ' is the single stage error. For ϵ' sufficiently small, the output of the transformer is indeed close to the desired solution. To conclude the proof of Theorem 7, we finally train Ψ_L. As in Sec. 5.3.1, Ψ_L grows large and the transformer outputs $e_{N,\text{hop}_i^{2^{\ell-1}}(j)} = e_{N,\text{hop}_i^k(j)}$, as desired.

5.4. Implicit Curriculum via Data Mixture

In practice, designing a curriculum involving multiple stages with tasks of increasing difficulty can be quite challenging. As such, practitioners often rely on training data consisting of mixtures of various tasks with different skills and difficulties (Xie et al., 2024; Liu et al., 2024b; Dubey et al., 2024). We thus consider an arguably simpler training scheme, where instead of stage-wise learning with a curriculum, we directly train on a *mixture of easy-to-hard data* simultaneously. Consider the following objective for the mixture problem, which is the empirical loss summed over tasks with varying hops:

$$\mathcal{L}^{\text{M}}(\theta) := \sum_{\ell=1}^{L} \mathcal{L}^{(\ell)}(\theta) = -\frac{1}{M} \sum_{\ell=1}^{L} \sum_{m=1}^{M} \left[\sum_{s' \in [N]} \mathbf{1}\{s' = \text{hop}_{i_m}^{2^{\ell-1}}(\sigma^{(m)}, j_m)\} \log\big(\mathcal{S}(\Psi_\ell^\top \text{TF}_\theta(X_m)_{(i_m,j_m)})_{s'}\big) \right] \tag{3}$$

We remark that this objective is equivalent, at the population level, to predicting multiple hops for each input sequence at the same time, which relates to *multi-token prediction*, a technique that has recently been found to be effective in practice (Bachmann and Nagarajan, 2024; Gloeckle et al., 2024; Liu et al., 2024a). The following theorem shows that training on a mixture of easy-to-hard data also enables a transformer to learn the k-fold composition task by inducing an equivalent curriculum as Algorithm 1.

Theorem 8 (Guarantee for mixed data training) *Assume $k = 2^{L-1}$, $M \ge \tilde{\Omega}(k^4 N^6)$ and $\eta \ge \tilde{\Omega}(\frac{k^2 N^3}{\beta_0} \log \frac{1}{\epsilon})$. For sufficiently small $\epsilon > 0$, with high probability the final output $\hat{\theta}$ of Algorithm 2 satisfies that over any draw of input permutations σ and query index (i, j), for any $\ell \in [L]$,*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_\ell^\top \text{TF}_{\hat{\theta}}(X(\sigma))_{(i,j)}) - e_{\text{hop}_i^{2^{\ell-1}}(j)} \right\|_\infty \le \epsilon.$$

In particular, the transformer approximates the k-fold composition task when $\ell = L$.

The training algorithm (Algorithm 2) and the proof of Theorem 8 are deferred to Appendix D. The high level proof idea is as follows. Consider an "idealized" population gradient dynamics with all the attention outputs either one-hot or uniform. The key observation is that if all layers starting from the t-th layer are zero, i.e. $W_{KQ}^{(\ell)} = 0$ for $\ell \ge t$, the gradients of all later layers $\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{\text{M}} = 0$ for $\ell \ge t + 1$. This is because if the layer $W_{KQ}^{(\ell-1)}$ is zero, the output of the softmax will be the uniform distribution $\frac{1}{kN}\mathbf{1}_{kN}$, which contains no signal. This will cancel out with the mean-centering term in the Jacobian, leading to zero gradient. We can therefore show that in the idealized population dynamics, the ℓ-th layer is learned in the ℓ-th gradient step, thus mimicking Algorithm 1.

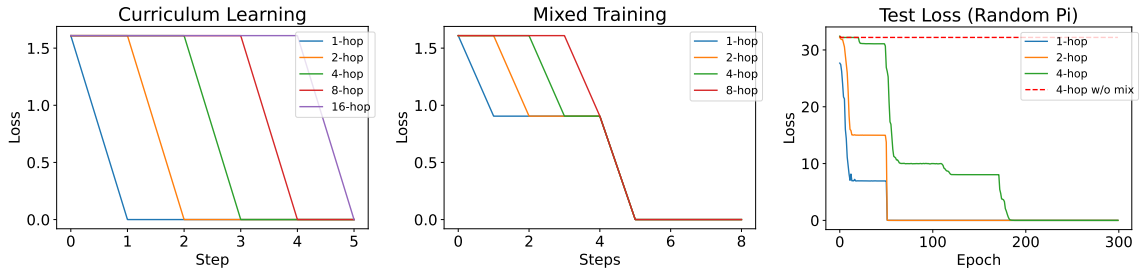

Figure 3: **Left:** Curriculum learning (Algorithm 1). **Middle:** Learning with data mixture (Algorithm 2). **Right:** Comparison between training with and without mixed data on a standard transformer.

6. Experiments

In this section, we provide empirical support for the conclusions of Theorem 7 and Theorem 8. In the leftmost plot of Figure 3, we consider training a 5 layer transformer with architecture and initialization exactly matching that of Theorem 7 with $k = 16, N = 5$. Our curriculum training procedure exactly follows that of the Algorithm 1: for $\ell \in [5]$, during the ℓ-th stage we train on $2^{\ell-1}$ hop data, and take a single gradient step on $W_{KQ}^{(\ell)}$ followed by a single gradient step on $\Psi^{(\ell)}$. We observe that the model's loss on $2^{\ell-1}$-hop examples decreases to zero after the ℓ-th stage, and thus after the final stage the model has perfectly learned the 16-fold composition. In the middle pane, we train a 4 layer transformer with architecture, initialization, and training procedure exactly matching that of the mixed training algorithm (Algorithm 2), with $k = 8, N = 5$. We observe that for $\ell \in [4]$, the model's loss on the $2^{\ell-1}$-hop examples decreases, and at the end of training,[2] the model has perfectly learned the 8-fold composition.

In the rightmost plot, we consider training a standard transformer on the 4-fold composition task. To more closely match standard language modeling tasks, each token in the sequence is simply the element $\sigma_i(j) \in [N]$, and the desired hop number is prepended to the beginning of the sequence. We use Adam Kingma (2014) as the optimizer and train a standard encoder transformer with learned embeddings, MLPs and layer normalization. For the training distribution, we first train the transformer with uniformly mixed $M = 10^5$ examples from $\{1, 2, 4\}$-fold composition data, and then train another transformer with only 4-fold compositions. We observe a similar phenomenology as predicted by Theorem 7 – if we only train on the 4-fold composition data, then the transformer is unable to learn, yet training with a curriculum learning strategy helps the model to correctly learn the 4-fold task.

7. Discussion and Future Directions

Connection to k-sparse parity. Theorem 5 implies a statistical-to-computational gap under the SQ framework: a covering number argument on the function class $\mathcal{F}$ yields an information-theoretic sample complexity of $kN \log N$, whereas an SQ learner with polynomial compute requires $N^{\Omega(k)}$ samples. This mirrors the case for k-sparse parities over d variables, where the SQ complexity

2. In Algorithm 2, the first 4 steps are on the $W_{KQ}^{(\ell)}$ matrices, and the remainder of the steps are on the readout matrices Ψ_ℓ, and thus the loss can only be decreased to 0 once the readout matrices are trained.

$q/\tau^2 \sim d^{\Omega(k)}$ is also statistically suboptimal. Heuristically speaking, both problems exhibit a "global" structure in which intermediate steps (partial solutions) are uncorrelated with the true function, and hence a learner using only correlational information pays a complexity exponential in k. For more discussion on similar computational lower bounds see Abbe et al. (2024, Appendix A.4).

For the parity problem, recent works have shown that the statistical and optimization complexity can be improved by introducing a curriculum (Abbe et al., 2023; Panigrahi et al., 2024). More relevant to our results, Kim and Suzuki (2024); Wen et al. (2024) showed that by decomposing the problem into subtasks of intermediate parities and employing process supervision (i.e., forcing the model to predict the intermediate steps), a single-layer transformer can learn k-parity with $\mathrm{poly}(d, k)$ samples, thereby avoiding exponential dependence in k. At a high level, such task decomposition resembles our curriculum learning objective, which constructs a "staircase" (Abbe et al., 2021, 2022) to guide gradient-based learning. Note that while parity can be represented by a single-layer transformer, it is believed that for tasks similar to our k-fold composition such as the k-hop induction head, $\log k$ layers are necessary for parameter-efficient representation (Sanford et al., 2024b).

On the embedding dimension and model size. Sanford et al. (2024b) show an advantage of transformers over recurrent neural networks (RNNs) for expressing the k-hop induction head task. Via a reduction to a communication complexity lower bound for the pointer-chasing problem, an RNN requires either depth at least k or width at least $N/\mathrm{poly}(k)$ to solve the k-hop task. In contrast, the $O(\log k)$ depth transformer construction succeeds with embedding dimension $d = O(1)$. Similarly, the $O(\log k)$-depth construction for the automata simulation task in Liu et al. (2023) requires embedding dimension and MLP width of $\mathrm{poly}(N)$, independent of k. In comparison, our construction in Theorem 4 appears suboptimal in that the embedding dimension is $\mathrm{poly}(Nk)$. However, since we require the embedding ϕ of the transformer to be fixed (that is, independent, of the target function f_π), the size of the representation is Ld^2p, where p is the bit precision. The representation size must be at least the log packing number, which is $\Theta(kN \log N)$ (Corollary 13). As such, an embedding dimension of $d = \mathrm{poly}(Nk)$ is necessary when $p = \tilde{O}(1)$. On the other hand, if we allow for ϕ to be trainable (i.e., depend on the target f_π), then we can encode $\mathrm{hop}_i^1(\sigma, j)$ using ϕ to yield a valid construction with embedding dimension $d = \tilde{O}(1)$, for example, by leveraging rotary embeddings similar to Sanford et al. (2023). We leave understanding whether the learned model can be distilled into a smaller dimensional model, or whether a smaller model with a trainable embedding map can still learn the task, to future work.

Learning the value matrix. Our analysis of learning dynamics in Section 5 assumes that the value matrix $W_{OV}^{(\ell)}$ at each layer is fixed to the desired block-sparse matrix with an identity block that matches our construction in Theorem 4. We note that while this simplifies the problem, it does not encode any information about the hidden permutations π in the target f_π. In practice, these matrices are not fixed at the identity, and we show in Appendix E that their population gradient at zero initialization vanishes under our data model. This suggests that other ingredients are needed for successfully learning the value matrices, such as exploiting a non-zero initialization or changing the data distribution. We leave a precise study of such learning dynamics to future work.

Generalization to $k \neq 2^\ell$. Our analysis assumes for simplicity that k is a power of two. When k is not a power of two, the construction in Theorem 4 does not work, as keeping only the 2^ℓ-hops is insufficient. For general k, we believe that a similar analysis applies provided a larger embedding dimension $\Theta(k^2 N)$ to encode all ℓ-hops with $\ell \in [k]$.

Acknowledgments

This collaboration began during the "Modern Paradigms in Generalization" and "Special Year on Large Language Models and Transformers, Part 1" programs at the Simons Institute for the Theory of Computing, Berkeley in 2024. DH acknowledges support from the ONR under grant N00014-24-1-2700. JDL acknowledges support of the NSF CCF 2002272, NSF IIS 2107304, NSF CIF 2212262, ONR Young Investigator Award, and NSF CAREER Award 2144994.

References

Emmanuel Abbe, Enric Boix-Adsera, Matthew S Brennan, Guy Bresler, and Dheeraj Nagaraj. The staircase property: How hierarchical structure can guide deep learning. *Advances in Neural Information Processing Systems*, 34:26989–27002, 2021.

Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for sgd learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, pages 4782–4887. PMLR, 2022.

Emmanuel Abbe, Elisabetta Cornacchia, and Aryo Lotfi. Provable advantage of curriculum learning on parity targets with mixed inputs. *Advances in Neural Information Processing Systems*, 36: 24291–24321, 2023.

Emmanuel Abbe, Samy Bengio, Aryo Lotfi, Colin Sandon, and Omid Saremi. How far can transformers reason? The locality barrier and inductive scratchpad. In *Advances in Neural Information Processing Systems*, 2024.

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

Sepehr Assadi and Vishvajeet N. Graph streaming lower bounds for parameter estimation and property testing via a streaming xor lemma. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 612–625, 2021.

Gregor Bachmann and Vaishnavh Nagarajan. The pitfalls of next-token prediction. In *International Conference on Machine Learning (ICML)*, 2024.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553380. URL https://doi.org/10.1145/1553374.1553380.

Satwik Bhattamishra, Michael Hahn, Phil Blunsom, and Varun Kanade. Separations in the representational capabilities of transformers and recurrent architectures. *arXiv preprint arXiv:2406.09347*, 2024.

Alberto Bietti, Vivien Cabannes, Diane Bouchacourt, Herve Jegou, and Leon Bottou. Birth of a transformer: A memory viewpoint. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. In *Advances in neural information processing systems*, 2022.

Lijie Chen, Binghui Peng, and Hongxun Wu. Theoretical limitations of multi-layer transformer. *arXiv preprint arXiv:2412.02975*, 2024a.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Training dynamics of multi-head softmax attention for in-context learning: Emergence, convergence, and optimality. *arXiv preprint arXiv:2402.19442*, 2024b.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Unveiling induction heads: Provable training dynamics and feature learning in transformers. *arXiv preprint arXiv:2409.10559*, 2024c.

Sitao Cheng, Liangming Pan, Xunjian Yin, Xinyi Wang, and William Yang Wang. Understanding the interplay between parametric and contextual knowledge for large language models. *arXiv preprint arXiv:2410.08414*, 2024.

Alexandru Damian, Jason Lee, and Mahdi Soltanolkotabi. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pages 5413–5452. PMLR, 2022.

Yuntian Deng, Yejin Choi, and Stuart Shieber. From explicit cot to implicit cot: Learning to internalize cot step by step. *arXiv preprint arXiv:2405.14838*, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Sean Welleck, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi. Faith and fate: Limits of transformers on compositionality (2023). *arXiv preprint arXiv:2305.18654*, 2023.

Jeffrey L Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48(1):71–99, 1993.

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36, 2024.

Cheng Gao, Yuan Cao, Zihao Li, Yihan He, Mengdi Wang, Han Liu, Jason Matthew Klusowski, and Jianqing Fan. Global convergence in training large-scale transformers. *arXiv preprint arXiv:2410.23610*, 2024.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. *arXiv preprint arXiv:2404.19737*, 2024.

William Timothy Gowers and Emanuele Viola. Interleaved group products. *SIAM Journal on Computing*, 48(2):554–580, 2019.

Jianhao Huang, Zixuan Wang, and Jason D Lee. Transformers learn to implement multi-step gradient descent with chain of thought. *arXiv preprint arXiv:2502.21212*, 2025.

Yu Huang, Yuan Cheng, and Yingbin Liang. In-context convergence of transformers. *arXiv preprint arXiv:2310.05249*, 2023.

Sungjin Im, Ravi Kumar, Silvio Lattanzi, Benjamin Moseley, and Sergei Vassilvitskii. Massively parallel computation: Algorithms and applications. *Foundations and Trends® in Optimization*, 5 (4):340–417, 2023.

Samy Jelassi, Michael Sander, and Yuanzhi Li. Vision transformers provably learn spatial structure. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Samy Jelassi, David Brandfonbrener, Sham M Kakade, and Eran Malach. Repeat after me: Transformers are better than state space models at copying. *arXiv preprint arXiv:2402.01032*, 2024.

Michael Kearns. Efficient noise-tolerant learning from statistical queries. *J. ACM*, 45(6):983–1006, 1998.

Juno Kim and Taiji Suzuki. Transformers provably solve parity efficiently with chain of thought. *arXiv preprint arXiv:2410.08633*, 2024.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Jack Lanchantin, Shubham Toshniwal, Jason Weston, and Sainbayar Sukhbaatar. Learning to reason and memorize with self-notes. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yuchen Li, Yuanzhi Li, and Andrej Risteski. How do transformers learn topic structure: Towards a mechanistic understanding. In *ICML*, 2023.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. *arXiv preprint arXiv:2402.12875*, 2024.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024a.

Bingbin Liu, Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Transformers learn shortcuts to automata. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=De4FYqjFueZ.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. *arXiv preprint arXiv:2407.01492*, 2024b.

William Merrill and Ashish Sabharwal. The expresssive power of transformers with chain of thought. *arXiv preprint arXiv:2310.07923*, 2023a.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023b.

Eshaan Nichani, Alex Damian, and Jason D Lee. How transformers learn causal structure with gradient descent. In *International Conference on Machine Learning (ICML)*, 2024a.

Eshaan Nichani, Jason D Lee, and Alberto Bietti. Understanding factual recall in transformers via associative memories. *arXiv preprint arXiv:2412.06538*, 2024b.

Noam Nisan and Avi Wigderson. Rounds in communication complexity revisited. *SIAM Journal on Computing*, 22(1):211–219, 1993. doi: 10.1137/0222016. URL https://doi.org/10.1137/0222016.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Abhishek Panigrahi, Bingbin Liu, Sadhika Malladi, Andrej Risteski, and Surbhi Goel. Progressive distillation induces an implicit curriculum. *arXiv preprint arXiv:2410.05464*, 2024.

Christos H. Papadimitriou and Michael Sipser. Communication complexity. *Journal of Computer and System Sciences*, 28(2):260–269, 1984. ISSN 0022-0000. doi: https://doi.org/10.1016/0022-0000(84)90069-2. URL https://www.sciencedirect.com/science/article/pii/0022000084900692.

Binghui Peng, Srini Narayanan, and Christos Papadimitriou. On limitations of the transformer architecture. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=KidynPuLNW.

Yunwei Ren, Zixuan Wang, and Jason D Lee. Learning and transferring sparse contextual bigrams with linear transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. Representational strengths and limitations of transformers. In *Advances in Neural Information Processing Systems 36*, 2023.

Clayton Sanford, Bahare Fatemi, Ethan Hall, Anton Tsitsulin, Mehran Kazemi, Jonathan Halcrow, Bryan Perozzi, and Vahab Mirrokni. Understanding transformer reasoning capabilities via graph algorithms. *arXiv preprint arXiv:2405.18512*, 2024a.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. Transformers, parallel computation, and logarithmic depth. In *Forty-First International Conference on Machine Learning*, 2024b.

Sainbayar Sukhbaatar, Jason Weston, and Rob Fergus. End-to-end memory networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Balázs Szörényi. Characterizing statistical query learning: simplified notions and proofs. In *International Conference on Algorithmic Learning Theory*, pages 186–200. Springer, 2009.

Yuandong Tian, Yiping Wang, Beidi Chen, and Simon Du. Scan and snap: Understanding training dynamics and token composition in 1-layer transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process-and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Zixuan Wang, Stanley Wei, Daniel Hsu, and Jason D. Lee. Transformers provably learn sparse token selection while fully-connected nets cannot. In *ICML*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Kaiyue Wen, Huaqing Zhang, Hongzhou Lin, and Jingzhao Zhang. From sparse dependence to sparse attention: unveiling how chain-of-thought enhances transformer sample efficiency. *arXiv preprint arXiv:2410.05459*, 2024.

Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.

Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart Van Merriënboer, Armand Joulin, and Tomas Mikolov. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*, 2015.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? *arXiv preprint arXiv:2402.16837*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Amir Yehudayoff. Pointer chasing via triangular discrimination. *Combinatorics, Probability and Computing*, 29(4):485–494, 2020.

Ruiqi Zhang, Spencer Frei, and Peter L Bartlett. Trained transformers learn linear models in-context. *arXiv preprint arXiv:2306.09927*, 2023.

Appendix A. Constructions

A.1. Construction for k-fold Composition

Proof [Proof of Theorem 4] Choose the embedding function as follows:

$$\phi(i, j, \sigma_i(j)) = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ 0_{knL} \end{bmatrix}$$

The first layer encodes the hidden permutation π in the key-query matrix $W_{KQ}^{(1)}$. In particular, set

$$W_{KQ}^{(1)} = \beta_0 \sum_{i\in[k],j\in[N]} \begin{bmatrix} e_{k,i} \otimes e_{N,\pi_i(j)} \\ 0_{kN(L+1)} \end{bmatrix} \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ 0_{kN(L+1)} \end{bmatrix}^\top$$

for large constant β_0. As such, the pre-attention weights from the position (i, j) are given by

$$X^{(0)\top} W_{KQ}^{(1)} X_{(i,j)}^{(0)} = \beta_0 e_{k,i} \otimes e_{N,\pi_i(j)}.$$

Taking $\beta_0 \to \infty$, the attention weight from (i, j) position concentrates on the $(i, \pi_i(j))$ position, so

$$\left(X^{(0)} \mathcal{S}\left(X^{(0)\top} W_{KQ}^{(1)} X^{(0)}\right)\right)_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,\pi_i(j)} \\ e_{k,i} \otimes e_{N,\sigma_i(\pi_i(j))} \\ 0_{knL} \end{bmatrix}$$

Finally, setting the output value matrix to be $W_{OV}^{(1)} = (e_{L+2,3} e_{L+2,2}^\top) \otimes I_{kN\times kN}$ yields

$$(X^{(1)})_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\sigma_i(\pi_i(j))} \\ 0_{kN(L-1)} \end{bmatrix}$$

The remainder of the construction proceeds inductively. For simplicity of notation, let us define the permutation hop_i^r by

$$\mathrm{hop}_i^r := \sigma_{i+r-1} \circ \pi_{i+r-1} \circ \cdots \circ \sigma_{i+1} \circ \pi_{i+1} \circ \sigma_i \circ \pi_i,$$

where the dependence on σ and π is implicit.

We will prove by induction that the output of the ℓth layer of transformer satisfies, for $i \in [k + 1 - 2^{\ell-1}]$ and $j \in [N]$,

$$(X^{(\ell)})_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}_i^1(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ e_{k,i+3} \otimes e_{N,\mathrm{hop}_i^4(j)} \\ \vdots \\ e_{k,i+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} \\ 0_{kN(L-\ell)} \end{bmatrix}.$$

We have already shown that this is indeed satisfied for $\ell = 1$.

Assume that the inductive hypothesis holds for some ℓ. Set the key-query matrix of layer $\ell + 1$ to be

$$W_{KQ}^{(\ell+1)} = \beta_{\ell+1} \sum_{a=1}^{k} \sum_{b \in [N]} \begin{bmatrix} e_{k,a+2^{\ell-1}} \otimes e_{N,b} \\ 0_{kN(L+1)} \end{bmatrix} \begin{bmatrix} 0_{kN(\ell+1)} \\ e_{k,a+2^{\ell-1}-1} \otimes e_{N,b} \\ 0_{kN(L-\ell)} \end{bmatrix}^\top.$$

Here when $a + 2^{\ell-1} > k$, we denote $e_{k,a+2^{\ell-1}} := e_{k,a+2^{\ell-1}-k}$, i.e. the indices of the permutation or embedding are taken modulo k. Consider some position $(i, j) \in [k] \times [N]$. The pre-attention weights from position (i, j) are given by

$$X^{(\ell)^\top} W_{KQ}^{(\ell+1)} X_{(i,j)}^{(\ell)} = \beta_{\ell+1} \cdot e_{k,i+2^{\ell-1}} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}.$$

Taking $\beta_{\ell+1} \to \infty$, the attention weight from the (i, j) position will concentrate on the $(i + 2^{\ell-1}, \mathrm{hop}_i^{2^{\ell-1}}(j))$ position. Setting the value matrix to be $W_{OV}^{(\ell+1)} = e_{L+2,\ell+3} e_{L+2,\ell+2}^\top \otimes I_{kN \times kN}$ ensures that

$$\begin{aligned} \left(W_{OV}^{(\ell+1)} X^{(\ell)} \mathcal{S}(X^{(\ell)^\top} W_{KQ}^{(\ell+1)} X^{(\ell)})\right)_{i,j} &= \begin{bmatrix} 0_{kN(\ell+2)} \\ e_{k,i+2^{\ell-1}+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_{i+2^{\ell-1}}^{2^{\ell-1}}\left(\mathrm{hop}_i^{2^{\ell-1}}(j)\right)} \\ 0_{kN(L-\ell-1)} \end{bmatrix} \\ &= \begin{bmatrix} 0_{kN(\ell+2)} \\ e_{k,i+2^{\ell}-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell}}(j)} \\ 0_{kN(L-\ell-1)} \end{bmatrix} \end{aligned}$$

Therefore $X_{(i,j)}^{(\ell+1)}$ satisfies

$$(X^{(\ell+1)})_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}_i^1(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ e_{k,i+3} \otimes e_{N,\mathrm{hop}_i^4(j)} \\ \vdots \\ e_{k,i+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_i^{2^\ell}(j)} \\ 0_{kN(L-\ell-1)} \end{bmatrix},$$

as desired. Therefore by induction, the claim holds for $\ell = L - 1$ and thus

$$(X^{(L)})_{(1,j)} = \begin{bmatrix} e_{k,1} \otimes e_{N,j} \\ e_{k,1} \otimes e_{N,\sigma_1(j)} \\ e_{k,1} \otimes e_{N,\mathrm{hop}_1^1(j)} \\ e_{k,2} \otimes e_{N,\mathrm{hop}_1^2(j)} \\ e_{k,4} \otimes e_{N,\mathrm{hop}_1^4(j)} \\ \vdots \\ e_{k,k} \otimes e_{N,\mathrm{hop}_1^k(j)} \end{bmatrix}$$

To conclude, set the readout layer Ψ to be

$$\Psi^\top = \begin{bmatrix} 0_{N\times kN(L+1)}, & e_{k,k} \otimes I_{N\times N}. \end{bmatrix}$$

The output of the transformer then satisfies

$$\left(\Psi^\top \mathrm{TF}_\theta(X)\right)_{(1,j)} = e_{N,\mathrm{hop}_1^k(j)} = e_{N,f_\pi(\sigma,j)},$$

as desired. ■

A.2. The m-sparse k-fold Composition

The k-fold composition task requires composing k hidden permutations with k input permutations. In practice, multi-step reasoning tasks may contain more contextual knowledge than parametric knowledge. We thus introduce a variant of the k-fold composition task, called the *m-sparse k-fold composition*, which requires composing m hidden permutations with k input permutations for $m \le k$.

Recall that the k-fold composition task is defined as $f_\pi : \mathcal{X} \to [N]$:

$$f_\pi(\sigma, x) := (\sigma_k \circ \pi_k \circ \sigma_{k-1} \circ \pi_{k-1} \circ \cdots \circ \sigma_1 \circ \pi_1)(x).$$

Our m-sparse k-fold composition target function class is defined as

$$\mathcal{F}_m := \{f_\pi : \pi \in (S_N)^k, \sum_{i=1}^k \mathbf{1}\{\pi_i = \mathrm{Id}\} \ge k - m\},$$

i.e those f_π where at most m of the π are not the identity. The goal is to learn $\mathcal{F}_m$ with respect to the uniform distribution over $\mathcal{X}$. We similarly define the cyclic task $\mathcal{F}_m^{\mathrm{cyc}} = \{f_\pi^{\mathrm{cyc}} : \pi \in (S_N)^k, \sum_{i=1}^k \mathbf{1}\{\pi_i = \mathrm{Id}\} \ge k - m\}$.

The following construction shows that the m-sparse k-fold composition is expressible by a $\Theta(\log k)$-depth transformer with embedding dimension $d = \tilde{\Theta}(mN)$.

Theorem 9 *Assume that k is a power of two. There exists an embedding function ϕ with $d = (m+2)N(3+\log_2 k)$ such that, for any $\pi \in (S_N)^k$ containing at most m non-identity hidden permutations, there exists an $L = \log_2 k + 1$ layer transformer which can exactly express the permutation composition function, i.e*

$$(\Psi^\top \mathrm{TF}_\theta(X(\sigma)))_{(1,j)} = e_{N,f_\pi(\sigma,x)} \quad \textit{for all } (\sigma, x) \in \mathcal{X}.$$

Proof [Proof of Theorem 9] Suppose the index set of non-identity hidden permutation π_i is $\mathcal{M} = \{i_1, ..., i_m\}$. Choose the embedding function as follows:

$$\phi(i, j, \sigma_i(j)) = \begin{bmatrix} p_i \otimes e_{N,j} \\ p_i \otimes e_{N,\sigma_i(j)} \\ 0_{kNL} \end{bmatrix}$$

where the positional encoding for the block position combines sinusoidal and one-hot positional encoding. We further denote $\hat{p}_i = (\cos\left(\frac{2\pi i}{k}\right), \sin\left(\frac{2\pi i}{k}\right))^\top \in \mathbb{R}^2$ as the sinusoidal embedding.

$$p_i := \begin{bmatrix} 0_m \\ \hat{p}_i \end{bmatrix} = \begin{bmatrix} 0_m \\ \cos\left(\frac{2\pi i}{k}\right) \\ \sin\left(\frac{2\pi i}{k}\right) \end{bmatrix} \in \mathbb{R}^{m+2} \text{ if } i \notin \mathcal{M}, \quad p_{i_j} := \begin{bmatrix} 0_m \\ \hat{p}_{i_y} \end{bmatrix} = \begin{bmatrix} e_{m,j} \\ \cos\left(\frac{2\pi i_j}{k}\right) \\ \sin\left(\frac{2\pi i_j}{k}\right) \end{bmatrix} \in \mathbb{R}^{m+2} \text{ if } i_j \in \mathcal{M}$$

The first layer encodes the hidden permutation π in the key-query matrix $W_{KQ}^{(1)}$. In particular, set

$$W_{KQ}^{(1)} = \begin{bmatrix} A_1 & 0 \\ 0 & 0 \end{bmatrix}, A_1 \in \mathbb{R}^{(m+2)N \times (m+2)N}$$

where the top-left block is

$$A_1 = \beta_1 \begin{bmatrix} 0_{m\times m} & 0 \\ 0 & I_2 \end{bmatrix} \otimes I_N + \beta_2 \sum_{i_y \in \mathcal{M}, j \in [N]} \begin{bmatrix} e_{m,y} \\ 0 \end{bmatrix} \begin{bmatrix} e_{m,y} \\ 0 \end{bmatrix}^\top \otimes e_{N,\pi_{i_y}(j)} e_{N,j}^\top$$

for large constant $\beta_2 \gg \beta_1 \gg 1$. As such, the pre-attention weights from the position (i, j) are given by

$$X^{(0)\top} W_{KQ}^{(1)} X_{(i,j)}^{(0)} = \beta_1 \sum_{i'=1}^{k} \cos\left(\frac{2\pi(i'-i)}{k}\right) e_{k,i} \otimes e_{N,j} \text{ if } i \notin \mathcal{M} .$$

$$X^{(0)\top} W_{KQ}^{(1)} X_{(i_y,j)}^{(0)} = \beta_1 \sum_{i'=1}^{k} \cos\left(\frac{2\pi(i'-i)}{k}\right) e_{k,i} \otimes e_{N,j} + \beta_2 e_{k,i_y} \otimes e_{N,\pi_{i_y}(j)} \text{ if } i_y \in \mathcal{M} .$$

Taking $\beta_2/\beta_1, \beta_1 \to \infty$, the attention weight from (i, j) position concentrates on the $(i, \pi_i(j))$ position (either $i \in \mathcal{M}$ or not), so

$$\left(X^{(0)} \mathcal{S}\left(X^{(0)\top} W_{KQ}^{(1)} X^{(0)}\right)\right)_{(i,j)} = \begin{bmatrix} p_i \otimes e_{N,\pi_i(j)} \\ p_i \otimes e_{N,\sigma_i(\pi_i(j))} \\ 0_{(m+2)NL} \end{bmatrix}$$

Finally, setting the output value matrix to be $W_{OV}^{(1)} = (e_{L+2,3} e_{L+2,2}^\top) \otimes I_{(m+2)N \times (m+2)N}$ yields

$$(X^{(1)})_{(i,j)} = \begin{bmatrix} p_i \otimes e_{N,j} \\ p_i \otimes e_{N,\sigma_i(j)} \\ p_i \otimes e_{N,\sigma_i(\pi_i(j))} \\ 0_{(m+2)N(L-1)} \end{bmatrix}$$

The remainder of the construction proceeds inductively. For simplicity of notation, let us define the permutation hop_i^r by

$$\mathrm{hop}_i^r := \sigma_{i+r-1} \circ \pi_{i+r-1} \circ \cdots \circ \sigma_{i+1} \circ \pi_{i+1} \circ \sigma_i \circ \pi_i,$$

where the dependence on σ and π is implicit.

We will prove by induction that the output of the ℓth layer of transformer satisfies, for $i \in [k+1-2^{\ell-1}]$ and $j \in [N]$,

$$(X^{(\ell)})_{(i,j)} = \begin{bmatrix} p_i \otimes e_{N,j} \\ p_i \otimes e_{N,\sigma_i(j)} \\ p_i \otimes e_{N,\mathrm{hop}_i^1(j)} \\ p_{i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ p_{i+3} \otimes e_{N,\mathrm{hop}_i^4(j)} \\ \vdots \\ p_{i+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} \\ 0_{(m+2)N(L-\ell)} \end{bmatrix}.$$

We have already shown that this is indeed satisfied for $\ell = 1$.

Assume that the inductive hypothesis holds for some ℓ. Set the key-query matrix of layer $\ell+1$ to

$$W_{KQ}^{(\ell+1)} = \beta_{\ell+1} \begin{bmatrix} 0_{(m+2)N\times(m+2)N(\ell+1)} & A^{(\ell+1)} & 0_{(m+2)N\times(m+2)N(L-\ell)} \\ 0_{(m+2)N(L+1)\times(m+2)N(\ell+1)} & 0_{(m+2)N\times(m+2)N} & 0_{(m+2)N(L+1)\times(m+2)N(L-\ell)} \end{bmatrix}$$

where the block $A^{(\ell+1)}$ should be

$$A^{(\ell+1)} = \begin{bmatrix} 0_{m\times m} & 0_m & 0_m \\ 0_m^\top & \cos\frac{2\pi}{k} & -\sin\frac{2\pi}{k} \\ 0_m^\top & \sin\frac{2\pi}{k} & \cos\frac{2\pi}{k} \end{bmatrix} \otimes I_N$$

Consider some position $(i,j) \in [k]\times[N]$. The pre-attention weights from position (i,j) are given by

$$X^{(\ell)\top} W_{KQ}^{(\ell+1)} X_{(i,j)}^{(\ell)} = \beta_{\ell+1} \cdot \sum_{i'=1}^{k} \cos\left(\frac{2\pi(i'-i-2^{\ell-1})}{k}\right) e_{k,i'+2^{\ell-1}} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}.$$

Taking $\beta_{\ell+1} \to \infty$, the attention weight from the (i,j) position will concentrate on the $(i+2^{\ell-1}, \mathrm{hop}_i^{2^{\ell-1}}(j))$ position. Setting the value matrix to be $W_{OV}^{(\ell+1)} = e_{L+2,\ell+3} e_{L+2,\ell+2}^\top \otimes I_{(m+2)N\times(m+2)N}$ ensures that

$$\begin{aligned} \left(W_{OV}^{(\ell+1)} X^{(\ell)} \mathcal{S}(X^{(\ell)\top} W_{KQ}^{(\ell+1)} X^{(\ell)})\right)_{i,j} &= \begin{bmatrix} 0_{(m+2)N(\ell+2)} \\ p_{i+2^{\ell-1}+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_{i+2^{\ell-1}}^{2^{\ell-1}}\left(\mathrm{hop}_i^{2^{\ell-1}}(j)\right)} \\ 0_{(m+2)N(L-\ell-1)} \end{bmatrix} \\ &= \begin{bmatrix} 0_{(m+2)N(\ell+2)} \\ p_{i+2^{\ell}-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell}}(j)} \\ 0_{(m+2)N(L-\ell-1)} \end{bmatrix} \end{aligned}$$

Therefore $X_{(i,j)}^{(\ell+1)}$ satisfies

$$(X^{(\ell+1)})_{(i,j)} = \begin{bmatrix} p_i \otimes e_{N,j} \\ p_i \otimes e_{N,\sigma_i(j)} \\ p_i \otimes e_{N,\mathrm{hop}_i^1(j)} \\ p_{i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ p_{i+3} \otimes e_{N,\mathrm{hop}_i^4(j)} \\ \vdots \\ p_{i+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_i^{2^\ell}(j)} \\ 0_{kN(L-\ell-1)} \end{bmatrix},$$

as desired. Therefore by induction, the claim holds for $\ell = L-1$ and thus

$$(X^{(L)})_{(1,j)} = \begin{bmatrix} p_1 \otimes e_{N,j} \\ p_1 \otimes e_{N,\sigma_1(j)} \\ p_1 \otimes e_{N,\mathrm{hop}_1^1(j)} \\ p_2 \otimes e_{N,\mathrm{hop}_1^2(j)} \\ p_4 \otimes e_{N,\mathrm{hop}_1^4(j)} \\ \vdots \\ p_k \otimes e_{N,\mathrm{hop}_1^k(j)} \end{bmatrix}$$

To conclude, set the readout layer Ψ to be

$$\Psi^\top = \begin{bmatrix} 0_{N\times(m+2)N(L+1)}, & p_k \otimes I_{N\times N} \cdot \end{bmatrix}$$

The output of the transformer then satisfies

$$\left(\Psi^\top \mathrm{TF}_\theta(X)\right)_{(1,j)} = e_{N,\mathrm{hop}_1^k(j)} = e_{N,f_\pi(\sigma,j)},$$

as desired. ■

Appendix B. SQ Lower Bound

Connection to Gradient Descent. Consider a neural network $F_\theta : \mathcal{X} \to \mathbb{R}^N$ which outputs the logits for predicting $f_\pi(\sigma, x)$. The standard cross entropy loss is given by

$$L(F_\theta) = \mathbb{E}_{\sigma,x}\left[\sum_{m\in[N]} \mathbf{1}(f_\pi(\sigma,x)=m)\log \mathcal{S}(F_\theta(\sigma,x))_m\right],$$

and thus the gradient descent update on the population loss is

$$\theta' \leftarrow \theta - \eta\mathbb{E}_{\sigma,x}\left[\sum_{m\in[N]} \mathbf{1}(f_\pi(\sigma,x)=m)\nabla_\theta F_\theta(\sigma,x)_m\right] + \eta\mathbb{E}_{\sigma,x}\left[\frac{\sum_{m\in[N]}\exp\left(F_\theta(\sigma,x)_m\right)\nabla_\theta F_\theta(\sigma,x)_m}{\sum_{m\in[N]}\exp\left(F_\theta(\sigma,x)_m\right)}\right].$$

The first term is exactly a (vector-valued) SQ query with $g(\sigma, x, y) = \nabla_\theta F_\theta(\sigma, x)_y$, while the second normalization term is independent of the unknown target function f_π. We remark that this connection is still heuristic, due to the mismatch between i.i.d noise for gradient descent and adversarial noise in the SQ framework.

On the normalization. The zero mean assumption is indeed without loss of generality, since for any query g, the mean-centered query $\bar{g}(\sigma, x, y) = g(\sigma, x, y) - \mathbb{E}_{\sigma'}[g(\sigma', x, y)]$ satisfies

$$\begin{aligned}\mathbb{E}_{\sigma,x}[\bar{g}(\sigma, x, f_\pi(\sigma, x))] &= \mathbb{E}_{\sigma,x}[g(\sigma, x, f_\pi(\sigma, x))] - \mathbb{E}_{\sigma,x,\sigma'}\big[g(\sigma', x, f_\pi(\sigma, x))\big] \\ &= \mathbb{E}_{\sigma,x}[g(\sigma, x, f_\pi(\sigma, x))] - \frac{1}{N^2}\sum_{x,y\in[N]} \mathbb{E}_\sigma[g(\sigma, x, y)],\end{aligned}$$

The last term is independent of the hidden permutation π, and thus the queries g and $\bar{g}$ reveal the same information about π.

Notation. For $\pi \in (S_N)^k$, we will let $f_\pi(\sigma)$ denote the permutation $f_\pi(\sigma) = \sigma_k \circ \pi_k \circ \cdots \circ \sigma_1 \circ \pi_1$. We will additionally overload notation, to let $f_\pi(\sigma)$ refer to the corresponding $N \times N$ permutation matrix. Finally, for two matrices $A, B \in \mathbb{R}^{N\times N}$, we let $\langle A, B\rangle = \mathrm{Tr}\big(A^\top B\big)$ denote the standard matrix inner product.

Our first goal is to construct a large subset of $\mathcal{F}$, which are nearly orthogonal under the inner product

$$\langle f_\pi, f_\rho\rangle := \mathbb{P}_{\sigma,x}(f_\pi(\sigma, x) = f_\rho(\sigma, x)) - \frac{1}{N} = \frac{1}{N}(\mathbb{E}_\sigma[\langle f_\pi(\sigma), f_\rho(\sigma)\rangle] - 1).$$

We first derive an explicit formula for the inner product between two permutation composition functions.

Lemma 10 (Inner product between functions) *For two permutation composition functions $f_\pi, f_\rho \in \mathcal{F}$,*

$$\langle f_\pi, f_\rho\rangle = \frac{1}{N(N-1)^{k-1}}\prod_{i=1}^{k}(\langle \pi_k, \rho_k\rangle - 1),$$

where $\langle \pi_k, \rho_k\rangle = \sum_{i\in[N]} \mathbf{1}(\pi_k(i) = \rho_k(i))$.

Proof Let us overload notation, so that for a permutation π_i, we also let π_i be the $N \times N$ permutation matrix. We see that

$$\langle f_\pi, f_\rho\rangle = \frac{1}{N}\Big(\mathbb{E}_\sigma\Big[\mathrm{Tr}\Big(\sigma_k\pi_k \cdots \sigma_1\pi_1\rho_1^\top\sigma_1^\top \cdots \rho_k^\top\sigma_k^\top\Big)\Big] - 1\Big)$$

Let us first compute this expectation with respect to σ_1.

Define $A := \rho_2^\top\sigma_2^\top \cdots \rho_k^\top\sigma_k^\top\sigma_k\pi_k \cdots \sigma_2\pi_2$, $B = \pi_1\rho_1^\top$. We have that

$$\mathrm{Tr}\Big(A\sigma_1 B\sigma_1^\top\Big) = \sum_{a,b,c,d\in[N]} A_{ab}(\sigma_1)_{bc}B_{cd}(\sigma_1)_{ad}.$$

Since σ_1 sampled uniformly at random from S_N, we have that

$$\mathbb{E}_{\sigma_1}[(\sigma_1)_{bc}(\sigma_1)_{ad}] = \begin{cases} \frac{1}{N} & a=b, c=d \\ 0 & a=b, c\neq d \\ 0 & a\neq b, c=d \\ \frac{1}{N(N-1)} & a\neq b, c\neq d \end{cases},$$

where the last equality is because

$$\begin{aligned}\mathbb{E}_{\sigma_1}[(\sigma_1)_{bc}(\sigma_1)_{ad}] &= \mathbb{P}(\sigma_1(a)=d, \sigma(b)=c) \\ &= \mathbb{P}(\sigma_1(a)=d)\cdot\mathbb{P}(\sigma_1(b)=c \mid \sigma_1(a)=d) \\ &= \frac{1}{N(N-1)}.\end{aligned}$$

Altogether,

$$\begin{aligned}&\mathbb{E}_{\sigma_1}\Big[\mathrm{Tr}\Big(A\sigma_1 B\sigma_1^\top\Big)\Big] = \frac{1}{N}\sum_{a,c} A_{aa}B_{cc} + \frac{1}{N(N-1)}\sum_{a\neq b, c\neq d} A_{ab}B_{cd} \\ &= \frac{1}{N}\mathrm{Tr}(A)\,\mathrm{Tr}(B) + \frac{1}{N(N-1)}\left(\sum_{a,b} A_{ab} - \mathrm{Tr}(A)\right)\left(\sum_{c,d} B_{c,d} - \mathrm{Tr}(B)\right) \\ &= \frac{1}{N}\mathrm{Tr}(A)\,\mathrm{Tr}(B) + \frac{1}{N(N-1)}(N-\mathrm{Tr}(A))(N-\mathrm{Tr}(B)) \\ &= \frac{(\mathrm{Tr}(A)-1)(\mathrm{Tr}(B)-1)}{N-1} + 1.\end{aligned}$$

where the second-to-last equality uses the property that A, B are both permutation matrices, and thus $\sum_{a,b} A_{ab} = \sum_{c,d} B_{c,d} = N$. Plugging everything back in, we see that

$$\begin{aligned}&\mathbb{E}_{\sigma_1,\ldots,\sigma_k}\left[\mathrm{Tr}\left(\sigma_k\pi_k\cdots\sigma_1\pi_1\rho_1^\top\sigma_1^\top\cdots\rho_k^\top\sigma_k^\top\right)\right] - 1 \\ &= \mathbb{E}_{\sigma_1,\ldots,\sigma_k}\left[\mathrm{Tr}\Big(A\sigma_1 B\sigma_1^\top\Big)\right] - 1 \\ &= \mathbb{E}_{\sigma_2,\ldots,\sigma_k}\left[\frac{(\mathrm{Tr}(A)-1)(\mathrm{Tr}(B)-1)}{N-1}\right] \\ &= \frac{\langle\pi_1,\rho_1\rangle - 1}{N-1}\cdot(\mathbb{E}_{\sigma_2,\ldots,\sigma_k}[\mathrm{Tr}(A)] - 1) \\ &= \frac{\langle\pi_1,\rho_1\rangle - 1}{N-1}\cdot\Big(\mathbb{E}_{\sigma_2,\ldots,\sigma_k}\Big[\mathrm{Tr}\Big(\sigma_k\pi_k\cdots\sigma_2\pi_2\rho_2^\top\sigma_2^\top\cdots\rho_k^\top\sigma_k^\top\Big)\Big] - 1\Big).\end{aligned}$$

Computing this quantity recursively, we get

$$\begin{aligned}&\mathbb{E}_{\sigma_1,\ldots,\sigma_k}\left[\mathrm{Tr}\left(\sigma_k\pi_k\cdots\sigma_1\pi_1\rho_1^\top\sigma_1^\top\cdots\rho_k^\top\sigma_k^\top\right)\right] - 1 \\ &= \frac{\langle\pi_1,\rho_1\rangle - 1}{N-1}\cdots\frac{\langle\pi_k,\rho_k\rangle - 1}{N-1}\cdot(\mathrm{Tr}(I) - 1)\end{aligned}$$

$$= \frac{1}{(N-1)^{k-1}} \prod_{i=1}^{k} (\langle \pi_k, \rho_k \rangle - 1),$$

and thus

$$\langle f_\pi, f_\rho \rangle = \frac{1}{N}\Big(\mathbb{E}_{\sigma_1,\ldots,\sigma_k}\Big[\mathrm{Tr}\left(\sigma_k \pi_k \cdots \sigma_1 \pi_1 \rho_1^\top \sigma_1^\top \cdots \rho_k^\top \sigma_k^\top\right)\Big] - 1\Big)$$
$$= \frac{1}{N(N-1)^{k-1}} \prod_{i=1}^{k} (\langle \pi_k, \rho_k \rangle - 1).$$

■

Remark 11 *For two cyclic permutation functions $f_\pi^{\mathrm{cyc}}, f_\rho^{\mathrm{cyc}}$, we define the inner product by*

$$\langle f_\pi^{\mathrm{cyc}}, f_\rho^{\mathrm{cyc}} \rangle := \mathbb{P}_{\sigma,i,x}(f_\pi(\sigma,i,x) = f_\rho(\sigma,i,x)) - \frac{1}{N} = \frac{1}{k}\sum_{i=1}^{k}\left(\frac{1}{N}\,\mathbb{E}_\sigma[\langle f_\pi^{(i)}(\sigma), f_\rho^{(i)}(\sigma)] - \frac{1}{N}\right),$$

where we have overloaded notation to let $f^{(i)}(\sigma)$ denote the permutation $\sigma_{i+k-1} \circ \pi_{i+k-1} \circ \cdots \circ \sigma_i \circ \pi_i$, so that $f^{(i)}(\sigma)(x) = f(\sigma, i, x)$. By Lemma 10,

$$\frac{1}{N}\,\mathbb{E}_\sigma[\langle f_\pi^{(i)}(\sigma), f_\rho^{(i)}(\sigma)\rangle] - \frac{1}{N} = \frac{1}{N(N-1)^{k-1}} \prod_{i=1}^{k} (\langle \pi_k, \rho_k \rangle - 1)$$

independent of the index i, and therefore $\langle f_\pi, f_\rho \rangle = \langle f_\pi^{\mathrm{cyc}}, f_\rho^{\mathrm{cyc}} \rangle$.

We will next construct a subset of $\mathcal{F}$ of nearly orthogonal functions. By the above remark, this also corresponds to a subset of $\mathcal{F}^{\mathrm{cyc}}$ of nearly orthogonal functions.

Lemma 12 (Nearly orthogonal subset) *Pick any $r \in [N]$. There exists a subset $\mathcal{F}_r \subset \mathcal{F}$ such that $|\mathcal{F}_r| \geq (r!/4)^{k/2}$, and for any $f, f' \in \mathcal{F}_r$ with $f \neq f'$, $|\langle f, f' \rangle| \leq (2r/N)^{k/2}$.*

Proof Let Π_r be a maximal packing of S_N, such that for any $\pi, \pi' \in \Pi_r$, $\langle \pi, \pi' \rangle < r$.

For a fixed permutation π, let us first count the number of permutations satisfying $\langle \pi, \pi' \rangle = i$. There are $\binom{N}{i}$ choices for which i elements are the same. Then, there are D_{S-i} ways to assign the rest of the permutation, where D_n counts the number of derangements on n elements. Therefore the number of such π' is $\binom{N}{i} D_{N-i} \leq \frac{N!}{i!(N-i)!} \cdot \left(\frac{(N-i)!}{e} + 1\right) = \frac{N!}{e \cdot i!} + \binom{N}{i}$. In total, the number of permutations that agree with π on more than r elements is at most

$$\sum_{i=r+1}^{N}\left(\frac{N!}{e \cdot i!} + \binom{N}{i}\right) = \frac{N!}{e \cdot r!}\sum_{i=r+1}^{N}\left(\frac{r!}{i!} + \frac{e r!}{i!(N-i)!}\right) \leq \frac{N!}{e \cdot r!}\sum_{i=r+1}^{N}\frac{1}{(i-r)!} \leq \frac{N!}{r!}.$$

Therefore if we construct Π_r greedily, we get that $M := |\Pi_r| \geq r!$.

We next construct a maximal packing $\mathcal{I}$ of $[M]^k$, such that for any $\alpha, \alpha' \in \mathcal{I}$, α and α' agree on at most $k/2$ coordinates. For any fixed $\alpha \in \mathcal{I}$, the number of tuples $\alpha' \in \mathcal{I}$ which agree with α on at least $k/2$ coordinates is

$$\sum_{j \le k/2} \binom{k}{j} (M-1)^j \le (M-1)^{k/2} 2^k.$$

Therefore if we construct $\mathcal{I}$ greedily, we get $|\mathcal{I}| \ge M^{k/2} 2^{-k}$.

We now construct the subset $\mathcal{F}_r$. Let the elements of Π_r be $\pi^{(1)}, \pi^{(2)}, \dots, \pi^{(M)}$ in some order. Define $\mathcal{F}_r := \{f_{(\pi^{(\alpha_1)}, \dots, \pi^{(\alpha_k)})} : \alpha \in \mathcal{I}\}$. Since any two elements of $\mathcal{I}$ differ in at least $k/2$ coordinates, and any two permutations in Π_r have at most r common values, for any $f, f' \in \mathcal{F}_r$, by Lemma 10 we have

$$\left|\langle f, f' \rangle\right| \le \frac{1}{N(N-1)^{k-1}} (N-1)^{k/2} (r-1)^{k/2} \le \left(\frac{r-1}{N-1}\right)^{k/2} \le \left(\frac{2r}{N}\right)^{k/2}.$$

Finally, we see that $|\mathcal{F}_r| = |\mathcal{I}| \ge (M/4)^{k/2} = (r!/4)^{k/2}$. ■

An immediate corollary of Lemma 12 is a bound on the packing number of $\mathcal{F}$.

Corollary 13 (Packing number) *There exists a subset $\mathcal{N} \subset \mathcal{F}$ of size $\log |\mathcal{N}| \gtrsim kN \log N$ such that, for any $f \ne f' \in \mathcal{N}$, $\|f - f'\|^2 \ge \frac{1}{2}$.*

Proof Since $\|f\|^2 = 1 - 1/N$ for all $f \in \mathcal{F}$, $|\langle f, f' \rangle| \le \varepsilon$ implies that $\|f - f'\|^2 \ge 2 - 2\varepsilon - 2/N \ge 1 - 2\varepsilon$. Select $\mathcal{N} = \mathcal{F}_r$ for $r = N/8$. Then $|\langle f, f' \rangle| \le (1/4)^{k/2} \le \frac{1}{4}$, so $\|f - f'\|^2 \ge 1/2$. Finally, we have that $\log \mathcal{N} \ge k/2 \cdot \log((N/8)!/4) \gtrsim kN \log N$, as desired. ■

We can now use standard SQ arguments to conclude the proof of Theorem 5. The following follows the proof of Lemma 2 in Damian et al. (2022) and Theorem 2 in Szörényi (2009).

Proof [Proof of Theorem 5] We will begin by showing the lower bound for the regular permutation composition task $\mathcal{F}$, and later show how the proof can be adapted to yield the same lower bound for the cyclic task $\mathcal{F}^{\text{cyc}}$.

We will show that there exist two functions $f, f' \in \mathcal{F}_r$ such that $|\mathbb{E}_{\sigma,x}[g_l(\sigma, x, f(\sigma, x))]| \le \tau$, $|\mathbb{E}_{\sigma,x}[g_i(\sigma, x, f'(\sigma, x))]| \le \tau$ for each query g_l made by the learner. As such, an adversary can respond with 0 to each query, and the learner will be unable to distinguish between f and f'. The learner must then incur a loss of at least

$$\begin{aligned}
\max\left(\mathbb{P}_{\sigma,x}(\hat{f}(\sigma, x) \ne f(\sigma, x)), \mathbb{P}_{\sigma,x}(\hat{f}(\sigma, x) \ne f'(\sigma, x))\right) &\ge \frac{1}{2}\mathbb{P}_{\sigma,x}(f'(\sigma, x) \ne f(\sigma, x)) \\
&= \frac{1}{2}\left(1 - \frac{1}{N} - \langle f, f' \rangle\right) \\
&= \Omega(1)
\end{aligned}$$

on either f or f'.

For the ith query g_l, define

$$\mathcal{A}_l^+ := \{f \in \mathcal{F}_r : \mathbb{E}_{\sigma,x}[g_i(\sigma, x, f(\sigma, x))] \geq \tau\}$$
$$\mathcal{A}_l^- := \{f \in \mathcal{F}_r : \mathbb{E}_{\sigma,x}[g_i(\sigma, x, f(\sigma, x))] \leq -\tau\}.$$

Let us overload notation to let $g_l : (S_N)^k \to \mathbb{R}^{N\times N}$ be the mapping with $g_l(\sigma)_{x,y} := g_l(\sigma, x, y)$. Each query can be written as

$$\mathbb{E}_{\sigma,x}[g_l(\sigma, x, f(\sigma, x))] = \frac{1}{N}\mathbb{E}_\sigma[\langle g_l(\sigma), f(\sigma)\rangle],$$

and thus we have that

$$\begin{aligned}
\left|\mathcal{A}_l^+\right|^2\tau^2 &\leq \left(\sum_{f\in\mathcal{A}_l^+} \mathbb{E}_{\sigma,x}[g_l(\sigma, x, f(\sigma, x))]\right) \\
&\leq \left(\frac{1}{N}\sum_{f\in\mathcal{A}_l^+} \mathbb{E}_\sigma[\langle g_l(\sigma), f(\sigma)\rangle]\right)^2 \\
&= \frac{1}{N^2}\mathbb{E}_\sigma\left[\left\langle g_l(\sigma), \sum_{f\in\mathcal{A}_l^+} f(\sigma)\right\rangle\right]^2 \\
&= \frac{1}{N^2}\mathbb{E}_\sigma\left[\left\langle g_l(\sigma), \sum_{f\in\mathcal{A}_l^+} \left(f(\sigma) - \frac{1}{N}1_N 1_N^\top\right)\right\rangle\right]^2,
\end{aligned}$$

where the last line uses the fact that g_l are mean zero. By Cauchy-Schwarz, we have

$$\begin{aligned}
\left|\mathcal{A}_l^+\right|^2\tau^2 &\leq \frac{1}{N^2}\mathbb{E}_\sigma\|g_l(\sigma)\|_F^2 \cdot \mathbb{E}_\sigma\left\|\sum_{f\in\mathcal{A}_l^+} \left(f(\sigma) - \frac{1}{N}1_N 1_N^\top\right)\right\|_F^2 \\
&\leq \frac{1}{N}\sum_{f,f'\in\mathcal{A}_l^+} \mathbb{E}_\sigma\left[\langle f(\sigma) - \frac{1}{N}1_N 1_N^\top, f'(\sigma) - \frac{1}{N}1_N 1_N^\top\rangle\right] \\
&= \frac{1}{N}\sum_{f,f'\in\mathcal{A}_l^+} (\mathbb{E}_\sigma\left[\langle f(\sigma), f'(\sigma)\rangle\right] - 1) \\
&= \sum_{f,f'\in\mathcal{A}_l^+} \langle f, f'\rangle \\
&\leq \left|\mathcal{A}_l^+\right| + (2r/N)^{k/2}(\left|\mathcal{A}_l^+\right|^2 - \left|\mathcal{A}_l^+\right|),
\end{aligned}$$

where the second inequality uses $\mathbb{E}_\sigma \|g_l(\sigma)\|_F^2 = \sum_{x,y\in[N]} \mathbb{E}_\sigma[g(\sigma, x, y)^2] = N$ from our choice of normalization. Altogether,

$$\left|\mathcal{A}_l^+\right| \leq \frac{1}{\tau^2 - (2r/N)^{k/2}}.$$

Similarly, $\left|\mathcal{A}_l^-\right| \le \frac{1}{\tau^2-(2r/N)^{k/2}}$. As such, the ith query can eliminate at most $\frac{2}{\tau^2-(2r/N)^{k/2}}$ elements of $\mathcal{F}_r$, and thus the learner must make at least

$$|\mathcal{F}_r| \cdot \frac{\tau^2-(2r/N)^{k/2}}{2} \ge (r!/4)^{k/2} \cdot \frac{\tau^2-(2r/N)^{k/2}}{2}$$

queries to recover f. Therefore we must either have tolerance $\tau^2 \le 2(2r/N)^{k/2}$, or make at least $\frac{1}{2}(r!/4)^{k/2} \cdot (2r/N)^{k/2}$ queries. Choosing $r = \log N$, we see that we must have tolerance

$$\tau^2 \le 2(2\log N/N)^{-k/2} \lesssim \frac{\log^{k/2} N}{N^{k/2}},$$

or make at least

$$\frac{1}{2}\left(\frac{(\log N)!\log N}{2N}\right)^{k/2} \gtrsim N^{k/2\cdot\log N}$$

queries.

We now consider the function class $\mathcal{F}^{\mathrm{cyc}}$. Let $\mathcal{F}_r^{\mathrm{cyc}}$ be the nearly orthogonal subset of $\mathcal{F}^{\mathrm{cyc}}$ constructed from Lemma 12. The queries are now of the form $g : \mathcal{X}^{\mathrm{cyc}} \times [N] \to \mathbb{R}$; to a query g the SQ oracle returns a response $\hat{q}$ with $|\hat{q} - \mathbb{E}_{\sigma,x,i}[g(\sigma,x,i,f(\sigma,x,i))]| \le \tau$. We assume that $\sum_{y\in N}\mathbb{E}_{\sigma,i,x}[g(\sigma,i,x,y)^2] = 1$, and $\mathbb{E}_\sigma[g(\sigma,i,x,y)] = 0$ for all $(i,x,y) \in [k]\times[N]\times[N]$.

The proof proceeds identically to the non-cyclic case. On the lth query g_l, define

$$\mathcal{A}_l^+ := \{f \in \mathcal{F}_r^{\mathrm{cyc}} : \mathbb{E}_{\sigma,x,i}[g_l(\sigma,x,i,f(\sigma,x,i))] \ge \tau\}$$
$$\mathcal{A}_l^- := \{f \in \mathcal{F}_r^{\mathrm{cyc}} : \mathbb{E}_{\sigma,x,i}[g_l(\sigma,x,i,f(\sigma,x,i))] \le -\tau\}.$$

We will overload notation to let $g_l^{(i)} : (S_N)^k \to \mathbb{R}^{N\times N}$ be defined as $g_l^{(i)}(\sigma)_{x,y} := g_l(\sigma,x,i,y)$. We then see each query is of the form

$$\mathbb{E}_{\sigma,x,i}[g_l(\sigma,x,i,f(\sigma,x,i))] = \frac{1}{Nk}\sum_{i=1}^k \mathbb{E}\left[\langle g_l^{(i)}(\sigma), f^{(i)}(\sigma)\rangle\right].$$

Then we can analogously bound

$$\begin{aligned}
\left|\mathcal{A}_l^+\right|^2\tau^2 &\le \left(\sum_{f\in\mathcal{A}_l^+}\mathbb{E}_{\sigma,i,x}[g_l(\sigma,i,x,f(\sigma,i,x))]\right) \\
&\le \frac{1}{N^2k^2}\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\langle g_l^{(i)}(\sigma), \sum_{f\in\mathcal{A}_l^+} f^{(i)}(\sigma)\right\rangle\right]^2 \\
&= \frac{1}{N^2k^2}\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\langle g_l^{(i)}(\sigma), \sum_{f\in\mathcal{A}_l^+}\left(f^{(i)}(\sigma) - \frac{1}{N}1_N1_N^\top\right)\right\rangle\right]^2,
\end{aligned}$$

and thus by Cauchy-Schwarz

$$\left|\mathcal{A}_l^+\right|^2\tau^2 \le N^{-2}k^{-2}\left(\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\|g_l^{(i)}(\sigma)\right\|_F^2\right]\right)\cdot\left(\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\|\sum_{f\in\mathcal{A}_l^+}\left(f^{(i)}(\sigma)-\frac{1}{N}1_N1_N^\top\right)\right\|_F^2\right]\right).$$

We see that

$$\begin{aligned}\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\|\sum_{f\in\mathcal{A}_l^+}\left(f^{(i)}(\sigma)-\frac{1}{N}1_N1_N^\top\right)\right\|_F^2\right] &= \sum_{i=1}^k\sum_{f,f'\in\mathcal{A}_l^+}\mathbb{E}_\sigma\left[\left\langle f^{(i)}(\sigma)-\frac{1}{N}1_N1_N^\top, f'^{(i)}(\sigma)-\frac{1}{N}1_N1_N^\top\right\rangle\right]\\ &= \sum_{f,f'\in\mathcal{A}_l^+}\sum_{i=1}^k\left(\mathbb{E}_\sigma[\langle f^{(i)}(\sigma), f'^{(i)}(\sigma)\rangle - 1]\right)\\ &= Nk\sum_{f,f'\in\mathcal{A}_l^+}\langle f, f'\rangle.\end{aligned}$$

Additionally,

$$\mathbb{E}_\sigma\left[\sum_{i=1}^k\left\|g_l^{(i)}(\sigma)\right\|_F^2\right] = Nk\sum_y\mathbb{E}_{\sigma,i,x}[g_l^{(i)}(\sigma,i,x,y)^2] \le Nk.$$

Combining these together, we have established $\left|\mathcal{A}_l^+\right|^2\tau^2 \le \sum_{f,f'\in\mathcal{A}_l^+}\langle f, f'\rangle$, an identical result as in the lower bound in the non-cyclic case. The remainder of the proof proceeds analogously. ■

Appendix C. Upper Bound: Analyzing the Gradient Dynamics

Throughout this section, we simply denote $X := X(\sigma), X_m := X(\sigma^{(m)}), \mathrm{hop}_i^r(\sigma, j) := \mathrm{hop}_i^r(j)$ when σ is clear from context for clarity. Note all the hop functions depend on the input σ.

C.1. Architecture

We use L-layer non-causal self-attention layers to learn this task, where $L = 1 + \log_2 k$. Define $X^{(0)} := X$. We will let $X^{(\ell)}$ denote the output of the ℓth layer of the ground truth transformer which exactly computes the k-fold composition, defined in Theorem 4. In particular, if $\{(W_{OV}^{*,(\ell)}, W_{KQ}^{*,(\ell)})\}_{\ell\in[L]}$ are the weights from Theorem 4, then

$$\begin{aligned}X^{(\ell)} &= X^{(\ell-1)} + f^{(\ell)}(X^{(\ell-1)})\\ f^{(\ell)}(X^{(\ell-1)}) &:= W_{OV}^{*,(\ell)}X^{(\ell-1)}\mathcal{S}({X^{(\ell-1)}}^\top W_{KQ}^{*,(\ell)}X^{(\ell-1)}).\end{aligned}$$

We refer to these $X^{(\ell)}$ as the *ideal inputs* to each layer. Moreover, given some parameter vector $\theta := \{(W_{OV}^{(\ell)}, W_{KQ}^{(\ell)})\}_{\ell\in[L]}$, define $\hat{X}^{(\ell)}$ to be the output of the ℓth layer of the transformer with parameters θ (where $\hat{X}^{(0)} = X^{(0)} = X$):

$$\hat{X}^{(\ell)} = \hat{X}^{(\ell-1)} + f^{(\ell)}(\hat{X}^{(\ell-1)})$$

$$f^{(\ell)}(\hat{X}^{(\ell-1)}) := W_{OV}^{(\ell)}\hat{X}^{(\ell-1)}\mathcal{S}(\hat{X}^{(\ell-1)\top}W_{KQ}^{(\ell)}\hat{X}^{(\ell-1)}).$$

Throughout this section, we will consider θ to be an intermediate of Algorithm 1, and note that $\hat{X}^{(\ell)}$ depends implicitly on the stage of the layerwise training algorithm. We further assume the value matrices $W_{OV}^{(\ell)}$ are fixed to their ground truth values $W_{OV}^{*,(\ell)}$, which only updates the ℓth hop's block. In the end, we unembed the final output and predict the hop using a readout layer Ψ_ℓ for the $2^{\ell-1}$-th hop. We define the final output as $\mathrm{TF}_\theta(X) := \hat{X}^{(L)}$.

The population loss function for the $2^{\ell-1}$-th hop is the cross-entropy loss

$$\mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) = -\,\mathbb{E}_{\sigma_1,\sigma_2,\cdots,\sigma_k,(i,j)}\left[\sum_{s'\in[N]}\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}\log\Big(\mathcal{S}(\Psi_\ell^\top\mathrm{TF}_\theta(X)_{(i,j)})_{s'}\Big)\right] \tag{4}$$

where $\theta = \Big(W_{KQ}^{(1)}, W_{KQ}^{(2)}, \cdots, W_{KQ}^{(L)}, \Psi_1, \cdots, \Psi_L\Big)$ are the trainable parameters. For clarity, we ignore the subscripts of the expectation. If the loss is based on finite samples, we define

$$\mathcal{L}^{(\ell)}(\theta) = -\frac{1}{M}\sum_{m=1}^{M}\left[\sum_{s'\in[N]}\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}\log\Big(\mathcal{S}(\Psi_\ell^\top\mathrm{TF}_\theta(X_m)_{(i,j)})_{s'}\Big)\right] \tag{5}$$

C.2. Gradient Computation

Formally, the model is initialized at $W_{KQ}^{(\ell)}(0) = 0_{d\times d}$ for all key-query matrices, and the readout/unembedding layer $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2}\otimes\mathbf{1}_k\otimes I_{N\times N}$ for small initialization scale β_0. We fix the value matrix for each layer as $W_{OV}^{(\ell)} = e_{L+2,\ell+2}e_{L+2,\ell+1}^\top\otimes I_{kN\times kN}$.

We expand the transformer into:

$$\mathrm{TF}_\theta(X) = X^{(0)} + \sum_{l=1}^{L} f^{(\ell)}(\hat{X}^{(\ell-1)})$$

By the initialization of Ψ_ℓ and $W_{OV}^{(\ell)}$ and the definition of the embedding, we have

$$(\Psi_{\ell'}^\top W_{OV}^{(\ell)})_{s'}{}^\top = \beta_0\delta_{\ell\ell'}\cdot e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes e_{N,s'}.$$

where $\delta_{ij}=1$ iff $i=j$. That means the projected output $\Psi_\ell^\top f^{(\ell)}$ is non-zero only in stage ℓ when training on loss $\mathcal{L}^{(\ell)}$. Thus, we only need to consider the gradient of $W_{KQ}^{(\ell')}$ ($\ell'\le\ell$) in stage ℓ with the loss $\mathcal{L}^{(\ell)}$.

Remark. Note that $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ is **not exactly zero** for $\ell'<\ell$, because $X^{(\ell-1)}$ depends on $W_{KQ}^{(\ell')}$. However, since the ℓ'th layer is already trained when $\ell'<\ell$, the softmax of that layer saturates. The Jacobian of the gradient is thus close to zero, making the update also close to zero and preserving the trained parameter $W_{KQ}^{(\ell')}$. It is formally proved in Section C.7. For simplicity, we only consider the main parts of the update in each stage ℓ, i.e. $\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{(\ell)}$, in the following argument.

Population dynamics. The population gradient of the loss of stage ℓ becomes:

$$\nabla \mathcal{L}_{\mathcal{D}}^{(\ell)} = -\mathbb{E}\left[\sum_{s'\in[N]}\Big(\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}-\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\Big)\nabla(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\right]$$

The model differential is

$$\mathrm{d}f_{(i,j)}^{(\ell)} = W_{OV}^{(\ell)}\hat{X}^{(\ell-1)}J^{(\ell)}(X,i,j)\hat{X}^{(\ell-1),\top}\mathrm{d}W_{KQ}^{(\ell)}\hat{X}_{(i,j)}^{(\ell-1)} + \mathrm{d}W_{OV}^{(\ell)}\hat{X}^{(\ell-1)}\mathcal{S}(\hat{X}^{(\ell-1)}W_{KQ}^{(\ell)}X_{(i,j)}^{(\ell-1)\top}),$$

where

$$\begin{aligned}J^{(\ell)}(X,i,j) &:= (\mathrm{diag}(\mathcal{S}^{(\ell)}(X,i,j)) - \mathcal{S}^{(\ell)}(X,i,j)(\mathcal{S}^{(\ell)}(X,i,j))^\top)\\ \mathcal{S}^{(\ell)}(X,i,j) &:= \mathcal{S}(\hat{X}^{(\ell-1)\top}W_{KQ}^{(\ell)}X_{(i,j)}^{(\ell-1)}).\end{aligned}$$

The gradient of the loss $\mathcal{L}^{(\ell)}$ with respect to $W_{KQ}^{(\ell)}$ is

$$\begin{aligned}&\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta)\\ &= -\mathbb{E}\left[\sum_{s'\in[N]}\Big(\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}-\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\Big)\nabla(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\right]\\ &= -\mathbb{E}\left[\sum_{s'\in[N]}\Big(\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}-\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\Big)\hat{X}^{(\ell-1)}J^{(\ell)}(X,i,j)\right.\\ &\qquad\left.\hat{X}^{(\ell-1)\top}(\Psi_\ell^\top W_{OV}^{(\ell)})_{s'}{}^\top\hat{X}_{(i,j)}^{(\ell-1)\top}\right]\end{aligned}$$

The gradient with respect to Ψ_ℓ is

$$\begin{aligned}\nabla_{\Psi_\ell}\mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) &= -\mathbb{E}\left[\sum_{s'\in[N]}\Big(\mathbf{1}\{s'=\mathrm{hop}_i^{2^{\ell-1}}(j)\}-\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\Big)\nabla_{\Psi_\ell}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})_{s'}\right]\\ &= -\mathbb{E}\left[\Big(e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}-\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})\Big)(f^{(\ell)}(\hat{X}^{(\ell-1)})_{(i,j)})^\top\right].\end{aligned}$$

Throughout the proof in the following sections, we will want to compute "exact gradients", where we assume that $\hat{X}^{(\ell')} = X^{(\ell')}$ for all $\ell' < \ell$.

Finally, we note that the initial output probabilities (after the $\mathrm{softmax}(\cdot)$) for the ideal input can be computed as

$$\begin{aligned}\mathcal{P}_{s'}^{(\ell)}(X,i,j) &:= \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X^{(\ell-1)})_{s'})\\ &= \mathcal{S}(\Psi_\ell^\top(X^{(\ell-1)} + W_{OV}^{(\ell)}X^{(\ell-1)}\mathcal{S}(X^{(\ell-1)^\top}W_{KQ}^{(\ell)}X_{(i,j)}^{(\ell-1)})))_{s'}\\ &= \mathcal{S}\left(0+\frac{\beta_0}{kN}\cdot(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes I_{N\times N})^\top X^{(\ell-1)}\mathbf{1}_{kN}\right)_{s'}\end{aligned}$$

Given that $X^{(\ell-1)}$ are ideal inputs for each layer ℓ, $(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes I_{N\times N})^\top X^{(\ell-1)}\mathbf{1}_{kN} = k\mathbf{1}_N$. That means the output probability $\mathcal{P}_{s'}(X,i,j)$ is $\frac{1}{N}$. We note that when the input indices (i,j) are clear from context (e.g. when the single query (i,j) is given), we ignore the (X,i,j) in both probability vectors $\mathcal{P}^{(\ell)}(X,i,j)$ and $\mathcal{S}^{(\ell)}(X,i,j)$ as well as the Jacobian $J^{(\ell)}(X,i,j)$.

C.3. Proof of Main Theorem

Here we restate the main theorem.

Theorem 14 (Guarantee for Algorithm 1) *Assume $M \geq \tilde{\Omega}(k^4N^6)$ and $\eta \geq \tilde{\Omega}(\frac{k^2N^3}{\beta_0}\log\frac{1}{\epsilon})$. For any ϵ satisfying $0 < \epsilon\log\frac{1}{\epsilon} \leq \tilde{O}(\frac{1}{k^6N^6})$ with probability 0.99, the final output $\hat{\theta}$ of Algorithm 1 satisfies that over any draw of the input permutation σ and the query index (i,j),*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_L^\top \mathrm{TF}_{\hat{\theta}}(X(\sigma))_{(i,j)}) - e_{\mathrm{hop}_i^k(j)} \right\|_\infty \leq \epsilon$$

Proof We provide an outline of the proof in this section.

Stage 1. We first prove that in **stage 1**, the first layer $W_{KQ}^{(1)}$ learns all the hidden permutations.

Using Lemma 15, after the first step gradient we have for any (i,j), the softmax probability satisfies

$$\mathcal{S}^{(1)}_{(i,\pi_i(j))} := \mathcal{S}(X^{(0)}W_{KQ}^{(1)}X^{(0)}_{(i,j)}) \geq 1 - \frac{1}{2}\epsilon.$$

We can then calculate the intermediate sequence $\hat{X}^{(1)}$:

$$\hat{X}^{(1)} = X^{(0)} + W_{OV}^{(1)}X^{(0)}\mathcal{S}^{(1)} = X^{(1)} + W_{OV}^{(1)}X^{(0)}(\mathcal{S}^{(1)} - \mathcal{S}^{(1)}_{\text{ideal}}).$$

where $\mathcal{S}_{\text{ideal}}$ is the ideal one-hot softmax attention score without the non-satuaration error. The ideal input sequence for the second layer should be

$$X^{(1)} = \left(\begin{array}{ccc|c|ccc} e_{k,1}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} & \cdots & e_{k,k}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} \\ e_{k,1}\otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(N)} \\ e_{k,1}\otimes e_{N,\sigma_1\pi_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1\pi_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(N)} \\ 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} \end{array}\right)$$

By Lemma 17, $\|X^{(1)} - \hat{X}^{(1)}\|_\infty \leq \epsilon$. By Lemma 18, we prove that after a large step of GD on Ψ_1 with $M \geq \tilde{\Omega}(k^4N^6)$, there is a probability at least $1 - 0.01/L$ that the transformer learns the 1-hop with ϵ error, thus completing the proof for stage 1.

Stage 2. By Lemma 19 we have after the second stage, for any (i,j)

$$\mathcal{S}^{(2)}_{(i+1,\mathrm{hop}_i^1(j))} := \mathcal{S}\Big((\hat{X}^{(1)})^\top W_{KQ}^{(2)}\hat{X}^{(1)}_{(i,j)}\Big) \geq 1 - \frac{1}{2}\epsilon.$$

The ideal input sequence for the third layer should be

$$X^{(2)} = \left(\begin{array}{ccc|c|ccc} e_{k,1}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} & \cdots & e_{k,k}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} \\ e_{k,1}\otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(N)} \\ e_{k,1}\otimes e_{N,\sigma_1\pi_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1\pi_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(N)} \\ e_{k,2}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,2}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,1}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,1}\otimes e_{N,\mathrm{hop}_k^2(N)} \\ 0_{(L-2)kN} & \cdots & 0_{(L-2)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-2)kN} \end{array}\right)$$

By Lemma 22, $\|X^{(2)} - \hat{X}^{(2)}\|_\infty \leq 2\epsilon$. By Lemma 23, we prove that with another large step of GD with $M \geq \tilde{\Omega}(k^4N^6)$, there is a probability at least $1 - 0.01/L$ the transformer learns the 2-hop with ϵ error. Thus we finish the proof for stage 2.

Stage ℓ. By Lemma 24, after training the key-query matrix for layer ℓ, we have

$$\mathcal{S}^{(\ell)}_{(i+2^{\ell-2},\mathrm{hop}_i^{2^{\ell-2}}(j))} := \mathcal{S}((\hat{X}^{(\ell-1)})^\top W_{KQ}^{(\ell)} \hat{X}^{(\ell-1)}_{(i,j)})_{(i+2^{\ell-2},\mathrm{hop}_i^{2^{\ell-2}}(j))} \geq 1 - \frac{1}{2}\epsilon.$$

The ideal input for each column (i, j) should be

$$X^{(\ell)}_{(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}_i^1(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ \vdots \\ e_{k,i+2^{\ell-1}-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} \\ 0_{(L-\ell)kN} \end{bmatrix}$$

By Lemma 27, $\|X^{(\ell)} - \hat{X}^{(\ell)}\|_\infty \leq \ell\epsilon$. By Lemma 28, we prove that with another large step of GD with $M \geq \tilde{\Omega}(k^4 N^6)$, there is a probability at least $1 - 0.01/L$ the transformer learns the $2^{\ell-1}$-hop with ϵ error. When $\ell = L$, the result implies that the transformer learns the k-fold composition task. The failure probability is upper bounded by 0.99 using union bound. Thus, we complete the proof.

■

In the following sections, we provide detailed proofs for the supplementary lemmas for each stage.

C.4. Stage 1: Learning the Hidden Permutations π_i

We first consider the gradient of the first layer. We show that during the first stage of training, $W_{KQ}^{(1)}$ learns all the hidden permutations π_i. As a result, the first layer attention predicts the correct one-hop $\mathrm{hop}_i^1(j) = \sigma_i\pi_i(j)$ for all (i, j).

Lemma 15 (Empirical gradient of $W_{KQ}^{(1)}$ learns all $\pi_i(\cdot)$ (Stage 1)) *Let $W_{KQ}^{(\ell)}(0) = 0_{d\times d}$ for all layers $l \in [L]$. After one-step of gradient descent on the first stage finite sample loss $\mathcal{L}^{(1)}$ with M training sequences and learning rate η, satisfying $\beta_0 \leq 1$, $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}, \eta \gtrsim \frac{k^2N^3\log\frac{kN}{\epsilon}}{\beta_0}$, then with probability $1 - \delta$ we have for any $(i, j) \in [k] \times [N]$,*

$$\mathcal{S}^{(1)}_{(i,\pi_i(j))} := \mathcal{S}(X^{(0)}W_{KQ}^{(1)}X^{(0)}_{(i,j)})_{(i,\pi_i(j))} \geq 1 - \frac{1}{2}\epsilon.$$

Furthermore, if we pick $\eta \gtrsim \frac{Ck^2N^3\log k\log\frac{kN}{\epsilon}}{\beta_0}$, we have $\mathcal{S}^{(1)}_{(i,\pi_i(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{CL}}$ for any absolute constant C.

Proof We first compute the population gradient, and then do the finite sample analysis to bound the noise of the empirical gradient. For simplicity, we ignore the superscipt $^{(0)}$ of the input sequence and simply denote the input sequence as X in the following calculation when it is clear from context.

Recall the gradient on the population loss is

$$\nabla_{W_{OV}^{(1)}} \mathcal{L}_{\mathcal{D}}^{(1)} = -\mathbb{E}\left[\sum_{s'\in[N]} \left(\mathbf{1}\{s' = \sigma_i\pi_i(j)\} - \mathcal{P}_{s'}^{(1)}\right) X J^{(1)} X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{s'}{}^\top X_{(i,j)}{}^\top\right]$$

Since the input for the first layer is exactly the ideal input, we have the output probability

$$\begin{aligned}
\mathcal{P}_{s'}^{(1)} &:= \mathcal{S}(\Psi_1^\top f^{(1)}(X^{(0)})_{(i,j)})_{s'} \\
&= \mathcal{S}(\Psi_1^\top (X^{(0)} + W_{OV}^{(1)} X^{(0)} \mathcal{S}(X^{(0)\top} W_{KQ}^{(1)} X_{(i,j)}^{(0)})))_{s'} \\
&= \mathcal{S}\left(0 + \frac{\beta_0}{kN} \cdot (e_{L+2,2} \otimes \mathbf{1}_k \otimes I_{N\times N})^\top X^{(0)} \mathbf{1}_{kN}\right)_{s'} \\
&= \mathcal{S}\left(\frac{\beta_0}{N}\mathbf{1}_N\right)_{s'} = \frac{1}{N}.
\end{aligned}$$

So the gradient becomes

$$\nabla_{W_{KQ}^{(1)}} \mathcal{L}_{\mathcal{D}}^{(1)}(\theta) = -\mathbb{E}\left[\sum_{s'\in[N]} \left(\mathbf{1}\{s' = \sigma_i\pi_i(j)\} - \frac{1}{N}\right) X J^{(1)} X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{s'}{}^\top X_{(i,j)}^\top\right]$$

We first notice that the normalization term vanishes due to the Jacobian:

$$\mathbb{E}\left[\sum_{s'\in[N]} \frac{1}{N} X J X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{s'}{}^\top X_{(i,j)}{}^\top\right] = \mathbb{E}\left[\frac{1}{N} X J \mathbf{1}_{kN} X_{(i,j)}{}^\top\right] = 0.$$

Therefore, the idealized gradient equals to the signal term:

$$\begin{aligned}
\nabla_{W_{KQ}^{(1)}} \mathcal{L}_{\mathcal{D}}^{(1)}(\theta) &= -\mathbb{E}\left[\sum_{s'\in[N]} \mathbf{1}\{s' = \sigma_i\pi_i(v)\} X J X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{s'}{}^\top X_{(i,j)}^\top\right] \\
&= -\frac{1}{kN}\mathbb{E}\left[X\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top\right) X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{\sigma_i\pi_i(v)}{}^\top X_{(i,j)}^\top\right].
\end{aligned}$$

Now the input sequence in this stage is

$$X^{(0)} = \left(\begin{array}{ccc|c|ccc} e_{k,1}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} & \cdots & e_{k,k}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} \\ e_{k,1}\otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(N)} \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \end{array}\right)$$

The following term indicates which coordinate in each block corresponds to the hop label $\sigma_i\pi_i(j)$:

$$\begin{aligned}
&X^\top \left(\Psi_1^\top W_{OV}^{(1)}\right)_{\sigma_i\pi_i(j)}{}^\top \\
&= \left(\begin{array}{ccc|c|ccc} e_{k,1}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} & \cdots & e_{k,k}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} \\ e_{k,1}\otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(N)} \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \end{array}\right)^\top \begin{bmatrix} 0_{kN} \\ \beta_0 \mathbf{1}_k \otimes e_{N,\sigma_i(\pi_i(j))} \\ 0_{knL} \end{bmatrix}
\end{aligned}$$

$$= \beta_0 \Big(e_{N,\sigma_1^{-1}\sigma_i\pi_i(j)}^\top \;\Big|\; e_{N,\sigma_2^{-1}\sigma_i\pi_i(j)}^\top \;\Big|\; \cdots \;\Big|\; e_{N,\sigma_k^{-1}\sigma_i\pi_i(j)}^\top \Big)^\top$$

Note that σ_i is a permutation for each block $j \in [k]$, so there is only one position mapping to $\sigma_i\pi_i(j)$. Since permutation is invertible, the coordinate is $\sigma_j^{-1}\sigma_i\pi_i(j)$.

With this expression, we can further simplify the gradient:

$$\begin{aligned}
\nabla \mathcal{L}_{\mathcal{D}}^{(1)}(\theta) &= -\frac{1}{kN}\,\mathbb{E}\left[X\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top \right) X^\top \left(\Psi_1^\top W_{OV}^{(1)} \right)_{\sigma_i\pi_i(j)}{}^\top X_{(i,j)}{}^\top \right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[X\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top \right) \left(e_{N,\sigma_1^{-1}\sigma_i\pi_i(j)}^\top \;\Big|\; \cdots \;\Big|\; e_{N,\sigma_k^{-1}\sigma_i\pi_i(j)}^\top \right)^\top X_{(i,j)}{}^\top \right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[X\left(\left(e_{N,\sigma_1^{-1}\sigma_i\pi_i(j)}^\top \;\Big|\; \cdots \;\Big|\; e_{N,\sigma_k^{-1}\sigma_i\pi_i(j)}^\top \right)^\top - \frac{1}{N}\mathbf{1} \right) X_{(i,j)}{}^\top \right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[\left(\begin{bmatrix} \sum_{p=1}^k e_{k,p}\otimes e_{N,\sigma_p^{-1}\sigma_i\pi_i(j)} \\ \sum_{p=1}^k e_{k,p}\otimes e_{N,\sigma_i\pi_i(j)} \\ 0_{kNL} \end{bmatrix} - \frac{1}{N}\begin{bmatrix} \mathbf{1}_{kN} \\ \mathbf{1}_{kN} \\ 0_{kNL} \end{bmatrix} \right) \begin{bmatrix} e_{k,i}\otimes e_{N,j} \\ e_{k,i}\otimes e_{N,\sigma_i(j)} \\ 0_{kNL} \end{bmatrix}^\top \right] \\
&:= -\frac{\beta_0}{kN}\,\mathbb{E}\begin{bmatrix} A_1 & A_2 & 0_{kN\times kNL} \\ A_3 & A_4 & 0_{kN\times kNL} \\ 0_{kNL\times kN} & 0_{kNL\times kN} & 0_{kNL\times kNL} \end{bmatrix} \qquad (A_1,\cdots,A_4 \in \mathbb{R}^{kN\times kN}.)
\end{aligned}$$

Suppose i is given. Since σ_p are independent of (i,j) and σ_i for $p \neq i$, the expectation

$$\mathbb{E}\left[(e_{k,p}\otimes e_{N,\sigma_p^{-1}\sigma_i\pi_i(j)}) \begin{bmatrix} e_{k,i}\otimes e_{N,v} \\ e_{k,i}\otimes e_{N,\sigma_i(v)} \end{bmatrix}^\top \Big| i, p\neq i \right] = \frac{1}{N^2}(e_{k,p}\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top$$

and cancels with the normalization term introduced by the jacobian.

Meanwhile, the i-th block recovers the adajency matrix of the permutation π_i:

$$\mathbb{E}\left[(e_{k,i}\otimes e_{N,\sigma_i^{-1}\sigma_i\pi_i(j)}) \begin{bmatrix} e_{k,i}\otimes e_{N,j} \\ e_{k,i}\otimes e_{N,\sigma_i(j)} \end{bmatrix}^\top \Big| i \right] = \frac{1}{N}\sum_{j=1}^N (e_{k,i}\otimes e_{N,\pi_i(j)})(e_{k,i}\otimes e_{N,j})^\top$$

Therefore, we have the first row of the block matrices

$$\mathbb{E}\left[A_1|i\right] = \frac{1}{N}\sum_{j=1}^N (e_{k,i}\otimes e_{N,\pi_i(j)})(e_{k,i}\otimes e_{N,j})^\top - \frac{1}{N^2}(e_{k,i}\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top,\ \mathbb{E}[A_2|i] = 0.$$

The overall expectation of A_1 is

$$\mathbb{E}[A_1] = \frac{1}{kN}\sum_{i=1}^k\sum_{j=1}^N (e_{k,i}\otimes e_{N,\pi_i(j)})(e_{k,i}\otimes e_{N,j})^\top - \frac{1}{kN^2}\sum_{i=1}^k (e_{k,i}\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top.$$

Similarly, the first expectation of the second row is $\mathbb{E}[A_3] = 0$. For A_4,

$$\mathbb{E}[A_4|i] = \sum_{p=1}^k \mathbb{E}[(e_{k,p}\otimes e_{N,\sigma_i\pi_i(j)})(e_{k,i}\otimes e_{N,\sigma_i(j)})^\top|i] - \frac{1}{N^2}(\mathbf{1}_k\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top$$

Note that given a fixed index $v \in [N]$, the expectation is $\frac{1}{N} I$ if $\pi_i(v) = v$, and $\frac{1}{N(N-1)} \mathbf{1}_N \mathbf{1}_N^\top - \frac{1}{N(N-1)} I_N$ when $\pi_i(v) \neq v$. Suppose $f(\pi_i)$ is the number of fixed point of π_i, we have

$$\mathbb{E}\left[A_4 | i\right] = (\mathbf{1}_k e_{k,i}^\top) \otimes \left(\frac{f(\pi_i) - 1}{N(N-1)} I_N + \frac{N - f(\pi_i)}{N^2(N-1)} \mathbf{1}_N \mathbf{1}_N^\top \right) - \frac{1}{N^2} (\mathbf{1}_k \otimes \mathbf{1}_N)(e_{k,i} \otimes \mathbf{1}_N)^\top.$$

The expectation for A_4 is

$$\mathbb{E}\left[A_4\right] = \frac{1}{k} \sum_{i=1}^{k} (\mathbf{1}_k e_{k,i}^\top) \otimes \left(\frac{f(\pi_i) - 1}{N(N-1)} I_N + \frac{N - f(\pi_i)}{N^2(N-1)} \mathbf{1}_N \mathbf{1}_N^\top \right) - \frac{1}{kN^2} (\mathbf{1}_k \otimes \mathbf{1}_N)(\mathbf{1}_k \otimes \mathbf{1}_N)^\top.$$

Now we consider the empirical estimate of the gradient. We define the matrix

$$g_1 = \mathbb{E} \begin{bmatrix} A_1 & A_2 & 0_{kN \times kNL} \\ A_3 & A_4 & 0_{kN \times kNL} \\ 0_{kNL \times kN} & 0_{kNL \times kN} & 0_{kNL \times kNL} \end{bmatrix}$$

The empirical gradient can be written as $\nabla_{W_{KQ}^{(1)}} \hat{\mathcal{L}}^{(1)} = -\frac{\beta_0}{kN} \hat{g}_1$, where $\hat{g}_1$ is the empirical estimate of g_1:

$$\hat{g}_1 = \frac{kN}{\beta_0 M} \sum_{m=1}^{M} \sum_{s' \in [N]} \mathbf{1}\{s' = \mathrm{hop}_{i_m}^1(j_m)\} X_m J X_m^\top \left(\Psi_1^\top W_{OV}^{(1)} \right)_{s'}^\top X_{(i_m, j_m)}^\top.$$

It suffices to show that $\|\hat{g}_1 - g_1\|_\infty$ is small, which is shown by the following lemma:

Lemma 16 *For any $\delta > 0$, we have that with probability $1 - \delta$,*

$$\|\hat{g}_1 - g_1\|_\infty \lesssim \frac{\log(kN/\delta)}{\sqrt{M}}$$

Proof First, the empirical gradient of the single sample X_m and (i_m, j_m) is

$$\begin{aligned} \hat{g}_{1,m} &= \frac{kN}{\beta_0} \sum_{s' \in [N]} \mathbf{1}\{s' = \mathrm{hop}_{i_m}^1(j_m)\} X_m J X_m^\top \left(\Psi_1^\top W_{OV}^{(1)} \right)_{s'}^\top X_{(i_m, j_m)}^\top \\ &= X_m (I - \frac{1}{kN} \mathbf{1}\mathbf{1}^\top) X_m^\top (e_{L+2,2} \otimes \mathbf{1}_k \otimes e_{N, \mathrm{hop}_{i_m}^1(j_m)}) X_{(i_m, j_m)}^\top \end{aligned}$$

The upper bound of the single entry of the random variable is

$$\begin{aligned} \|\hat{g}_{1,m}\|_\infty &\leq \left\| X_m (I - \frac{1}{kN} \mathbf{1}\mathbf{1}^\top) X_m^\top (e_{L+2,2} \otimes \mathbf{1}_k \otimes e_{N, \mathrm{hop}^1(j_m)}) \right\|_\infty \|X_{(i_m, j_m)}\|_\infty \\ &= \left\| X_m (I - \frac{1}{kN} \mathbf{1}\mathbf{1}^\top) X_m^\top (e_{L+2,2} \otimes \mathbf{1}_k \otimes e_{N, \mathrm{hop}^1(j_m)}) \right\|_\infty \qquad (\|X_{(i_m, j_m)}\|_\infty = 1.) \\ &= \left\| \left[\left(\begin{bmatrix} \sum_{j=1}^k e_{k,j} \otimes e_{N, \sigma_j^{-1} \mathrm{hop}^1(v_m)} \\ \sum_{j=1}^k e_{k,j} \otimes e_{N, \mathrm{hop}^1(j_m)} \\ 0_{kNL} \end{bmatrix} - \frac{1}{N} \begin{bmatrix} \mathbf{1}_{kN} \\ \mathbf{1}_{kN} \\ 0_{kNL} \end{bmatrix} \right) \right] \right\|_\infty \leq 1. \end{aligned}$$

Then we can concentrate $\hat{g}_1 = \frac{1}{M}\sum_{m=1}^{M} \hat{g}_{1,m}$ as:

$$\|\hat{g}_1 - g_1\|_\infty = \left\| \frac{1}{M}\sum_{m=1}^{M} \hat{g}_{1,m} - \mathbb{E}\,\hat{g}_1 \right\| \lesssim \frac{\log(d/\delta)}{\sqrt{M}}$$

with probability $1 - \frac{\delta}{d^2}$. Union bounding over all entries of $\hat{g}_1$ and we have the desired result since $d = O(kN\log k)$. ■

After the first step gradient, we get $W_{KQ}^{(1)} = \eta \hat{g}_1$. Now we compute the probability output of the softmax attention. Given a sample sequence X_m and index (i_m, j_m), we have the attention score

$$\eta X_m^\top \hat{g}_1 X_{(i_m,j_m)} = \frac{\beta_0 \eta}{kN}\left[X_m^\top g_1 X_{(i_m,j_m)} - X_m^\top \underbrace{(\hat{g}_1 - g_1)}_{\Delta g_1} X_{(i_m,j_m)} \right]$$

Note that the population attention score is

$$\begin{aligned} X_m^\top g_1 X_{(i_m,j_m)} &= X_m^\top \mathbb{E} \begin{bmatrix} A_1 & A_2 & 0_{kN\times kNL} \\ A_3 & A_4 & 0_{kN\times kNL} \\ 0_{kNL\times kN} & 0_{kNL\times kN} & 0_{kNL\times kNL} \end{bmatrix} \begin{bmatrix} e_{k,i_m} \otimes e_{N,j_m} \\ e_{k,i_m} \otimes e_{N,\sigma_{i_m}(j_m)} \\ 0_{kNL} \end{bmatrix} \\ &= X_m^\top \begin{bmatrix} \frac{1}{kN}\big(e_{k,i_m} \otimes e_{N,\pi_{i_m}(j_m)} - \frac{1}{N} e_{k,i_m} \otimes \mathbf{1}_N\big) \\ \frac{f(\pi_i)-1}{kN(N-1)}\big(\mathbf{1}_k \otimes e_{N,\sigma_{i_m}(j_m)} - \frac{1}{N}\mathbf{1}_k \otimes \mathbf{1}_N\big) \\ 0_{kNL} \end{bmatrix} \end{aligned}$$

Define the population/empirical separation between the correct position and the others as follows:

$$\Delta_{i_m,j_m}^{(1)} = \Big(X_m^\top g_1 X_{(i_m,j_m)}\Big)_{(i_m,\pi_{i_m}(j_m))} - \max_{p\neq(i_m,\pi_{i_m}(j_m))} \Big(X_m^\top g_1 X_{(i_m,j_m)}\Big)_p.$$

$$\hat{\Delta}_{i_m,j_m}^{(1)} = \Big(X_m^\top \hat{g}_1 X_{(i_m,j_m)}\Big)_{(i_m,\pi_{i_m}(j_m))} - \max_{p\neq(i_m,\pi_{i_m}(j_m))} \Big(X_m^\top \hat{g}_1 X_{(i_m,j_m)}\Big)_p.$$

If $\pi_{i_m}(v) = v$ for all $v \in [N]$, i.e. $f(\pi_{i_m}) = N$, we have the (i,j)-th entry of the pre-softmax attention score

$$\Big(X_m^\top g_1 X_{(i_m,j_m)}\Big)_{(i,j)} = \begin{cases} \frac{2N-2}{kN^2}, & i = i_m, j = j_m \\ -\frac{2}{kN^2}, & i = i_m,\ j \neq j_m \\ -\frac{1}{kN^2}, & i \neq i_m, \sigma_i(j) \neq \sigma_{i_m}(j_m) \\ \frac{N-1}{kN^2}, & i \neq i_m, \sigma_i(j) = \sigma_{i_m}(j_m) \end{cases}$$

So $\Delta_{i_m,j_m}^{(1)} \geq \frac{1}{kN^2}$ by $N \geq 2$.

If not all v satisfies $\pi_{i_m}(v) = v$, $f(\pi_i) \leq N-2$. We have the (i,j)-th entry of the pre-softmax attention score

$$\Big(X_m^\top g_1 X_{(i_m,j_m)}\Big)_{(i,j)} = \begin{cases} \frac{N-1}{kN^2}, & i = i_m, j = \pi_{i_m}(j_m) \\ \frac{f(\pi_{i_m})-1}{kN^2}, & \sigma_i(j) = \sigma_{i_m}(j_m) \\ -\frac{1}{kN^2} - \frac{f(\pi_{i_m})-1}{kN^2(N-1)}, & i = i_m,\ j \neq j_m, \pi_{i_m}(j_m) \\ -\frac{1}{kN^2}, & i \neq i_m, \sigma_i(j) \neq \sigma_{i_m}(j_m) \end{cases}$$

So $\Delta^{(1)}_{i_m,j_m} \geq \frac{1}{kN^2}$ by $N \geq 2$ for both cases. Therefore, we have the expected signal at least $\frac{1}{kN^2}$.

Now we bound the noise introduced by Δg_1. Since $\|\Delta g_1\|_\infty \lesssim \frac{\log \frac{d}{\delta}}{\sqrt{M}}$, we have

$$\left\| X_m^\top \Delta g_1 X_{(i_m,j_m)} \right\|_\infty \lesssim \frac{d \log \frac{d}{\delta}}{\sqrt{M}} \leq \frac{1}{2kN^2}$$

by $M \gtrsim k^2 N^4 d^2 \log^2 \frac{d}{\delta}$. Therefore, the noise term can be bounded and the empirical separation $\hat{\Delta}^{(1)}_{i_m,j_m} \geq \frac{1}{2kN^2}$. After one step gradient with learning rate $\eta \gtrsim \frac{k^2N^3}{\beta_0} \log \frac{kN}{\epsilon}$, the softmax output of the correct position can be lower bounded by

$$\mathcal{S}(X_m^\top W^{(1)}_{KQ} X_{(i_m,j_m)})_{i_m,\pi_{i_m}(j_m)} \geq \frac{\exp\left(\frac{\eta\beta_0}{kN} \cdot \hat{\Delta}^{(1)}_{i_m,j_m}\right)}{\exp\left(\frac{\eta\beta_0}{kN} \cdot \hat{\Delta}^{(1)}_{i_m,j_m}\right) + kN - 1} \geq 1 - \frac{1}{2}\epsilon.$$

If we increase the learning rate with $2C \log k$ times, we can accordingly get $\mathcal{S}^{(1)}_{(i,\pi_i(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{CL}}$ for some absolute constant C. ■

Perturbation analysis of the output After learning the first layer, we analyze the perturbation that non-saturation of the softmax prediction introduced to the ideal output.

Lemma 17 (Perturbation analysis of stage 1) *Under the conditions of Lemma 15, we have*

$$\|X^{(1)} - \hat{X}^{(1)}\|_\infty \leq \epsilon,$$

where $X^{(1)}$ is the ideal output with saturated softmax, and $\hat{X}^{(1)}$ is the true transformer output.

Proof After the first stage, the ideal input sequence for the second layer should be

$$X^{(1)} = \left(\begin{array}{ccc|c|ccc} e_{k,1} \otimes e_{N,1} & \cdots & e_{k,1} \otimes e_{N,N} & \cdots & e_{k,k} \otimes e_{N,1} & \cdots & e_{k,1} \otimes e_{N,N} \\ e_{k,1} \otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1} \otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k(N)} \\ e_{k,1} \otimes e_{N,\sigma_1\pi_1(1)} & \cdots & e_{k,1} \otimes e_{N,\sigma_1\pi_1(N)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k\pi_k(1)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k\pi_k(N)} \\ 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} \end{array}\right)$$

However, we should analyze the perturbation of the empirical sequence output $\hat{X}^{(1)}$ for the next layer analysis. Note that $X^{(1)} = X^{(0)} + W^{(1)}_{OV} X^{(0)} \mathcal{S}^{(1)}_{\text{ideal}}$, where $\mathcal{S}^{(1)}_{\text{ideal}}$ is the ideal one-hot softmax attention pattern. The empirical output $\hat{X}^{(1)}$ has some error introduced by the non-saturation of the softmax

$$\hat{X}^{(1)} = X^{(0)} + W^{(1)}_{OV} X^{(0)} \mathcal{S}^{(1)} = X^{(1)} + W^{(1)}_{OV} X^{(0)} \underbrace{(\mathcal{S}^{(1)} - \mathcal{S}^{(1)}_{\text{ideal}})}_{\Delta\mathcal{S}^{(1)}}.$$

By Lemma 15, the correct entry of the softmax probability vector $\mathcal{S}^{(1)}$ is greater than $1 - \frac{1}{2}\epsilon$ for all indices (i,j). And other probabilities has ϵ error in total, and they are all positive. Note that $\|X^{(0)}\|_\infty \leq 1$. As a result, the error of the input can be bounded as

$$\|X^{(1)} - \hat{X}^{(1)}\|_\infty = \max_{s,(i,j)} \left| (W^{(1)}_{OV} X^{(0)})_s^\top \Delta\mathcal{S}^{(1)}_{(i,j)} \right| \leq \|X^{(0)}\|_\infty \cdot 2 \cdot \frac{\epsilon}{2} \leq \epsilon.$$

So we conclude the proof. ■

At the end of Stage 1, we further train the readout layer with one gradient step to output the correct 1-hop for each position.

Lemma 18 *Under the conditions of Lemma 15, after one gradient step on Ψ_1 we have*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)}) - e_{\mathrm{hop}_i^1(j)} \right\|_\infty \le \epsilon.$$

Proof We calculate the population gradient for Ψ_1, and then do the finite sample analysis.

Recall the population gradient of the ℓth readout layer Ψ_ℓ:

$$\nabla_{\Psi_\ell} \mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) = -\mathbb{E}\left[\left(e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} - \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X^{(\ell-1)})_{(i,j)}) \right) (f^{(\ell)}(X^{(\ell-1)})_{(i,j)})^\top \right]$$

For 1-hop, $\ell = 1$ and we have the population gradient

$$\nabla_{\Psi_1} \mathcal{L}_{\mathcal{D}}^{(1)}(\theta) = -\mathbb{E}\left[\left(e_{N,\sigma_i\pi_i(j)} - \mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)}) \right) (f^{(1)}(X)_{(i,j)})^\top \right]$$

By Lemma 15, the output of the first layer would be

$$\begin{aligned} f^{(1)}(X)_{(i,j)} &= W_{OV}^{(1)} X^{(0)} \mathcal{S}^{(1)} = W_{OV}^{(1)} X^{(0)} \mathcal{S}_{\mathrm{ideal}}^{(1)} + W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)} \\ &= e_{L+2,3} \otimes e_{k,i} \otimes e_{N,\sigma_i\pi_i(j)} + W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)}. \end{aligned}$$

where $\|W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)}\|_\infty \le \epsilon$. Since $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2} \otimes \mathbf{1}_k \otimes I_{N\times N}$, we have the expansion

$$\begin{aligned} \mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)}) &= \mathcal{S}(\beta_0 e_{N,\sigma_i\pi_i(j)} + \Psi_1^\top W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)}) \\ &= \mathcal{S}(\beta_0 e_{N,\sigma_i\pi_i(j)}) + \underbrace{\tilde{J}\Psi_1^\top W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)}}_{\Delta_1}. \end{aligned}$$

Since $\|W_{OV}^{(1)} X^{(0)} \Delta\mathcal{S}^{(1)}\|_\infty \le \epsilon$, we have $\|\Delta_1\|_\infty \le \beta_0\epsilon$. The signal term

$$\mathcal{S}(\beta_0 e_{N,\sigma_i\pi_i(j)}) = \frac{\exp(\beta_0) - 1}{\exp(\beta_0) + N - 1} e_{N,\sigma_i\pi_i(j)} + \frac{1}{\exp(\beta_0) + N - 1} \mathbf{1}_N.$$

The population gradient is thus

$$\begin{aligned} \nabla_{\Psi_1} \mathcal{L}_{\mathcal{D}}^{(1)}(\theta) &= -\mathbb{E}\left[\left(e_{N,\sigma_i\pi_i(j)} - \mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)}) \right) (f^{(1)}(X)_{(i,j)})^\top \right] \\ &= -\mathbb{E}\left[\left(\frac{N}{\exp(\beta_0) + N - 1} e_{N,\sigma_i\pi_i(j)} - \frac{1}{\exp(\beta_0) + N - 1} \mathbf{1}_N - \Delta_1 \right) (f^{(1)}(X)_{(i,j)})^\top \right] \\ &= -\frac{1}{(\exp(\beta_0) + N - 1)k} \left(e_{L+2,3} \otimes \mathbf{1}_k \otimes (I_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^\top) \right) + O(\epsilon). \end{aligned}$$

where $O(\epsilon)$ denotes the terms with infinity norm smaller than ϵ with $\epsilon \lesssim \frac{1}{k^2 N^3}$.

Then we analyze the finite sample error. For any sample X_m, the upper bound of the empirical gradient for each sample

$$\nabla_{\Psi_1}\mathcal{L}^{(1)}(X_m) = -\Big[\Big(e_{N,\sigma_{i_m}\pi_{i_m}(j_m)} - \mathcal{S}(\Psi_1^\top f^{(1)}(X_m)_{(i_m,j_m)})\Big)(f^{(1)}(X_m)_{(i_m,j_m)})^\top\Big]$$

has infinity norm upper bounded by 1. Apply Hoeffding inequalities with $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}$, the empirical gradient has noise upper bounded by

$$\left\|\frac{1}{M}\sum_m \nabla_{\Psi_1}\hat{\mathcal{L}}^{(1)}(X_m) - \nabla_{\Psi_1}\mathcal{L}^{(1)}(X_m)\right\|_\infty \le \frac{\log(\frac{d}{\delta})}{\sqrt{M}} \lesssim \frac{1}{k^2N^3}.$$

Therefore, the error altogether is upper bounded by $O(\frac{1}{k^2N^3})$ in infinity norm.

After one step of gradient, we have the softmax score (Δ is the error term with $\|\Delta\|_\infty \le \epsilon$.)

$$\mathcal{S}(\Psi_1(1)^\top f^{(1)}(X)_{(i,j)}) = \mathcal{S}\bigg(\frac{\eta}{(\exp(\beta_0)+N-1)k}(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)e_{N,\sigma_i\pi_i(j)} + \beta_0 e_{N,\sigma_i\pi_i(j)} + \eta\Delta\bigg)$$

The separation between the $\sigma_i\pi_i(j)$-th entry and the others are lower bounded by:

$$\frac{\eta}{(\exp(\beta_0)+N-1)k}\frac{N-1}{N} - \eta\|\Delta\|_\infty \gtrsim \frac{\eta}{kN}.$$

By $\eta \gtrsim k^2N^3\log\frac{kN}{\epsilon}$, we have $\mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)})_{\sigma_i\pi_i(j)} \ge 1-\epsilon$ and thus

$$\sup_{\sigma,(i,j)}\left\|\mathcal{S}(\Psi_1^\top f^{(1)}(X)_{(i,j)}) - e_{\mathrm{hop}_i^1(j)}\right\|_\infty \le \epsilon.$$

■

C.5. Stage 2: Learning the 2-hop

Now we start to analyze the second stage GD with the input $\hat{X}^{(1)}$.

Lemma 19 (Empirical gradient of $W_{KQ}^{(2)}$ (Stage 2)) *Assume that $W_{KQ}^{(\ell)}(1) = 0_{d\times d}$ for all layers $\ell \ge 1$, and that $\|X^{(1)} - \hat{X}^{(1)}\|_\infty \le \epsilon$ before the second stage. After running one-step of gradient descent on the second stage finite sample loss $\mathcal{L}^{(2)}$ with M training sequences and learning rate η, satisfying $\beta_0 \le 1$, $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}, \eta \gtrsim \frac{k^2N^3\log\frac{kN}{\epsilon}}{\beta_0}$ for any $\epsilon \in (0, \frac{1}{k^2N^3\log^2 k})$, then with probability $1-\delta$, for any $(i,j)\in[k]\times[N]$, we have that after the second stage*

$$\mathcal{S}^{(2)}_{(i+1,\mathrm{hop}_i^1(j))} := \mathcal{S}\Big((\hat{X}^{(1)})^\top W_{KQ}^{(2)}\hat{X}^{(1)}_{(i,j)}\Big)_{(i+1,\mathrm{hop}_i^1(j))} \ge 1 - \frac{1}{2}\epsilon.$$

Furthermore, if we pick $\eta \gtrsim \frac{Ck^2N^3\log k\log\frac{kN}{\epsilon}}{\beta_0}$, we have that after the second stage $\mathcal{S}^{(2)}_{(i+1,\mathrm{hop}_i^1(j))} \ge 1 - \frac{\epsilon}{2(kNL)^{CL}}$ for any absolute constant C.

Proof We follow the strategy in stage 1, first computing the population gradient with the ideal input, then doing the finite sample analysis and finally controlling the perturbation of the input and the sample noise. We also ignore the superscript of $X^{(1)}/\hat{X}^{(1)}$ when it is clear from context in this subsection.

The population gradient of $W_{KQ}^{(2)}$ with ideal input sequence $X^{(1)}$ is

$$\nabla \mathcal{L}_{\mathcal{D}}^{(2)} = -\mathbb{E}\left[\sum_{s'\in[N]} \Big(\mathbf{1}\{s' = \mathrm{hop}_i^2(j)\} - (\mathcal{P}_{(i,j)}^{(2)})_{s'}\Big) X J^{(2)} X^\top (\Psi_2^\top W_{OV}^{(2)})_{s'}{}^\top (X_{(i,j)})^\top\right]$$

Since the normalization terms cancel out as in stage 1, the ideal gradient is the signal term:

$$\begin{aligned}\nabla \mathcal{L}_{\mathcal{D}}^{(2)} &= -\mathbb{E}\left[\sum_{s'\in[N]} \mathbf{1}\{s' = \sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(v)\} X J X^\top (\Psi_i^\top W_{OV}^{(i)})_{s'}{}^\top (X_{(i,j)}^{(1)})^\top\right] \\ &= -\frac{1}{kN}\mathbb{E}\left[X\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top\right) X^\top (\Psi_2^\top W_{OV}^{(2)})_{\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)}{}^\top (X_{(i,j)}^{(1)})^\top\right].\end{aligned}$$

We can also calculate the vector $(X^{(1)})^\top (\Psi_2^\top W_{OV}^{(2)})_{\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)}{}^\top$:

$$\left(e_{N,(\sigma_1\pi_1)^{-1}\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)}^\top \;\middle|\; e_{N,(\sigma_2\pi_2)^{-1}\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)}^\top \;\middle|\; \cdots \;\middle|\; e_{N,(\sigma_k\pi_k)^{-1}\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)}^\top\right)^\top$$

We further compute the gradient:

$$\begin{aligned}\nabla \mathcal{L}_{\mathcal{D}}^{(2)} &= -\frac{1}{kN}\mathbb{E}\left[(X^{(1)})^\top (\Psi_2^\top W_{OV}^{(2)})_{\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(v)}{}^\top - \frac{1}{N}\mathbf{1})(X_{(i,j)})^\top\right] \\ &= -\frac{\beta_0}{kN}\mathbb{E}\left[\left(\begin{bmatrix}\sum_{j=1}^k e_{k,j}\otimes e_{N,(\sigma_j\pi_j)^{-1}\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)} \\ \sum_{j=1}^k e_{k,j}\otimes e_{N,\sigma_j(\sigma_j\pi_j)^{-1}\sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j)} \\ \sum_{j=1}^k e_{k,j}\otimes \sigma_{i+1}\pi_{i+1}\sigma_i\pi_i(j) \\ \mathbf{1}_{kN(L-1)}\end{bmatrix} - \frac{1}{N}\begin{bmatrix}\mathbf{1}_{kN} \\ \mathbf{1}_{kN} \\ \mathbf{1}_{kN} \\ \mathbf{1}_{kN(L-1)}\end{bmatrix}\right)\begin{bmatrix}e_{k,i}\otimes e_{N,j} \\ e_{k,i}\otimes e_{N,\sigma_i(j)} \\ e_{k,i}\otimes e_{N,\sigma_i\pi_i(j)} \\ 0_{kN(L-1)}\end{bmatrix}^\top\right] \\ &:= -\frac{\beta_0}{kN}\begin{bmatrix}A_1 & A_2 & A_3 & 0 \\ A_4 & A_5 & A_6 & 0 \\ A_7 & A_8 & A_9 & 0 \\ 0 & 0 & 0 & 0\end{bmatrix} \qquad (A_i \in \mathbb{R}^{kN\times kN}, i = 1,2,...,9.)\end{aligned}$$

Similar to stage 1, since we have independent σ_{i+1} in the second and third row of block matrices, the expectation becomes 0 for A_1 and A_4 to A_9.

We focus on the calculation on A_2, A_3 and have the following expectation given (i,j) is in $[k]\times[N]$:

$$\mathbb{E}[A_3|i] = \frac{1}{N}\sum_{j=1}^N (e_{k,i+1}\otimes e_{N,j})(e_{k,i}\otimes e_{N,j})^\top - \frac{1}{N^2}(e_{k,i+1}\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top.$$

$$\mathbb{E}[A_2|i] = \frac{f(\pi_i)-1}{N(N-1)}\sum_{j=1}^N (e_{k,i+1}\otimes e_{N,j})(e_{k,i}\otimes e_{N,j})^\top - \frac{f(\pi_i)-1}{N^2(N-1)}(e_{k,i+1}\otimes \mathbf{1}_N)(e_{k,i}\otimes \mathbf{1}_N)^\top.$$

This population gradient can correctly learn the functionality of the second layer, and the signal can dominate the sum of the gradient noise and accumulated error. We will show this point after the noise/error analysis.

Now consider the empirical estimate of the gradient. Define the matrix

$$g_2 = \mathbb{E}\begin{bmatrix} A_1 & A_2 & A_3 & 0 \\ A_4 & A_5 & A_6 & 0 \\ A_7 & A_8 & A_9 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The empirical gradient can be written as $\nabla\hat{\mathcal{L}}^{(2)} = -\frac{\beta_0}{kN}\hat{g}_2$, where $\hat{g}_2$ is the empirical estimate of g_2 with perturbed inputs:

$$\hat{g}_2 = \frac{kN}{\beta_0 M}\sum_{m=1}^{M}\sum_{s'\in[N]}(\mathbf{1}\{s' = \mathrm{hop}^2_{i_m}(j_m)\} - \mathcal{P}^{(2)}_{s'})\hat{X}^{(1)}_m J \nabla_{W^{(\ell)}_{KQ}}\mathcal{L}^{(2)}(\hat{X}^{(1)}_m)^\top \left(\Psi_2^\top W^{(2)}_{OV}\right)_{s'}{}^\top (\hat{X}^{(1)}_{(i,j)})^\top$$

It suffices to show that $\|\hat{g}_2 - g_2\|_\infty$ is small. It is similar to the first stage analysis for the sample noise part, with the perturbed input error introduced:

Lemma 20 *Suppose $\|X^{(1)}_m - \hat{X}^{(1)}_m\|_\infty \le \epsilon$ for all m. For any $\delta > 0$, we have with probability $1-\delta$ s.t.*

$$\|\hat{g}_2 - g_2\|_\infty \lesssim \frac{\log(kN/\delta)}{\sqrt{M}} + d\epsilon$$

Proof First, the empirical gradient of the single ideal sample X_m is

$$\begin{aligned}\hat{g}^{\mathrm{ideal}}_{2,m} &= \frac{kN}{\beta_0}\sum_{s'\in[N]}\mathbf{1}\{s' = \mathrm{hop}^2_{i_m}(j_m)\}X_m J X_m^\top \left(\Psi_2^\top W^{(2)}_{OV}\right)_{s'}{}^\top (X^{(1)}_{(i_m,j_m)})^\top \\ &= X_m(I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,3}\otimes \mathbf{1}_k \otimes e_{N,\mathrm{hop}^2_{i_m}(j_m)})(X^{(1)}_{(i_m,j_m)})^\top\end{aligned}$$

Similar to the proof in Lemma 16, the upper bound of each entry of the random variable is

$$\|\hat{g}^{\mathrm{ideal}}_{2,m}\|_\infty \le \left\|X_m(I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,3}\otimes \mathbf{1}_k \otimes e_{N,\mathrm{hop}^2(v_m)})\right\|_\infty \|(X^{(1)}_{(i_m,j_m)})^\top\|_\infty \le 1. \quad (6)$$

Then we can concentrate $\hat{g}^{\mathrm{ideal}}_2 := \frac{1}{M}\sum_{m=1}^M \hat{g}^{\mathrm{ideal}}_{2,m}$ as:

$$\left\|\hat{g}^{\mathrm{ideal}}_2 - g_2\right\|_\infty = \left\|\frac{1}{M}\sum_{m=1}^{M}\hat{g}^{\mathrm{ideal}}_{2,m} - \mathbb{E}\,\hat{g}^{\mathrm{ideal}}_2\right\| \lesssim \frac{\log(d/\delta)}{\sqrt{M}}$$

with probability $1 - \frac{\delta}{d^2}$. By $d = O(kN\log k)$, we can union bound over all entries of $\hat{g}_2$ and we have the desired first term error.

After considering the ideal empirical gradient, we also need to bound the distance between the empirical gradients with ideal inputs X_m and those with perturbed input sequences $\hat{X}_m$.

Recall that the empirical gradient

$$\begin{aligned}\hat{g}_{2,m} &= \frac{kN}{\beta_0}\sum_{s'\in[N]}\Big(\mathbf{1}\{s'=\mathrm{hop}^2_{i_m}(j_m)\}-\mathcal{P}^{(2)}_{s'}\Big)\hat{X}_m J\hat{X}_m^\top\Big(\Psi_2^\top W_{OV}^{(2)}\Big)_{s'}{}^\top(\hat{X}^{(1)}_{(i_m,j_m)})^\top\\ &= \hat{X}_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,3}\otimes\mathbf{1}_k\otimes e_{N,\mathrm{hop}^2_{i_m}(j_m)})\hat{X}^\top_{(i_m,j_m)}\\ &- \sum_{s'\in[N]}\mathcal{S}_{s'}\hat{X}_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,3}\otimes\mathbf{1}_k\otimes e_{N,s'})\hat{X}^\top_{(i_m,j_m)}.\end{aligned}$$

Denote the following term as

$$\hat{\gamma}_{s',m} = \hat{X}_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,3}\otimes\mathbf{1}_k\otimes e_{N,s'})\hat{X}^\top_{(i_m,j_m)},$$

$$\gamma_{s',m} = X_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,3}\otimes\mathbf{1}_k\otimes e_{N,s'})X^\top_{(i_m,j_m)}.$$

Then we can rewrite the empirical gradient into

$$\hat{g}_{2,m}-\hat{g}^{\mathrm{ideal}}_{2,m} = \Delta\gamma_{\mathrm{hop}^2_{i_m}(j_m),m}-\sum_{s'\in[N]}\mathcal{S}_{s'}\Delta\gamma_{s',m}.$$

The error of the following difference $\|\gamma_{s',m}-\hat{\gamma}_{s',m}\|_\infty\le Cd\epsilon$ with some absolute constant for all possible $s'\in[N]$, since $\left\|\hat{X}_m-X_m\right\|_\infty\le\epsilon$ and $\|X_m\|_\infty\le 1$. Then, we have

$$\left\|\hat{g}_{2,m}-\hat{g}^{\mathrm{ideal}}_{2,m}\right\|_\infty \le \|\Delta\gamma_{\mathrm{hop}^2_{i_m}(j_m),m}\|_\infty+\sum_{s'\in[N]}\mathcal{S}_{s'}\left\|\Delta\gamma_{s',m}\right\|_\infty\le 2Cd\epsilon. \tag{7}$$

Combine both (6) and (7), we finished the proof. ■

After the one-step gradient, we get $W^{(2)}_{KQ}=\eta\hat{g}_2$. Given a sample sequence $\hat{X}^{(1)}_m$ and index (i_m,j_m), we have the attention score

$$\begin{aligned}\eta(\hat{X}^{(1)}_m)^\top\hat{g}_2\hat{X}^{(1)}_{(i_m,j_m)} = \frac{\beta_0\eta}{kN}\Bigg[&(X^{(1)}_m)^\top g_2X^{(1)}_{(i_m,j_m)}+(X^{(1)}_m)^\top\underbrace{(\hat{g}_2-g_2)}_{\Delta g_2}\hat{X}^{(1)}_{(i_m,j_m)}\Bigg]\\ &+\frac{\beta_0\eta}{kN}\underbrace{\Big[(\hat{X}^{(1)}_m)^\top\hat{g}_2\hat{X}^{(1)}_{(i_m,j_m)}-X_m^\top\hat{g}_2X_{(i_m,j_m)}\Big]}_{\text{Perturbation error}}\end{aligned}$$

Note that the population attention score is

$$X_m^\top g_2X_{(i_m,j_m)} = X_m^\top\mathbb{E}\begin{bmatrix}A_1&A_2&A_3&0\\A_4&A_5&A_6&0\\A_7&A_8&A_9&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}e_{k,i_m}\otimes e_{N,j_m}\\e_{k,i_m}\otimes e_{N,\sigma_{i_m}(j_m)}\\e_{k,i_m}\otimes e_{N,\sigma_{i_m}\pi_{i_m}(j_m)}\\0_{kN(L-1)}\end{bmatrix} = X_m^\top\begin{bmatrix}(*)\\0_{kN(L+1)}\end{bmatrix}$$

where the term $(*)$ is

$$\frac{1}{kN}\left(e_{k,i_m+1}\otimes e_{N,\sigma_{i_m}\pi_{i_m}(j_m)}-\frac{1}{N}e_{k,i_m+1}\otimes \mathbf{1}_N\right)+\frac{f(\pi_i)-1}{kN(N-1)}\left(e_{k,i_m+1}\otimes e_{N,\sigma_{i_m}(j_m)}-\frac{1}{N}e_{k,i_m+1}\otimes \mathbf{1}_N\right)$$

For simplicity, we ignore the superscript for the following proof in this subsection. Define the population/empirical separation between the correct position $p_2 := Ni_m + \sigma_{i_m}\pi_{i_m}(j_m)$, i.e. $(i_m + 1, \sigma_{i_m}\pi_{i_m}(j_m))$ position, and the others as follows:

$$\Delta^{(2)}_{i_m,j_m} = \left(X_m^\top g_2 X_{(i_m,j_m)}\right)_{p_2} - \max_{j\neq p_2}\left(X_m^\top g_2 X_{(i_m,j_m)}\right)_j.$$

$$\hat{\Delta}^{(2),\text{ideal}}_{i_m,j_m} = \left(X_m^\top \hat{g}_2 X_{(i_m,j_m)}\right)_{p_2} - \max_{j\neq p_2}\left(X_m^\top \hat{g}_2 X_{(i_m,j_m)}\right)_j.$$

$$\hat{\Delta}^{(2)}_{i_m,j_m} = \left(\hat{X}_m^\top \hat{g}_2 \hat{X}_{(i_m,j_m)}\right)_{p_2} - \max_{j\neq p_2}\left(\hat{X}_m^\top \hat{g}_2 \hat{X}_{(i_m,j_m)}\right)_j.$$

If $\pi_{i_m}(v) = v$ for all $v \in [N]$, i.e. $f(\pi_{i_m}) = N$, we have the $N(i-1)+j$-th entry of the pre-softmax attention score

$$\left(X_m^\top g_2 X_{(i_m,j_m)}\right)_{(i,j)} = \begin{cases} \frac{2N-2}{kN^2}, & i = i_m+1, j = \sigma_{i_m}\pi_{i_m}(j_m) \\ -\frac{2}{kN^2}, & i = i_m+1,\ j \neq \sigma_{i_m}\pi_{i_m}(j_m) \\ 0, & i \neq i_m \end{cases}$$

So $\Delta^{(2)}_{i_m,j_m} \geq \frac{1}{kN^2}$ by $N \geq 2$.

If not all v satisfies $\pi_i(v) = v$, $f(\pi_i) \leq N-2$. We have the $N(i-1)+j$-th entry of the pre-softmax attention score

$$\left(X_m^\top g_2 X_{(i_m,j_m)}\right)_{(i,j)} = \begin{cases} \frac{N-1}{kN^2}, & i = i_m+1, j = \sigma_{i_m}\pi_{i_m}(j_m) \\ \frac{f(\pi_{i_m})-1}{kN^2}, & i = i_m+1, j = \sigma_{i_m}(j_m) \\ -\frac{1}{kN^2} - \frac{f(\pi_{i_m})-1}{kN^2(N-1)}, & i = i_m+1,\ j \neq \sigma_{i_m}\pi_{i_m}(j_m), \sigma_{i_m}(j_m) \\ 0, & i \neq i_m+1 \end{cases}$$

So $\Delta^{(2)}_{i_m,j_m} \geq \frac{1}{kN^2}$ by $N \geq 2$ for both cases. Therefore, we have the expected signal at least $\frac{1}{kN^2}$.

Now we bound the noise introduced by Δg_2. Since $\|\Delta g_2\|_\infty \lesssim \frac{\log\frac{d}{\delta}}{\sqrt{M}}$, we have

$$\left\|X_m^\top \Delta g_2 X_{(i_m,j_m)}\right\|_\infty \lesssim \frac{d\log\frac{d}{\delta}}{\sqrt{M}} \leq \frac{1}{2kN^2}$$

by $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}$. Therefore, the noise term can be bounded and the empirical separation with the ideal input sequence $\hat{\Delta}^{(2),\text{ideal}}_{i_m,j_m} \geq \frac{1}{2kN^2}$. This also gives the upper bound for $\|\hat{g}_2\|_\infty \leq O(1/kN^2)$.

Finally, we add up the perturbation error for the pre-softmax attention score. However, the first layer parameter is also updated by a very small amount in the second stage. So we need to upper bound the perturbation again for $\hat{X}^{(1)}$ for the second stage conclusion. The following lemma shows that after the second gradient step, the first-layer parameter is almost unperturbed and the softmax score is still close to one-hot.

Lemma 21 ($W_{KQ}^{(1)}$ is unchanged) *Under the condition of Lemma 19, after the second step we still have*

$$\mathcal{S}^{(1)}_{(i,\pi_i(j))} := \mathcal{S}(X^{(0)}W_{KQ}^{(1)}X^{(0)}_{(i,j)})_{(i,\pi_i(j))} \geq 1 - \frac{1}{2}\epsilon.$$

Proof By Lemma 29, the gradient norm for the first layer is upper bounded by $O(\beta_0 k^{5/2}N^{3/2}L^{3/2}\epsilon)$. With the gradient update in the second stage, $W_{KQ}^{(1)}$ will only be perturbed by

$$\left\|\eta\nabla_{W_{KQ}^{(1)}}\mathcal{L}^{(2)}\right\| \lesssim (kN)^{4.5}\log\frac{kN}{\epsilon}\log^{2.5}k\cdot\epsilon \ll N\log\frac{kN}{\epsilon}.$$

where $N\log\frac{kN}{\epsilon}$ is the scale of each entry of $W_{KQ}^{(1)}$. Therefore, the previous bound on the attention score in Lemma 15 still holds. ■

This implies that $\left\|\hat{X}_m - X_m\right\|_\infty \leq \epsilon$ still holds using the same perturbation argument in stage 1. Therefore, we have the perturbation error to the pre-softmax attention score

$$\begin{aligned}
&\left\|(\hat{X}_m^{(1)})^\top\hat{g}_2\hat{X}^{(1)}_{(i_m,j_m)} - X_m^\top\hat{g}_2X_{(i_m,j_m)}\right\|_\infty \\
\leq&\left\|(\hat{X}_m^{(1)})^\top\hat{g}_2(\hat{X}^{(1)}_{(i_m,j_m)} - X_{(i_m,j_m)})\right\|_\infty + \left\|(\hat{X}_m^{(1)} - X_m)^\top\hat{g}_2X_{(i_m,j_m)}\right\|_\infty \\
\leq&\|\hat{g}_2\|_2\left\|(\hat{X}_m^{(1)})^\top(\hat{X}^{(1)}_{(i_m,j_m)} - X_{(i_m,j_m)})\right\|_\infty + \|\hat{g}_2\|_2\left\|(\hat{X}_m^{(1)} - X_m)^\top X_{(i_m,j_m)}\right\|_\infty \\
\leq&\|\hat{g}_2\|_F\left\|(\hat{X}_m^{(1)})^\top(\hat{X}^{(1)}_{(i_m,j_m)} - X_{(i_m,j_m)})\right\|_\infty + \|\hat{g}_2\|_F\left\|(\hat{X}_m^{(1)} - X_m)^\top X_{(i_m,j_m)}\right\|_\infty \\
\lesssim&\sqrt{\frac{1}{(kN^2)^2}\cdot d^2}\cdot d\epsilon.
\end{aligned}$$

Since $\epsilon \leq O(\frac{1}{d^2}) = O(\frac{1}{k^2N^3L^2})$, the perturbation error is upper bounded by $O(\frac{1}{kN^2})$. Therefore, the empirical separation $\hat{\Delta}^{(2)}_{i_m,j_m} \geq \Omega(\frac{1}{kN^2})$. After one step gradient with learning rate $\eta \gtrsim \frac{k^2N^3}{\beta_0}\log\frac{kN}{\epsilon}$, the softmax output of the correct position can be lower bounded by

$$\mathcal{S}(\hat{X}_m^\top W_{KQ}^{(2)}\hat{X}_{(i_m,j_m)})_{i_m,\mathrm{hop}^1(X_{(i_m,j_m)})} \geq \frac{\exp\left(\frac{\eta\beta_0}{kN}\cdot\frac{1}{kN^2}\right)}{\exp\left(\frac{\eta\beta_0}{kN}\cdot\frac{1}{kN^2}\right) + kN - 1} \geq 1 - \frac{1}{2}\epsilon.$$

Therefore, we finish the proof. ■

Perturbation analysis of Stage 2 The lemma presents the perturbation analysis for stage 2.

Lemma 22 (Perturbation analysis of stage 2) *Under the condition of Lemma 19, we have*

$$\|X^{(2)} - \hat{X}^{(2)}\| \leq 2\epsilon,$$

where $X^{(2)}$ is the ideal output with saturate softmax, and $\hat{X}^{(2)}$ is the transformer output.

Proof Similar to stage 1, the ideal output for the second layer is

$$X^{(2)} = X^{(1)} + W_{KQ}^{(2)} X^{(1)} \mathcal{S}_{\text{ideal}}^{(2)},$$

where $\mathcal{S}_{\text{ideal}}^{(2)}$ is the ideal one-hot softmax attention pattern. The empirical output $\hat{X}^{(2)}$ has the error introduced by the non-saturation of the softmax, together with the previous error in $\hat{X}^{(1)}$:

$$\begin{aligned}\hat{X}^{(2)} &= \hat{X}^{(1)} + W_{OV}^{(2)} \hat{X}^{(1)} \mathcal{S}^{(2)} \\ &= X^{(2)} + W_{OV}^{(2)} X^{(1)} \underbrace{(\mathcal{S}^{(2)} - \mathcal{S}_{\text{ideal}}^{(2)})}_{\Delta\mathcal{S}^{(2)}\text{, Non-saturation error}} + \underbrace{(\hat{X}^{(1)} - X^{(1)}) + W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}) \mathcal{S}^{(2)}}_{\text{Accumulated perturbation error}}.\end{aligned}$$

We first bound the non-saturation error term. By Lemma 19, the correct entry of the softmax probability vector $\mathcal{S}^{(2)}$ is greater than $1 - \frac{1}{2}\epsilon$ for all index v. And other probabilities have ϵ error in total, and they are all positive. Since $\|X^{(1)}\|_\infty \le 1$, the error of the non-saturation error can be bounded as

$$\|W_{OV}^{(2)} X^{(1)} (\mathcal{S}^{(2)} - \mathcal{S}_{\text{ideal}}^{(2)})\|_\infty = \max_{s,(i,j)} \left| (W_{OV}^{(2)} X^{(1)})_s^\top \Delta\mathcal{S}_{(i,j)}^{(2)} \right| \le \|X^{(1)}\|_\infty \cdot 2 \cdot \frac{\epsilon}{2} \le \epsilon.$$

Now consider the accumulated perturbation error. By the perturbation analysis in stage 1, we have $\left\| \hat{X}^{(1)} - X^{(1)} \right\|_\infty \le \epsilon$. Note that the error in $\hat{X}^{(1)} - X^{(1)}$ are in different rows of the matrices from $W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}) \mathcal{S}^{(2)}$. Therefore, $\hat{X}^{(1)} - X^{(1)}$ won't introduce extra error in this stage, and we only need to consider $W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}) \mathcal{S}^{(2)}$:

$$\begin{aligned}\left\| W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}) \mathcal{S}^{(2)} \right\|_\infty &= \max_{s,(i,j)} \left| (W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}))_s^\top \mathcal{S}_{(i,j)}^{(2)} \right| \\ &= \max_{s,(i,j)} \left| \sum_{p=1}^{kN} (W_{OV}^{(2)} (\hat{X}^{(1)} - X^{(1)}))_{s,p} (\mathcal{S}_{(i,j)}^{(2)})_p \right| \\ &\le \|\hat{X}^{(1)} - X^{(1)}\|_\infty \qquad (\text{Since } \textstyle\sum_p (\mathcal{S}_{(i,j)}^{(2)})_p = 1, (\mathcal{S}_{(i,j)}^{(2)})_p \ge 0.)\end{aligned}$$

Combine both parts of error, we have $\left\| \hat{X}^{(2)} - X^{(2)} \right\|_\infty \le 2\epsilon$. ■

In later stages, we use a similar argument and inductively show that the perturbation error grows with the depth ℓ. At the end of stage 2, we further train the readout layer with one gradient step to output the correct 2-hop for each position.

Lemma 23 *Under the conditions of Lemma 19, after one gradient step on Ψ_2 we have*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_2^\top f^{(2)}(\hat{X}^{(1)})_{(i,j)}) - e_{\text{hop}_i^2(j)} \right\|_\infty \le \epsilon.$$

Proof We calculate the population gradient for Ψ_2, and then do the finite sample analysis.

For 2-hop, the population gradient is

$$\nabla_{\Psi_2}\mathcal{L}_{\mathcal{D}}^{(2)}(\theta) = -\mathbb{E}\left[\left(e_{N,\mathrm{hop}_i^2(j)} - \mathcal{S}(\Psi_2^\top f^{(2)}(X)_{(i,j)})\right)(f^{(2)}(X)_{(i,j)})^\top\right]$$

By Lemma 19, the output of the second layer would be

$$\begin{aligned} f^{(2)}(X)_{(i,j)} &= W_{OV}^{(2)}\hat{X}^{(1)}\mathcal{S}^{(2)} = W_{OV}^{(2)}X^{(1)}\mathcal{S}_{\mathrm{ideal}}^{(2)} + W_{OV}^{(2)}(\hat{X}^{(1)} - X^{(1)})\mathcal{S}_{\mathrm{ideal}}^{(2)} + W_{OV}^{(2)}X^{(1)}\Delta\mathcal{S}^{(2)} \\ &= e_{L+2,4}\otimes e_{k,i}\otimes e_{N,\mathrm{hop}_i^2(j)} + \Delta_2. \end{aligned}$$

where $\|\Delta_2\|_\infty \le 2\epsilon$ by Lemma 22. Since $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2}\otimes \mathbf{1}_k \otimes I_{N\times N}$, we have the expansion

$$\mathcal{S}(\Psi_2^\top f^{(2)}(X)_{(i,j)}) = \mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^2(j)} + \Psi_2^\top\Delta_2) = \mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^2(j)}) + \tilde{J}\Psi_2^\top\Delta_2.$$

Since $\|\Delta_2\|_\infty \le \epsilon$, we have $\left\|\tilde{J}\Psi_2^\top\Delta_2\right\|_\infty \le \beta_0\epsilon$. The signal term

$$\mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^2(j)}) = \frac{\exp(\beta_0)-1}{\exp(\beta_0)+N-1}e_{N,\mathrm{hop}_i^2(j)} + \frac{1}{\exp(\beta_0)+N-1}\mathbf{1}_N.$$

The population gradient is thus

$$\begin{aligned} \nabla_{\Psi_2}\mathcal{L}_{\mathcal{D}}^{(2)}(\theta) &= -\mathbb{E}\left[\left(\frac{N}{\exp(\beta_0)+N-1}e_{N,\mathrm{hop}_i^2(j)} - \frac{1}{\exp(\beta_0)+N-1}\mathbf{1}_N - \tilde{J}\Psi_2^\top\Delta_2\right)(f^{(2)}(X)_{(i,j)})^\top\right] \\ &= -\frac{1}{(\exp(\beta_0)+N-1)k}\left(e_{L+2,4}\otimes\mathbf{1}_k\otimes(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)\right) + O(\epsilon). \end{aligned}$$

where $O(\epsilon)$ denotes the terms with infinity norm smaller than ϵ with $\epsilon \lesssim \frac{1}{k^2N^3L^2}$.

Then we analyze the finite sample error. For any sample X_m, the upper bound of the empirical gradient for each sample

$$\nabla_{\Psi_2}\mathcal{L}^{(2)}(X_m) = -\left[\left(e_{N,\mathrm{hop}_{i_m}^2(j_m)} - \mathcal{S}(\Psi_2^\top f^{(2)}(X_m)_{(i_m,j_m)})\right)(f^{(2)}(X_m)_{(i_m,j_m)})^\top\right]$$

has infinity norm upper bounded by 1. Apply Hoeffding inequalities with $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}$, the empirical gradient has noise upper bounded by

$$\left\|\frac{1}{M}\sum_m \nabla_{\Psi_2}\hat{\mathcal{L}}^{(2)}(X_m) - \nabla_{\Psi_2}\mathcal{L}^{(2)}(X_m)\right\|_\infty \le \frac{\log(\frac{d}{\delta})}{\sqrt{M}} \lesssim \frac{1}{k^2N^3}.$$

Therefore, the error altogether is upper bounded by $O(\frac{1}{k^2N^3})$ in infinity norm.

After one step of gradient, we have the softmax score (Δ is the error term with $\|\Delta\|_\infty \le \epsilon$.)

$$\mathcal{S}(\Psi_2(1)^\top f^{(2)}(X)_{(i,j)})) = \mathcal{S}\left(\frac{\eta}{(\exp(\beta_0)+N-1)k}(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)e_{N,\mathrm{hop}_i^2(j)} + \beta_0 e_{N,\mathrm{hop}_i^2(j)} + \eta\Delta\right)$$

The separation between the $\mathrm{hop}_i^2(j)$-th entry and the others are lower bounded by:

$$\frac{\eta}{(\exp(\beta_0)+N-1)k}\frac{N-1}{N} - \eta\|\Delta\|_\infty \gtrsim \frac{\eta}{kN}.$$

By $\eta \gtrsim k^2N^3 \log \frac{kN}{\epsilon}$, we have $\mathcal{S}(\Psi_2^\top f^{(2)}(X)_{(i,j)})_{\mathrm{hop}_i^2(j)} \geq 1-\epsilon$ and thus

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_2^\top f^{(2)}(X)_{(i,j)}) - e_{\mathrm{hop}_i^2(j)} \right\|_\infty \leq \epsilon.$$

■

C.6. Stage ℓ ($3 \leq \ell \leq 1+\log_2 k$): Learning the $2^{\ell-1}$-hop

Similar to the second stage, we can learn the ℓth layer by one step of gradient descent, shown in the following lemma.

Lemma 24 (Empirical gradient of $W_{KQ^{(\ell)}}$ (Stage ℓ)) *Suppose the input sequence $\hat{X}_m^{(\ell-1)}$ for the ℓth layer satisfies $\|\hat{X}_m^{(\ell-1)} - X_m^{(\ell-1)}\|_\infty \leq (\ell-1)\epsilon$ before stage ℓ where $X_m^{(\ell-1)}$ is the ideal sequence. Assume that $W_{KQ}^{(\ell')}(\ell-1) = 0_{d\times d}$ for all layers $\ell' \geq \ell$. After one step of gradient descent on the ℓth stage finite sample loss $\mathcal{L}^{(\ell)}$ with M training sequences and learning rate η, satisfying $\beta_0 \leq 1$, $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}, \eta \gtrsim \frac{k^2N^3\log\frac{kN}{\epsilon}}{\beta_0}$ for any $\epsilon \in (0, \frac{1}{k^2N^3\log^2 k})$, then for any $(i,j) \in [k]\times[N]$, after stage ℓ we have*

$$\mathcal{S}^{(\ell)}_{(i+2^{\ell-2},\mathrm{hop}_i^{2^{\ell-2}}(j))} := \mathcal{S}((\hat{X}^{(\ell-1)})^\top W_{KQ}^{(\ell)} \hat{X}^{(\ell-1)}_{(i,j)})_{(i+2^{\ell-2},\mathrm{hop}_i^{2^{\ell-2}}(j))} \geq 1 - \frac{1}{2}\epsilon.$$

Furthermore, if we pick $\eta \gtrsim \frac{Ck^2N^3\log k\log\frac{kN}{\epsilon}}{\beta_0}$, we have $\mathcal{S}^{(\ell)}_{(i+2^{\ell-2},\mathrm{hop}_i^{2^{\ell-2}}(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{CL}}$ for any absolute constant C.

Proof We follow the first and second stage strategy, first computing the population gradient and then doing the finite-sample analysis and controlling the perturbation error. Similarly, we ignore the subscript of $X^{(\ell-1)}$ when it is clear from context in this subsection.

The ideal input for each layer $\ell(\ell \geq 3)$ is in the following form:

$$X^{(\ell-1)} = \left(\begin{array}{ccc|c|ccc} e_{k,1}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} & \cdots & e_{k,k}\otimes e_{N,1} & \cdots & e_{k,1}\otimes e_{N,N} \\ e_{k,1}\otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k(N)} \\ e_{k,1}\otimes e_{N,\sigma_1\pi_1(1)} & \cdots & e_{k,1}\otimes e_{N,\sigma_1\pi_1(N)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(1)} & \cdots & e_{k,k}\otimes e_{N,\sigma_k\pi_k(N)} \\ e_{k,2}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,2}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,1}\otimes e_{N,\mathrm{hop}_1^2(1)} & \cdots & e_{k,1}\otimes e_{N,\mathrm{hop}_k^2(N)} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ e_{k,2^{\ell-2}}\otimes e_{N,\mathrm{hop}_1^{2^{\ell-2}}(1)} & \cdots & e_{k,2^{\ell-2}}\otimes e_{N,\mathrm{hop}_1^{2^{\ell-2}}(N)} & \cdots & e_{k,k+2^{\ell-2}-1}\otimes e_{N,\mathrm{hop}_k^{2^{\ell-2}}(1)} & \cdots & e_{k,k+2^{\ell-2}-1}\otimes e_{N,\mathrm{hop}_k^{2^{\ell-2}}(N)} \\ 0_{(L-\ell+1)kN} & \cdots & 0_{(L-\ell+1)kN} & \cdots & 0_{(L-\ell+1)kN} & \cdots & 0_{(L-\ell+1)kN} \end{array}\right)$$

We have the population gradient

$$\begin{aligned}\nabla\mathcal{L}^{(\ell)} &= -\mathbb{E}\left[\sum_{s'\in[N]} \mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} X^{(\ell-1)} J^{(\ell)} (X^{(\ell-1)})^\top (\Psi_\ell^\top W_{OV}^{(\ell)})_{s'}{}^\top (X^{(\ell-1)}_{(i,j)})^\top\right] \\ &= -\frac{1}{kN}\mathbb{E}\left[X^{(\ell-1)}\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top\right)(X^{(\ell-1)})^\top (\Psi_\ell^\top W_{OV}^{(\ell)})_{\mathrm{hop}_i^{2^{\ell-1}}}{}^\top X_{(i,j)}^\top\right].\end{aligned}$$

Similar to previous stages, we can directly calculate $(X^{(\ell-1)})^\top (\Psi_\ell^\top W_{OV}^{(\ell)})_{\mathrm{hop}_i^{2^{\ell-1}}}{}^\top$. The p-th block of the vector should be the one-hot vector of the following positions ($p \in [N]$)

$$e_{(\mathrm{hop}_p^{2^{\ell-2}})^{-1}\,\mathrm{hop}_i^{2^{\ell-1}}(j)} = e_{(\sigma_{p+2^{\ell-2}-1}\pi_{p+2^{\ell-2}-1}\cdots\sigma_p\pi_p)^{-1}\sigma_{i+2^{\ell-1}-1}\pi_{i+2^{\ell-1}-1}\cdots\sigma_i\pi_i(j)}.$$

Therefore, the population gradient become

$$\begin{aligned}
\nabla\mathcal{L}^{(\ell)} &= -\frac{1}{kN}\,\mathbb{E}\left[X^{(\ell-1)}\left(I_{kN\times kN} - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top\right)(X^{(\ell-1)})^\top(\Psi_\ell^\top W_{OV}^{(\ell)})_{\mathrm{hop}_i^{2^{\ell-1}}}{}^\top X_{(i,j)}^\top\right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[X^{(\ell-1)}\left((X^{(\ell-1)})^\top(\Psi_\ell^\top W_{OV}^{(\ell)})_{\mathrm{hop}_i^{2^{\ell-1}}}{}^\top - \frac{1}{N}\mathbf{1}\right)X_{(i,j)}^\top\right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[\left(\sum_{p=1}^{k} X^{(\ell-1)}\left(e_{k,p}\otimes e_{(\mathrm{hop}_p^{2^{\ell-2}})^{-1}\,\mathrm{hop}_i^{2^{\ell-1}}(j)}\right) - \frac{1}{N}\begin{bmatrix}\mathbf{1}_{kN(\ell+1)}\\ 0_{kN(L-\ell+1)}\end{bmatrix}\right)X_{(i,j)}^\top\right]
\end{aligned}$$

Consider each vector $X^{(\ell)}(e_{k,p}\otimes e_{(\mathrm{hop}_p^{2^{\ell-2}})^{-1}\,\mathrm{hop}_i^{2^{\ell-1}}(j)})$ in the first summation. Observe that when $p > 2^{\ell-2}+i$, there is a permutation $\sigma_{i+2^{\ell-2}}^{-1}$ that is independent of $X_{(i,j)}^{(\ell-1)}$, since it never appear in any existing hop encoded in $X_{(i,j)}^{(\ell-1)}$. Similarly, when $p < 2^{\ell-2}+i$, $\sigma_{i+2^{\ell-1}-1}$ is independent of $X_{(i,j)}^{(\ell-1)}$. Then the vector has the following expectation condition on $X_{(i,j)}^{(\ell-1)}$:

$$\frac{1}{N}\left[e_{k,p}\otimes\mathbf{1}_N, e_{k,p}\otimes\mathbf{1}_N, e_{k,p}\otimes\mathbf{1}_N, e_{k,p+1}\otimes\mathbf{1}_N, \cdots, e_{k,p+2^{\ell-2}-1}\otimes\mathbf{1}_N, 0_{kN(L-\ell+1)}\right]^\top$$

and they cancel with the normalization term.

Therefore, the remaining index is $p^\star = 2^{\ell-2}+i$. We denote the normalization vector for p^*:

$$\phi_i = \frac{1}{N}\left[e_{k,p^*}\otimes\mathbf{1}_N, e_{k,p^*}\otimes\mathbf{1}_N, e_{k,p^*}\otimes\mathbf{1}_N, e_{k,p^*+1}\otimes\mathbf{1}_N, \cdots, e_{k,p^*+2^{\ell-2}-1}\otimes\mathbf{1}_N, 0_{kN(L-\ell+1)}\right]^\top$$

And the ideal population gradient becomes

$$\begin{aligned}
\nabla\mathcal{L}_{\mathcal{D}}^{(\ell)} &= -\frac{\beta_0}{kN}\,\mathbb{E}\left[\left(X^{(\ell-1)}(e_{k,2^{\ell-2}+i}\otimes e_{(\mathrm{hop}_{2^{\ell-2}+i}^{2^{\ell-2}})^{-1}\,\mathrm{hop}_i^{2^{\ell-1}}(j)}) - \frac{1}{N}\phi_i\right)X_{(i,j)}^{(\ell)}\right] \\
&= -\frac{\beta_0}{kN}\,\mathbb{E}\left[\left(X^{(\ell-1)}(e_{k,2^{\ell-2}+i}\otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(j)}) - \frac{1}{N}\phi_i\right)X_{(i,j)}^{(\ell)}\right]
\end{aligned}$$

The last identity is due to the definition of the hop:

$$e_{(\mathrm{hop}_{p^\star}^{2^{\ell-2}})^{-1}\,\mathrm{hop}_i^{2^{\ell-1}}(j)} = e_{(\sigma_{p^\star+2^{\ell-2}-1}\pi_{p^\star+2^{\ell-2}-1}\cdots\sigma_{p^\star}\pi_{p^\star})^{-1}\sigma_{i+2^{\ell-1}-1}\pi_{i+2^{\ell-1}-1}\cdots\sigma_i\pi_i(j)} = \mathrm{hop}_i^{2^{\ell-2}}(j).$$

We expand the two vectors $X^{(\ell-1)}(e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(j)})$ and $X^{(\ell-1)}_{(i,j)}$:

$$X^{(\ell-1)}(e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(j)}) = \begin{bmatrix} \underline{e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(j)}} \\ e_{k,2^{\ell-2}+i} \otimes e_{\sigma_{2^{\ell-2}+i}\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}^1_{2^{\ell-2}+i}\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ e_{k,2^{\ell-2}+i+1} \otimes e_{\mathrm{hop}^2_{2^{\ell-2}+i}\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ \vdots \\ e_{k,2^{\ell-1}+i-1} \otimes e_{\mathrm{hop}^{2^{\ell-2}}_{2^{\ell-2}+i}\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ 0_{(L-\ell+1)kN} \end{bmatrix}, x_v = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}^1_i(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}^2_i(j)} \\ \vdots \\ e_{k,2^{\ell-2}+i-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ 0_{(L-\ell+1)kN} \end{bmatrix}$$

Observe that $X^{(\ell-1)}_{(i,j)}$ can be decomposed into

$$X^{(\ell-1)}_{(i,j)} = \underbrace{\begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}^1_i(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}^2_i(j)} \\ \vdots \\ e_{k,2^{l-3}+i-1} \otimes e_{N,\mathrm{hop}_i^{2^{l-3}}(j)} \\ 0_{(L-\ell+2)kN} \end{bmatrix}}_{x_1} + \underbrace{\begin{bmatrix} 0_{kN\ell} \\ e_{k,2^{\ell-2}+i-1} \otimes e_{N,\mathrm{hop}_i^{2^{\ell-2}}(j)} \\ 0_{(L-\ell+1)kN} \end{bmatrix}}_{x_2}$$

Given the first vector and $\ell \geq 3$, the whole vector $X^{(\ell-1)}(e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(j)})$ has expectation

$$\mathbb{E}\left[X^{(\ell-1)}(e_{k,2^{\ell-2}+i} \otimes e_{\mathrm{hop}_i^{2^{\ell-2}}(v)})\middle| x_{v,1}\right] = \frac{1}{N}\begin{bmatrix} \phi_i \\ 0_{kN(L-\ell+1)} \end{bmatrix}$$

because $\sigma_{2^{\ell-2}+i-2}$ is independent of x_1, and thus cancel with the normalization term.

Similarly, given the second vector x_2, the only block vector that does not have uniform expectation conditioned on x_2 is the first block (underlined). After taking expectation, the rest block vectors cancels with the corresponding blocks of the normalization vector ϕ_i. Therefore, the population gradient can be written into the following form (only the $(1, \ell+1)$-th block is non-zero):

$$\nabla \mathcal{L}^{(\ell)}_{\mathcal{D}} = -\frac{\beta_0}{kN}\,\mathbb{E}\begin{bmatrix} 0_{kN\times kN\ell} & A & 0_{kN\times(L-\ell+1)kN} \\ & 0_{kN(L+1)\times kN(L+2)} & \end{bmatrix}$$

and the expectation of A is

$$\frac{1}{kN}\sum_{i=1}^{k}\sum_{p=1}^{N}(e_{k,2^{\ell-2}+i}\otimes e_{N,p})(e_{k,2^{\ell-2}+i-1}\otimes e_{N,p})^\top - \frac{1}{kN^2}\sum_{i=1}^{k}(e_{k,2^{\ell-2}+i}\otimes \mathbf{1}_N)(e_{k,2^{\ell-2}+i-1}\otimes \mathbf{1}_N)^\top$$

We will later show that this gradient can correctly learn the functionality of the ℓth layer, combined with the following analysis upper bounding the gradient noise and accumulated error.

Following similar strategy from the second stage, we consider the empirical estimate of the gradient. Define the population gradient matrix

$$g_\ell = \mathbb{E}\begin{bmatrix} 0_{kN\times kN\ell} & A & 0_{kN\times(L-\ell+1)kN} \\ & 0_{kN(L+1)\times kN(L+2)} & \end{bmatrix}$$

The empirical gradient can be written as $\nabla \hat{L}^{(\ell)} = -\frac{\beta_0}{kN}\hat{g}_\ell$, where $\hat{g}_\ell$ is the empirical estimate of g_ℓ with perturbed inputs:

$$\hat{g}_\ell = \frac{kN}{\beta_0 M}\sum_{m=1}^{M}\sum_{s'\in[N]}(\mathbf{1}\{s'=\mathrm{hop}_{i_m}^{2^{\ell-1}}(j_m)\}-\mathcal{S}_{s'})\hat{X}_m^{(\ell-1)}J^{(\ell)}(\hat{X}_m^{(\ell-1)})^\top\left(\Psi_\ell^\top W_{OV}^{(\ell)}\right)_{s'}^\top(\hat{X}_{(i_m,j_m)}^{(\ell-1)})^\top$$

It suffices to show that $\|\hat{g}_\ell - g_\ell\|_\infty$ is small. It is similar to the first stage analysis for the sample noise part, with the perturbed input error introduced:

Lemma 25 *Suppose $\|X_m^{(\ell-1)} - \hat{X}_m^{(\ell-1)}\|_\infty \le (\ell-1)\epsilon$ for all m. For any $\delta > 0$, we have with probability $1-\delta$ s.t.*

$$\|\hat{g}_\ell - g_\ell\|_\infty \lesssim \frac{\log(kN/\delta)}{\sqrt{M}} + d\ell\epsilon$$

Proof For simplicity, we ignore the superscript $^{(\ell-1)}$ in the proof of this lemma. The proof in large follows Lemma 20.

First, the empirical gradient of the single ideal sample X_m is

$$\hat{g}_{\ell,m}^{\mathrm{ideal}} = X_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes e_{N,\mathrm{hop}^{2^{\ell-1}}(v_m)})(X_{(i_m,j_m)})^\top$$

Similar to the proof in Lemma 20, the upper bound of each entry of the random variable is

$$\|\hat{g}_{\ell,m}^{\mathrm{ideal}}\|_\infty \le \left\|X_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes e_{N,\mathrm{hop}_{i_m}^{2^{\ell-1}}(j_m)})\right\|_\infty \|X_{(i_m,j_m)}^{(\ell-1)}\|_\infty \le 1. \tag{8}$$

Then we can concentrate $\hat{g}_\ell^{\mathrm{ideal}} := \frac{1}{M}\sum_{m=1}^M \hat{g}_{2,m}^{\mathrm{ideal}}$ as:

$$\left\|\hat{g}_\ell^{\mathrm{ideal}} - g_\ell\right\|_\infty = \left\|\frac{1}{M}\sum_{m=1}^{M}\hat{g}_{l,m}^{\mathrm{ideal}} - \mathbb{E}\,\hat{g}_l^{\mathrm{ideal}}\right\| \lesssim \frac{\log(d/\delta)}{\sqrt{M}}$$

with probability $1-\frac{\delta}{d^2}$. By $d = O(kN\log k)$, we can union bound over all entries of $\hat{g}_2$ and we have the desired first term error.

Then we bound the perturbation error between the empirical gradients with X_m and those with perturbed $\hat{X}_m$. Recall that the empirical gradient

$$\begin{aligned}\hat{g}_{\ell,m} = {} & \hat{X}_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes e_{N,\mathrm{hop}_{i_m}^{2^{\ell-1}}(j_m)})(\hat{X}_{(i_m,j_m)}^{(\ell-1)})^\top \\ & - \sum_{s'\in[N]}\mathcal{S}_{s'}\hat{X}_m(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,\ell+1}\otimes\mathbf{1}_k\otimes e_{N,s'})(\hat{X}_{(i_m,j_m)}^{(\ell-1)})^\top.\end{aligned}$$

Denote the following term as

$$\hat{\gamma}_{s',m;\ell} = \hat{X}_m(I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top)\hat{X}_m^\top(e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'})(\hat{X}^{(\ell-1)}_{(i_m,j_m)})^\top,$$

$$\gamma_{s',m;\ell} = X_m(I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top)X_m^\top(e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'})(X^{(\ell-1)}_{(i_m,j_m)})^\top.$$

and we define $\Delta\gamma_{s',m;\ell} = \hat{\gamma}_{s',m;\ell} - \gamma_{s',m;\ell}$. Then we can rewrite the empirical gradient into

$$\hat{g}_{\ell,m} - \hat{g}^{\text{ideal}}_{\ell,m} = \Delta\gamma_{\text{hop}^{2^{\ell-1}}(v_m),m} - \sum_{s'\in[N]} \mathcal{S}_{s'}\Delta\gamma_{s',m}.$$

The error of the following difference $\|\Delta\gamma_{s',m;\ell}\|_\infty = \|\gamma_{s',m;\ell} - \hat{\gamma}_{s',m;\ell}\|_\infty \le Cd\ell\epsilon$ with some absolute constant for all possible $s' \in [N]$, since $\left\|\hat{X}_m - X_m\right\|_\infty \le (\ell-1)\epsilon$ and $\|X_m\|_\infty \le 1$. We have the perturbation error upper bounded by $O(d\ell\epsilon)$:

$$\left\|\hat{g}_{\ell,m} - \hat{g}^{\text{ideal}}_{\ell,m}\right\|_\infty \le \|\Delta\gamma_{\text{hop},m;\ell}\| + \sum_{s'} \mathcal{S}_{s'}\|\Delta\gamma_{s',m;\ell}\| \le 2Cd\ell\epsilon. \tag{9}$$

Combine both (8) and (9), we finished the proof. ■

After the one-step gradient, we get $W^{(\ell)}_{KQ} = \eta\hat{g}_\ell$. Given a sample sequence $\hat{X}^{(\ell)}_m$ and index (i_m, j_m), we have the attention score

$$\eta(\hat{X}^{(\ell-1)}_m)^\top\hat{g}_\ell\hat{X}^{(\ell-1)}_{(i_m,j_m)} = \frac{\beta_0\eta}{kN}\left[(X^{(\ell-1)}_m)^\top g_l X^{(\ell-1)}_{(i_m,j_m)} + (X^{(\ell-1)}_m)^\top \underbrace{(\hat{g}_\ell - g_\ell)}_{\Delta g_\ell}\hat{X}^{(\ell-1)}_{(i_m,j_m)}\right]$$
$$+ \frac{\beta_0\eta}{kN}\underbrace{\left[(\hat{X}^{(\ell-1)}_m)^\top\hat{g}_\ell\hat{X}^{(\ell-1)}_{(i_m,j_m)} - X_m^\top\hat{g}_\ell X_{(i_m,j_m)}\right]}_{\text{Perturbation error}}$$

For simplicity, we ignore the superscript for the following proof in this subsection. Note that the population attention score is

$$X_m^\top g_\ell X_{(i_m,j_m)} = X_m^\top\begin{bmatrix}\frac{1}{kN}\left(e_{k,2^{\ell-2}+i_m} \otimes e_{N,\text{hop}^{2^{\ell-2}}_{i_m}(j_m)} - \frac{1}{N}e_{k,2^{\ell-2}+i_m} \otimes \mathbf{1}_N\right)\\ 0_{kN(L+1)}\end{bmatrix}$$

Define the population/empirical separation between the correct position $p_\ell := N(i_m + 2^{\ell-2}) + \text{hop}^{2^{\ell-2}}_{i_m}(j_m)$, i.e. $(i_m + 2^{\ell-2}, \text{hop}^{2^{\ell-2}}_{i_m}(j_m))$ position, and the others as follows:

$$\Delta^{(\ell)}_{i_m,j_m} = \left(X_m^\top g_l X_{(i_m,j_m)}\right)_{p_\ell} - \max_{j\neq p_\ell}\left(X_m^\top g_l X_{(i_m,j_m)}\right)_j.$$

$$\hat{\Delta}^{(\ell),\text{ideal}}_{i_m,j_m} = \left(X_m^\top\hat{g}_\ell X_{(i_m,j_m)}\right)_{p_\ell} - \max_{j\neq p_\ell}\left(X_m^\top\hat{g}_\ell X_{(i_m,j_m)}\right)_j.$$

$$\hat{\Delta}^{(\ell)}_{i_m,j_m} = \left(\hat{X}_m^\top\hat{g}_\ell\hat{X}_{(i_m,j_m)}\right)_{p_\ell} - \max_{j\neq p_\ell}\left(\hat{X}_m^\top\hat{g}_\ell\hat{X}_{(i_m,j_m)}\right)_j.$$

We have the (i,j)-th entry of the pre-softmax attention score

$$\left(X_m^\top g_\ell X_{(i_m,j_m)}\right)_{(i,j)} = \begin{cases} \frac{N-1}{kN^2}, & i = i_m + 2^{\ell-2}, j = \mathrm{hop}_{i_m}^{2^{\ell-2}}(j_m) \\ -\frac{1}{kN^2}, & i = i_m + 2^{\ell-2},\ j \neq \mathrm{hop}_{i_m}^{2^{\ell-2}}(j_m) \\ 0, & i \neq i_m + 2^{\ell-2} \end{cases}$$

So $\Delta_{i_m,j_m}^{(\ell)} \geq \frac{1}{kN^2}$ by $N \geq 2$ and the expected signal is at least $\frac{1}{kN^2}$.

Now we bound the noise introduced by Δg_ℓ. Since $\|\Delta g_\ell\|_\infty \lesssim \frac{\log \frac{d}{\delta}}{\sqrt{M}} + d\ell\epsilon$, we have

$$\left\|X_m^\top \Delta g_\ell X_{(i_m,j_m)}\right\|_\infty \lesssim \frac{d \log \frac{d}{\delta}}{\sqrt{M}} + d\ell\epsilon \leq \frac{1}{2kN^2}$$

by $M \gtrsim k^2N^4d^2\log^2\frac{d}{\delta}$ and $\epsilon \lesssim \frac{1}{k^2N^3\log^2 k}$. Therefore, the noise term can be bounded and the empirical separation with the ideal input sequence $\hat{\Delta}_{i_m,j_m}^{(\ell),\mathrm{ideal}} \geq \frac{1}{2kN^2}$. This also gives the upper bound for $\|\hat{g}_\ell\|_\infty \leq O(1/kN^2)$.

Similar to stage 2, we check the upper bounds for gradients for previous layers, and make sure the perturbation error upper bounds for $\left\|\hat{X}_m^{(\ell)} - X_m^{(\ell)}\right\| \leq \ell\epsilon$ still hold. The following lemma shows that after the ℓth gradient step, the previous layer key-query matrices are almost unperturbed and the softmax score is still close to one-hot.

Lemma 26 ($W_{KQ}^{(\ell')}$ is unchanged) *Under the condition of Lemma 24, after the ℓth step we still have for all $\ell' < \ell$,*

$$\mathcal{S}_{(i,\pi_i(j))}^{(\ell')} := \mathcal{S}(\hat{X}^{(\ell'-1)}W_{KQ}^{(\ell')}\hat{X}_{(i,j)}^{(\ell'-1)})_{(i,\mathrm{hop}_i^{2^{\ell'-2}}(j))} \geq 1 - \frac{1}{2}\epsilon.$$

Proof By Lemma 29, the gradient norm for each previous layer is upper bounded by $O(\beta_0 k^{5/2}N^{3/2}L^{3/2}\epsilon)$. With the gradient update in the current stage, $W_{KQ}^{(\ell')}$ will only be perturbed by

$$\left\|\eta\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\| \lesssim (kN)^{4.5}\log\frac{kN}{\epsilon}\log^{2.5}k\cdot\epsilon \ll N\log\frac{kN}{\epsilon}.$$

where $N\log\frac{kN}{\epsilon}$ is the scale of each entry of $W_{KQ}^{(\ell')}$. Therefore, the lemmas on the attention score in previous stages still hold. ■

Given the lemma above, we have $\left\|\hat{X}_m^{(\ell)} - X_m^{(\ell)}\right\| \leq \ell\epsilon$ using the previous layer perturbation error analysis for $X_m^{(\ell)}$. Finally, we are ready to bound the perturbation error for pre-softmax attention score:

$$\begin{aligned}
&\left\|(\hat{X}_m^{(\ell)})^\top \hat{g}_\ell \hat{X}_{(i_m,j_m)}^{(\ell)} - X_m^\top \hat{g}_\ell X_{(i_m,j_m)}\right\|_\infty \\
\leq\ &\|\hat{g}_\ell\|_2\left\|(\hat{X}_m^{(\ell)})^\top(\hat{X}_{(i_m,j_m)}^{(\ell)} - X_{(i_m,j_m)})\right\|_\infty + \|\hat{g}_\ell\|_2\left\|(\hat{X}_m^{(\ell)} - X_m)^\top X_{(i_m,j_m)}\right\|_\infty \\
\leq\ &\|\hat{g}_\ell\|_F\left\|(\hat{X}_m^{(\ell)})^\top(\hat{X}_{(i_m,j_m)}^{(\ell)} - X_{(i_m,j_m)})\right\|_\infty + \|\hat{g}_\ell\|_F\left\|(\hat{X}_m^{(\ell)} - X_m)^\top X_{(i_m,j_m)}\right\|_\infty
\end{aligned}$$

$$\lesssim \sqrt{\frac{1}{(kN^2)^2} \cdot d^2} \cdot d\ell\epsilon.$$

Since $\epsilon = O(\frac{1}{k^2N^3L^2})$, the perturbation error is upper bounded by $O(\frac{1}{kN^2})$. Therefore, the empirical separation $\hat{\Delta}^{(\ell)}_{(i_m,j_m)} \geq \Omega(\frac{1}{kN^2})$. After one step gradient with learning rate $\eta \gtrsim \frac{k^2N^3}{\beta_0} \log \frac{kN}{\epsilon}$, the softmax output of the correct position can be lower bounded by

$$\mathcal{S}(\hat{X}_m^\top W_{KQ}^{(\ell)} \hat{X}_{(i_m,j_m)})_{i_m,\mathrm{hop}_{i_m}^{\ell-2}(j_m)} \geq \frac{\exp\left(\frac{\eta\beta_0}{kN} \cdot \frac{1}{kN^2}\right)}{\exp\left(\frac{\eta\beta_0}{kN} \cdot \frac{1}{kN^2}\right) + kN - 1} \geq 1 - \frac{1}{2}\epsilon.$$

■

Perturbation analysis of Stage ℓ The lemma presents the perturbation analysis for stage ℓ.

Lemma 27 (Perturbation analysis of stage ℓ) *Under the condition of Lemma 24, we have*

$$\|X^{(\ell)} - \hat{X}^{(\ell)}\| \leq \ell\epsilon,$$

where $X^{(\ell)}$ is the ideal output with saturated softmax, and $\hat{X}^{(\ell)}$ is the transformer output after the ℓth stage.

Proof The proof is similar to stage 2. The ideal output for the ℓth layer is

$$X^{(\ell)} = X^{(\ell-1)} + W_{OV}^{(\ell)} X^{(\ell-1)} \mathcal{S}_{\text{ideal}}^{(\ell)},$$

where $\mathcal{S}_{\text{ideal}}^{(\ell)}$ is the ideal one-hot softmax attention pattern. The empirical output $\hat{X}^{(\ell)}$ has the error introduced by the non-saturation of the softmax and the previous error in $\hat{X}^{(\ell-1)}$:

$$\begin{aligned}\hat{X}^{(\ell)} &= \hat{X}^{(\ell-1)} + W_{OV}^{(\ell)} \hat{X}^{(\ell-1)} \mathcal{S}^{(\ell)} \\ &= X^{(\ell)} + W_{OV}^{(\ell)} X^{(\ell-1)} \underbrace{(\mathcal{S}^{(\ell)} - \mathcal{S}_{\text{ideal}}^{(\ell)})}_{\Delta\mathcal{S}^{(\ell)}\text{,Non-saturation error}} + \underbrace{(\hat{X}^{(\ell-1)} - X^{(\ell)}) + W_{OV}^{(\ell)}(\hat{X}^{(\ell-1)} - X^{(\ell-1)})\mathcal{S}^{(\ell)}}_{\text{Accumulated perturbation error}}.\end{aligned}$$

First, we consider the non-saturation error term. By Lemma 19, the correct entry of the each softmax probability vector is greater than $1 - \frac{1}{2}\epsilon$ for all index v. And other probabilities has ϵ error in total, and they are all positive. Note that $\|X^{(\ell-1)}\|_\infty \leq 1$. As a result, the error of the non-saturation error can be bounded as

$$\|W_{OV}^{(\ell)} X^{(\ell-1)} (\mathcal{S}^{(\ell)} - \mathcal{S}_{\text{ideal}}^{(\ell)})\|_\infty = \max_{s,i,j} \left| (W_{OV}^{(\ell)} X^{(\ell-1)})_s^\top \Delta\mathcal{S}_{i,j}^{(\ell)} \right| \leq \|X^{(\ell-1)}\|_\infty \cdot 2 \cdot \frac{\epsilon}{2} \leq \epsilon.$$

Now consider the accumulated perturbation error. By the perturbation analysis in stage $\ell - 1$, we have $\left\|\hat{X}^{(\ell-1)} - X^{(\ell-1)}\right\|_\infty \leq (\ell-1)\epsilon$. Note that the error in $\hat{X}^{(\ell-1)} - X^{(\ell-1)}$ are in different rows of the matrices from $W_{OV}^{(\ell)}(\hat{X}^{(\ell-1)} - X^{(\ell-1)})\mathcal{S}^{(\ell)}$. Therefore, $\hat{X}^{(\ell-1)} - X^{(\ell-1)}$ won't introduce extra error in this stage, and we only need to consider $W_{OV}^{(\ell)}(\hat{X}^{(\ell-1)} - X^{(\ell-1)})\mathcal{S}^{(\ell)}$:

$$\left\| W_{OV}^{(\ell)} (\hat{X}^{(\ell-1)} - X^{(\ell-1)}) \mathcal{S}^{(\ell)} \right\|_\infty = \max_{s,(i,j)} \left| (W_{OV}^{(\ell)} (\hat{X}^{(\ell-1)} - X^{(\ell-1)}))_s^\top \mathcal{S}_{(i,j)}^{(\ell)} \right|$$

$$= \max_{s,(i,j)} \left| \sum_{p=1}^{kN} (W_{OV}^{(\ell)} (\hat{X}^{(\ell-1)} - X^{(\ell-1)}))_{s,p} (\mathcal{S}_{(i,j)}^{(\ell)})_p \right|$$
$$\leq \|\hat{X}^{(\ell-1)} - X^{(\ell-1)}\|_\infty$$
(Since $\sum_p (\mathcal{S}_{(i,j)}^{(2)})_p = 1, (\mathcal{S}_{(i,j)}^{(2)})_p \geq 0$.)

Combine both parts of error, we have $\left\|\hat{X}^{(\ell)} - X^{(\ell)}\right\|_\infty \leq \ell\epsilon$. ■

At the end of each stage ℓ, we further train the readout layer with one gradient step to output the correct $2^{\ell-1}$-hop for each position.

Lemma 28 *Under the condition of Lemma 24, after one gradient step on Ψ_ℓ we have*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X^{(\ell-1)})_{(i,j)}) - e_{\text{hop}_i^{2^{\ell-1}}(j)} \right\|_\infty \leq \epsilon.$$

Proof We calculate the population gradient for Ψ_ℓ, and then do the finite sample analysis. The population gradient is

$$\nabla_{\Psi_\ell} \mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) = -\mathbb{E}\left[\left(e_{N,\text{hop}_i^{2^{\ell-1}}(j)} - \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)}) \right) (f^{(\ell)}(X)_{(i,j)})^\top \right]$$

By Lemma 24, the output of the ℓth layer would be

$$f^{(\ell)}(X)_{(i,j)} = W_{OV}^{(\ell)} \hat{X}^{(\ell-1)} \mathcal{S}^{(\ell)} = W_{OV}^{(\ell)} X^{(\ell-1)} \mathcal{S}_{\text{ideal}}^{(\ell)} + W_{OV}^{(\ell)} (\hat{X}^{(\ell-1)} - X^{(\ell-1)}) \mathcal{S}_{\text{ideal}}^{(\ell)} + W_{OV}^{(\ell)} X^{(\ell-1)} \Delta\mathcal{S}^{(\ell)}$$
$$= e_{L+2,\ell+2} \otimes e_{k,i} \otimes e_{N,\text{hop}_i^{2^{\ell-1}}(j)} + \Delta_\ell.$$

where $\|\Delta_\ell\|_\infty \leq \ell\epsilon$ by Lemma 27. Since $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2} \otimes \mathbf{1}_k \otimes I_{N\times N}$, we have the expansion

$$\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)}) = \mathcal{S}(\beta_0 e_{N,\text{hop}_i^{2^{\ell-1}}(j)} + \Psi_\ell^\top \Delta_\ell) = \mathcal{S}(\beta_0 e_{N,\text{hop}_i^{2^{\ell-1}}(j)}) + \tilde{J}\Psi_\ell^\top \Delta_\ell.$$

Since $\|\Delta_\ell\|_\infty \leq \epsilon$, we have $\left\|\tilde{J}\Psi_\ell^\top \Delta_\ell\right\|_\infty \leq \beta_0\epsilon$. The signal term

$$\mathcal{S}(\beta_0 e_{N,\text{hop}_i^{2^{\ell-1}}(j)}) = \frac{\exp(\beta_0) - 1}{\exp(\beta_0) + N - 1} e_{N,\text{hop}_i^{2^{\ell-1}}(j)} + \frac{1}{\exp(\beta_0) + N - 1} \mathbf{1}_N.$$

The population gradient is thus

$$\nabla_{\Psi_\ell} \mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) = -\mathbb{E}\left[\left(\frac{N}{\exp(\beta_0) + N - 1} e_{N,\text{hop}_i^{2^{\ell-1}}(j)} - \frac{1}{\exp(\beta_0) + N - 1} \mathbf{1}_N - \tilde{J}\Psi_\ell^\top \Delta_\ell \right) (f^{(\ell)}(X)_{(i,j)})^\top \right]$$
$$= -\frac{1}{(\exp(\beta_0) + N - 1)k} \left(e_{L+2,\ell+2} \otimes \mathbf{1}_k \otimes (I_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^\top) \right) + O(\epsilon).$$

where $O(\epsilon)$ denotes the terms with infinity norm smaller than ϵ with $\epsilon \lesssim \frac{1}{k^2 N^3 L^2}$. And with similar analysis in stage 2, the error altogether is upper bounded by $O(\frac{1}{k^2 N^3})$ in infinity norm.

After one step of gradient, we have the softmax score (Δ is the error term with $\|\Delta\|_\infty \le \epsilon$.)

$$\mathcal{S}(\Psi_\ell(1)^\top f^{(\ell)}(X)_{(i,j)})) = \mathcal{S}\bigg(\frac{\eta}{(\exp(\beta_0)+N-1)k}(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \beta_0 e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \eta\Delta\bigg)$$

The separation between the correct entry and the others are lower bounded by:

$$\frac{\eta}{(\exp(\beta_0)+N-1)k}\frac{N-1}{N} - \eta\|\Delta\|_\infty \gtrsim \frac{\eta}{kN}.$$

By $\eta \gtrsim k^2N^3\log\frac{kN}{\epsilon}$, we have $\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)})_{\mathrm{hop}_i^{2^{\ell-1}}(j)} \ge 1-\epsilon$ and thus

$$\sup_{\sigma,(i,j)} \Big\|\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)}) - e_{\mathrm{hop}_i^{2^{\ell-1}}(j)}\Big\|_\infty \le \epsilon.$$

■

C.7. Gradient Upper Bound for Trained Layers

In this section, we prove the following technical lemma showing that the gradients $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ are very small with $\ell' < \ell$ in the ℓth stage due to softmax saturation. To be specific, we prove a more general version that also covers the mixed training algorithm.

Lemma 29 *Given a single sample (X,i,j) and suppose the training is in stage ℓ_0, for all $\ell' < \ell_0$,*

$$\mathcal{S}((\hat{X}^{(\ell'-1)})^\top W_{KQ}^{(\ell')}\hat{X}_{(i,j)}^{(\ell'-1)})_{(i+2^{\ell'-2},\mathrm{hop}_i^{2^{\ell'-2}}(j))} \ge 1 - \frac{1}{2}\epsilon,$$

and for $\ell'' > \ell_0$ the norm of the parameter $\|W_{KQ}^{(\ell'')}\|_2 \lesssim \frac{1}{k^2N^2L^2}$[3]. Then we have for any $\ell' < \ell_0$, the infinity norm of the ℓ_0th stage gradient is upper bounded by

$$\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\|_\infty \le 6\beta_0 k(kNL)^{3/2}\epsilon, \forall \ell > \ell'.$$

Proof Recall the gradient for a single sample (X,i,j)

$$\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)} = -\left[\sum_{s'\in[N]}\Big(\mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} - \mathcal{S}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\Big)\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right]$$

So the norm of the gradient is upper bounded by

$$\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\| \le \left\|\sum_{s'\in[N]}\Big(\mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} - \mathcal{S}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\Big)\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right\|$$

3. This condition holds for both curriculum training and mix training. In particular, the norm is exactly zero for curriculum training Algorithm 1.

$$\leq 2\left\|\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right\| = 2\left\|\nabla_{W_{KQ}^{(\ell')}}(e_{N,s'}^\top \Psi_\ell^\top f_{(i,j)}^{(\ell)})\right\|.$$

Now we calculate the gradient recursively through Taylor expansion: for all ΔW with $\|\Delta W\| \to 0$,

$$f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')} + \Delta W) = f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')}) + \langle \nabla_{W_{KQ}^{(\ell')}}(f_{(i,j)}^{(\ell)}), \Delta W\rangle + O(\|\Delta W\|_F^2).$$

While by Taylor expansion, we have (we ignore $(W_{KQ}^{(\ell')})$ when there is no perturbation ΔW.)

$$\begin{aligned} f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')} + \Delta W) = f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')}) &+ W_{OV}^{(\ell)} \nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W)\mathcal{S}((X^{(\ell-1)})^\top W_{KQ}^{(\ell)} X_{(i,j)}^{(\ell-1)}) \\ &+ W_{OV}^{(\ell)} X^{(\ell-1)} J^{(\ell)} \nabla((X^{(\ell-1)})^\top W_{KQ}^{(\ell)} X_{(i,j)}^{(\ell-1)})(\Delta W) + O(\|\Delta W\|_F^2). \end{aligned}$$

Here the gradient $\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)\in\mathbb{R}^{d\times kN\times d\times d}}$ is a 4-th order tensor, and $\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W) \in \mathbb{R}^{d\times kN}$. The second term can be expanded into

$$W_{OV}^{(\ell)} X^{(\ell-1)} J^{(\ell)} \nabla((X^{(\ell-1)})(\Delta W)^\top) W_{KQ}^{(\ell)} X_{(i,j)}^{(\ell-1)} + W_{OV}^{(\ell)} X^{(\ell-1)} J^{(\ell)} (X^{(\ell-1)})^\top W_{KQ}^{(\ell)} \nabla(X_{(i,j)}^{(\ell-1)})(\Delta W).$$

In this way, we reduce the problem to calculating the norm upper bound for $\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W)$.
We upper bound the infinity norm of the matrix by induction from layer ℓ' to ℓ. We prove that

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\|_\infty \leq (kNL)^{3/2}\epsilon(1+\frac{1}{k})^{t-\ell'}\|\Delta W\|_F$$

with $t \in [\ell', \ell-1]$.

Base case: $t = \ell'$ By Taylor expansion on $X^{(\ell')}$, we have

$$X^{(\ell')}(W_{KQ}^{(\ell')} + \Delta W) = X^{(\ell')} + W_{OV}^{(\ell')} X^{(\ell'-1)} J^{(\ell')} (X^{(\ell'-1)})^\top \Delta W X^{(\ell'-1)}) + O(\|\Delta W\|_F^2).$$

since previous layers are independent of $W_{KQ}^{(\ell')}$. Therefore, the first order term is

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\| = \left\|W_{OV}^{(\ell')} X^{(\ell'-1)} J^{(\ell')} (X^{(\ell'-1)})^\top \Delta W X^{(\ell'-1)})\right\|$$

Note that the softmax is close to one-hot,

$$\mathcal{S}((\hat{X}^{(\ell'-1)})^\top W_{KQ}^{(\ell')} \hat{X}_{(i,j)}^{(\ell'-1)})_{(i+2^{\ell'-2}, \mathrm{hop}_i^{2^{\ell'-2}}(j))} \geq 1 - \frac{1}{2}\epsilon,$$

we have the Jacobian $\left\|J^{(\ell)}\right\| \lesssim \epsilon$. Therefore, we can upper bound the first order term by

$$\begin{aligned} \|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_2 &\leq \left\|W_{OV}^{(\ell')}\right\| \left\|X^{(\ell'-1)}\right\|_F \left\|J^{(\ell')}\right\|_2 \left\|(X^{(\ell'-1)})^\top\right\|_F \|\Delta W\| \left\|X^{(\ell'-1)})\right\|_F \\ &\lesssim (kNL)^{3/2}\epsilon\|\Delta W\|_F. \end{aligned}$$

($\left\|X^{(\ell'-1)}\right\|_F^2 \leq O(kNL)$ since each embedding is either ϵ-close to one-hot or all 0.)

Since $\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_\infty \le \|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_2$, we finish the proof for base case.

Induction: $t \in [\ell', \ell - 1]$**.** Suppose the induction hypothesis holds:

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\|_\infty \le (knL)^{3/2}\epsilon(1+\frac{1}{k})^{t-\ell'}\|\Delta W\|_F.$$

We consider expanding $X^{(t+1)}(W_{KQ}^{(\ell')} + \Delta W)$ like the base case:

$$\begin{aligned}
X^{(t+1)}(W_{KQ}^{(\ell')} + \Delta W) &= X^{(t)}(W_{KQ}^{(\ell')} + \Delta W) + f^{(t+1)}(W_{KQ}^{(\ell')} + \Delta W) \\
&= X^{(t+1)} + \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) && \text{(From } X^{(t)}(W_{KQ}^{(\ell')} + \Delta W).\text{)} \\
&\quad + W_{OV}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \mathcal{S}\left(\left(X^{(t)}\right)^\top W_{KQ}^{(t+1)} X^{(t)}\right) \\
&\quad + W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \left(\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\right)^\top W_{KQ}^{(t+1)} X^{(t)} \\
&\quad + W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \left(X^{(t)}\right)^\top W_{KQ}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) + O(\|\Delta W\|_F^2) \\
&&& \text{(The last 3 terms are from } f^{(t+1)}(W_{KQ}^{(\ell')} + \Delta W).\text{)}
\end{aligned}$$

Note that $\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)$ and the rest three terms are in different rows since they uses different $W_{OV}^{(t)}$, so the infinity norm of the first order term should be the maximum of those two.

We first upper bound the last three terms. Since $W_{OV}^{(t+1)}$ is partial identity, and softmax is some weighted average, we have

$$\left\| W_{OV}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \mathcal{S}\left(\left(X^{(t)}\right)^\top W_{KQ}^{(t+1)} X^{(t)}\right)\right\| \le \left\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\right\|_\infty.$$

The rest two terms can be directly upper bounded by

$$\left\|W_{OV}^{(t+1)}\right\|_2 \left\|X^{(t)}\right\|_2 \left\|J^{(t+1)}\right\|_2 \left\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W))\right\|_2 \left\|W_{KQ}^{(t+1)}\right\|_2 \left\|X^{(t)}\right\|_2.$$

In previous stages, we either have (1) tth layer is trained: $\|X\|_F^2 \le knL$, $\|J^{(t+1)}\|_2 \le \epsilon$ and $\|W_{KQ}^{(t+1)}\|_2 \lesssim N \log \frac{kN}{\epsilon}$, or (2) tth layer is close to initialization: $\|J^{(t+1)}\|_2 \le 1$ and $\|W_{KQ}^{(t+1)}\|_2 \lesssim \frac{1}{knL}$. So these two terms can be both upper bounded by

$$kN^2 L\epsilon \log\frac{kN}{\epsilon} \left\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W))\right\|_2 \le \frac{1}{k}\left\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W))\right\|_\infty$$

since $\epsilon \log \frac{1}{\epsilon} \le k^6 N^6 L^6$. Combining two error terms, we have

$$\left\|\nabla_{W_{KQ}^{(\ell')}} X^{(t+1)}(\Delta W)\right\|_\infty \le (knL)^{3/2}\epsilon(1+\frac{1}{k})^{t-\ell'+1}\|\Delta W\|_F.$$

By induction, when $t = \ell - 1$ we have (since $\ell - \ell' < k$.)

$$\left\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W)\right\|_\infty \le (kNL)^{3/2}\epsilon(1+\frac{1}{k})^{\ell-\ell'}\|\Delta W\|_F \le 3(kNL)^{3/2}\epsilon\|\Delta W\|_F.$$

Finally, we can upper bound $\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\|$ by picking $\Delta W = \alpha\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ with $\alpha \to 0$:

$$\begin{aligned}
\alpha\left\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\|_F^2 &\le 2\left\langle \nabla_{W_{KQ}^{(\ell')}}(e_{N,s'}^\top \Psi_\ell^\top f_{(i,j)}^{(\ell)}), \Delta W\right\rangle \\
&\le 2e_{N,s'}^\top \Psi_\ell^\top W_{OV}^{(\ell)} \nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W)\mathcal{S}((X^{(\ell-1)})^\top W_{KQ}^{(\ell)} X_{(i,j)}^{(\ell-1)}) + O(\alpha^2) \\
&\le 2\beta_0 k\left\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W)\right\|_\infty \le 6\beta_0 k(kNL)^{3/2}\epsilon\|\Delta W\|_F.
\end{aligned}$$

Plug in $\Delta W = \alpha\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$, we have $\left\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\|_F \le 6\beta_0 k(kNL)^{3/2}\epsilon$. ■

Appendix D. Training on a Mixture of Different Hops

Recall that the parameter vector is $\theta := (\theta_{KQ}, \theta_\Psi)$, where $\theta_{KQ} = (W_{KQ}^{(1)}, \cdots, W_{KQ}^{(L)})$ and $\theta_\Psi = (\Psi_1, \cdots, \Psi_L)$, and that we consider the following mixed training loss, which is a summation of the loss on all 2^ℓ-hops:

$$\mathcal{L}^{\mathrm{M}}(\theta) := \sum_{\ell=1}^{L}\mathcal{L}^{(\ell)}(\theta) = -\frac{1}{M}\sum_{\ell=1}^{L}\sum_{m=1}^{M}\left[\sum_{s'\in[N]}\mathbf{1}\{s' = \mathrm{hop}_{i_m}^{2^{\ell-1}}(\sigma^{(m)}, j_m)\}\log\Big(\mathcal{S}(\Psi_\ell^\top \mathrm{TF}_\theta(X_m)_{(i_m,j_m)})_{s'}\Big)\right]$$

The learning algorithm proceeds by taking L gradient descent steps on the key-query matrices θ_{KQ}, followed by a single gradient descent step on the readout layer θ_Ψ. Pseudocode for the mixed training algorithm is presented in Algorithm 2.

Algorithm 2 Training Algorithm (Data Mixture)

Input: initialization size β_0; learning rates η

Initialize $W_{KQ}^{(\ell)}(0) = 0_{d\times d}$, $\Psi^{(\ell)}(0) = \beta_0 e_{L+2,\ell+2}\otimes \mathbf{1}_k \otimes I_{N\times N}$, $\ell \in [L]$

for $\ell = 1, \cdots, L$ **do**

  Use loss $\mathcal{L}^{\mathrm{M}}(\theta(\ell-1))$. ▷ Stage ℓ: train on mixture of all $2^{\ell-1}$ hops

  $\theta_{KQ}(\ell) \leftarrow \theta_{KQ}(\ell-1) - \eta\nabla_{\theta_{KQ}}\mathcal{L}^{\mathrm{M}}(\theta'(\ell-1))$ ▷ Train the key-query matrices θ_{KQ}

  $\theta'(\ell) \leftarrow (\theta_{KQ}(\ell), \theta_\Psi(0))$

end

$\theta_\Psi(L) \leftarrow \theta_\Psi(0) - \eta\nabla_{\theta_\Psi}\mathcal{L}^{\mathrm{M}}(\theta'(L))$ ▷ Train the readout layer θ_Ψ

$\theta(L) \leftarrow (\theta_{KQ}(L), \theta_\Psi(L))$

Output: $\hat{\theta} = \theta(L)$.

We restate the theorem for mixed training below. Here we require that the error is sufficiently small s.t. $0 < \epsilon \le \tilde{O}(\frac{1}{k^6N^6})$, $\epsilon\log^{2L}\frac{1}{\epsilon} \le 1$, which is needed to keep the accumulation error and gradient error small in the proof.

Theorem 8 (Guarantee for mixed data training) *Assume $k = 2^{L-1}$, $M \geq \tilde{\Omega}(k^4 N^6)$ and $\eta \geq \tilde{\Omega}(\frac{k^2 N^3}{\beta_0} \log \frac{1}{\epsilon})$. For sufficiently small $\epsilon > 0$, with high probability the final output $\hat{\theta}$ of Algorithm 2 satisfies that over any draw of input permutations σ and query index (i, j), for any $\ell \in [L]$,*

$$\sup_{\sigma,(i,j)} \left\| \mathcal{S}(\Psi_\ell^\top \mathrm{TF}_{\hat{\theta}}(X(\sigma))_{(i,j)}) - e_{\mathrm{hop}_i^{2^{\ell-1}}(j)} \right\|_\infty \leq \epsilon.$$

In particular, the transformer approximates the k-fold composition task when $\ell = L$.

D.1. Proof Outline

We provide a proof outline in this section, and defer the technical lemmas to the later subsections.

Proof [Proof of Theorem 8] We prove that the population gradient dynamics for θ_{KQ} is identical to the curriculum dynamics in Theorem 7. On the population trajectory, we first show the following key observation: **when the previous layer is not trained and stays zero, all later layers have zero gradient and also stay zero.** Therefore, the mixed training algorithm induces the same implicit curriculum training as Algorithm 1.

We first recall from Section C.2, the main signal term, i.e. the gradient for the ℓth layer $W_{KQ}^{(\ell)}$ over loss $\mathcal{L}^{(\ell)}$ is

$$\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)} = -\mathbb{E}\left[\sum_{s' \in [N]} \left(\mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} - \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X^{(\ell-1)})_{(i,j)})_{s'} \right) X J^{(\ell)} X^\top (\Psi_\ell^\top W_{OV}^{(\ell)})_{s'}{}^\top X_{(i,j)}^\top \right].$$

Suppose that we are in the tth step of gradient descent, so that the first untrained key-query matrix is the tth layer $W_{KQ}^{(t)}$. We will define $X_{t-1}^{(\ell'-1)}$ to be the "ideal" input to the ℓ'th layer after $t-1$ gradient steps ($\ell' > t$); the (i,j) column is given by

$$X_{t-1,(i,j)}^{(\ell'-1)} := \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\mathrm{hop}_i^1(j)} \\ e_{k,i+1} \otimes e_{N,\mathrm{hop}_i^2(j)} \\ \vdots \\ e_{k,i+2^{t-3}-1} \otimes e_{N,\mathrm{hop}_i^{2^{t-3}}(j)} \\ \frac{1}{kN}\mathbf{1}_{kN} \\ \vdots \\ \frac{1}{kN}\mathbf{1}_{kN} \\ 0_{(L-\ell'+1)kN} \end{bmatrix}.$$

Therefore, we have

$$J^{(\ell')}(X_{t-1}^{(\ell'-1)})^\top (\Psi_{\ell'}^\top W_{OV}^{(\ell')})_{s'}{}^\top = \beta_0 \cdot J^{(\ell')}(X_{t-1}^{(\ell'-1)})^\top e_{L+2,\ell'+1} \otimes \mathbf{1}_k \otimes e_{N,s'} = J^{(\ell')} \cdot k\mathbf{1}_{kN} = 0,$$

since the Jacobian when $W_{KQ}^{(\ell')} = 0$ is $J^{(\ell')} = \frac{1}{kN}(I - \frac{1}{kN}\mathbf{1}_{kN}\mathbf{1}_{kN}^\top)$. Therefore the gradient $\nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell')}$ on the ideal input is indeed equal to zero.

In mixed training, the nonsignal terms $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ for $\ell' \neq \ell$ can also play a role. For the case $t \leq \ell' < \ell$, we can generalize the above argument to compute the gradient $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ on the ideal input; the essential blocks extracted by $W_{OV}^{(\ell)}$ in the sequence are still $\frac{1}{kN}\mathbf{1}_{kN}$, which is independent of $W_{KQ}^{(\ell')}$, and thus the gradient on the ideal inputs is still zero. For the case $\ell' < t$, the softmax in layer ℓ' is fully saturated, and thus the gradient on ideal inputs is also zero. Finally, when $\ell' > \ell$, the gradient is trivially zero as well. Altogether in the tth step of the population dynamics, the only nonzero gradient (with ideal inputs) is $\nabla_{W_{KQ}^{(t)}}\mathcal{L}^{(t)}$, thus mimicking the curriculum dynamics.

Now we only need to upper bound the error that deviates from the population dynamics, including the sample noise, non-saturation error, and the additional error introduced during the accumulation when proprogating through layers. Compared to the proof of Theorem 7, there are **two additional source of error:**

1. During the tth gradient step, layers $\ell < t$ will still be updated. Since the softmax in layer ℓ is nearly saturated and thus close to one-hot, the update norm is very small and can be bounded as noise terms.

2. Due to the non-saturation error, the input is not ideal one-hot vectors for each hop. That means in the tth gradient step, there will be some small gradient updates for later layers $\ell' \geq t$ introduced by the perturbation. We can also upper bound this amount of noise.

The full proof proceeds as follows. In **stage 1** (Section D.2), we can use similar strategy as in Appendix C with curriculum. The proof is the same, since the later layers are not updated in this stage. **Note that now the sequence $\hat{X}^{(\ell)}$ evolves with time**, which is different from the analysis for Algorithm 1. We denote $\hat{X}_t^{(\ell)}$ as the intermediate sequence at time t, and each column as $\hat{X}_{t,(i,j)}^{(\ell)}$.

However, the gradient of the later layers ($\ell \geq 2$) after the first step ($t \geq 2$) is nonzero, because the first layer's attention pattern is not exactly one-hot. The error ϵ' introduced by softmax non-saturation accumulates to later layers, causing unwanted updates. Since the step-size is very large, each gradient step may introduce $\approx kNL\epsilon' \log \frac{1}{\epsilon}$ error, which means the accumulation error grows exponentially in the number of gradient steps ($t \leq L = O(\log k)$). Thus, we have to ensure that the error introduced by softmax is very small. In particular, we pick the error $\epsilon' = O(\frac{\epsilon}{(kNL)^{6L}})$; fortunately, this only requires the step size to be $\Omega(\frac{k^2N^3}{\beta_0}\log k \log \frac{kN}{\epsilon})$, which still satisfies the requirement in the theorem.

In later stages $t \geq 2$ (Section D.3 for stage 2, section D.4 for stage $t \geq 3$), we will inductively prove that $\left\|\hat{X}_t^{(\ell)} - X_t^{(\ell)}\right\| \lesssim \frac{\epsilon \log^{t-1}\frac{1}{\epsilon}}{(kNL)^{6L-6t+6}}$. Altogether, for the last stage L, the error can be bounded by $O(\frac{1}{k^6N^6L^6})$, which guarantees that the true transformer output $\hat{X}_L^{(L)}$ stores the correct $2^{\ell-1}$-hop information, as in the proof of Theorem 7.

To conclude, further training the readout layer leads to learning all the $2^{\ell-1}$-hop tasks, as desired (Lemma 36). ■

D.2. Stage 1 for Mixed Training

We first prove that in **stage 1**, the first layer $W_{KQ}^{(1)}$ learns all the hidden permutations. The proof is the same as the proof of Theorem 7, because there is no additional noise introduced in this stage by zero initialization.

Using Lemma 15, after the first step gradient we have for any (i, j), the softmax probability satisfies

$$\mathcal{S}^{(1)}_{(i,\pi_i(j))} := \mathcal{S}((\hat{X}^{(0)})^\top W_{KQ}^{(1)} \hat{X}^{(0)}_{(i,j)})_{(i,\pi_i(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{6L}}.$$

with a $\log(kN)$ larger learning rate. Note that $\log(kNL)^L = O(\log k \log kNL) \lesssim \operatorname{poly}\log(kN)$, so it falls in the required learning rate range. Before the softmax, we have the separation of the $(i, \pi_i(j))$ position and the others $\gtrsim \log k \log \frac{kN}{\epsilon}$.

Note that now the sequence $\hat{X}^{(\ell)}$ evolves with time. Recall $\hat{X}_t^{(\ell)}$ as the intermediate sequence at time t, and each column as $\hat{X}^{(\ell)}_{t,(i,j)}$. We can then calculate the intermediate sequence $\hat{X}_1^{(1)}$:

$$\hat{X}_1^{(1)} = X^{(0)} + W_{OV}^{(1)} X^{(0)} \mathcal{S}^{(1)} = X_1^{(1)} + W_{OV}^{(1)} X^{(0)} (\mathcal{S}^{(1)} - \mathcal{S}^{(1)}_{\text{ideal}}).$$

while the ideal input sequence for the second layer is

$$X_1^{(1)} = \left(\begin{array}{ccc|c|ccc} e_{k,1} \otimes e_{N,1} & \cdots & e_{k,1} \otimes e_{N,N} & \cdots & e_{k,k} \otimes e_{N,1} & \cdots & e_{k,1} \otimes e_{N,N} \\ e_{k,1} \otimes e_{N,\sigma_1(1)} & \cdots & e_{k,1} \otimes e_{N,\sigma_1(N)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k(1)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k(N)} \\ e_{k,1} \otimes e_{N,\sigma_1\pi_1(1)} & \cdots & e_{k,1} \otimes e_{N,\sigma_1\pi_1(N)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k\pi_k(1)} & \cdots & e_{k,k} \otimes e_{N,\sigma_k\pi_k(N)} \\ 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} & \cdots & 0_{(L-1)kN} \end{array}\right)$$

By Lemma 17, $\|X_1^{(1)} - \hat{X}_1^{(1)}\|_\infty \leq \frac{\epsilon}{(kNL)^{6L}}$. Further, we need to bound the distance between the ideal inputs for later layers $X_1^{(\ell)}$

$$X^{(\ell)}_{1,(i,j)} = \begin{bmatrix} e_{k,i} \otimes e_{N,j} \\ e_{k,i} \otimes e_{N,\sigma_i(j)} \\ e_{k,i} \otimes e_{N,\text{hop}_i^1(j)} \\ \frac{1}{kN} 1_{kN} \\ \vdots \\ \frac{1}{kN} 1_{kN} \\ 0_{(L-\ell)kN} \end{bmatrix}$$

and the actual input $\hat{X}_1^{(\ell)}$. The following lemma exhibits the upper bound.

Lemma 30 (Perturbation analysis of stage 1 for $X_1^{(\ell)}$) *Under the condition of Theorem 8, we have for all $\ell \geq 2$,*

$$\|X_1^{(\ell)} - \hat{X}_1^{(\ell)}\| \leq \frac{\epsilon}{(kNL)^{6L}},$$

where $X_1^{(\ell)}$ is the ideal output with saturated softmax, and $\hat{X}_1^{(\ell)}$ is the transformer output.

Proof The ideal output for the ℓ layer is

$$X^{(\ell)} = X^{(\ell-1)} + W_{KQ}^{(\ell)} X^{(\ell-1)} \mathcal{S}_{\text{ideal}}^{(\ell)},$$

where $\mathcal{S}_{\text{ideal}}^{(\ell)}$ is the ideal one-hot softmax attention pattern. The empirical output $\hat{X}_1^{(\ell)}$ has the error introduced by the non-saturation of the softmax, together with the previous error in $\hat{X}_1^{(\ell-1)}$:

$$\begin{aligned}\hat{X}_1^{(\ell)} &= \hat{X}_1^{(\ell-1)} + W_{OV}^{(\ell)} \hat{X}_1^{(\ell-1)} \mathcal{S}^{(\ell)} \\ &= X_1^{(\ell)} + W_{OV}^{(\ell)} X_1^{(\ell-1)} \underbrace{(\mathcal{S}^{(\ell)} - \mathcal{S}_{\text{ideal}}^{(\ell)})}_{\Delta\mathcal{S}^{(\ell)}\text{,non-uniform error}} + \underbrace{(\hat{X}^{(\ell-1)} - X^{(\ell-1)}) + W_{OV}^{(\ell)} (\hat{X}^{(\ell-1)} - X^{(\ell-1)}) \mathcal{S}^{(\ell)}}_{\text{Accumulated perturbation error}}.\end{aligned}$$

We first consider the non-uniform error term. Since the $W_{KQ}^{(\ell)}$ is not updated for $\ell \geq 2$,

$$\|W_{OV}^{(\ell)} X_1^{(\ell-1)} (\mathcal{S}^{(\ell)} - \mathcal{S}_{\text{ideal}}^{(\ell)})\|_\infty = 0.$$

Now consider the accumulated perturbation error. By the perturbation analysis inductively, we have $\left\|\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}\right\|_\infty \leq \frac{\epsilon}{(knL)^{6L}}$. Note that the error in $\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}$ are in different rows of the matrices from $W_{OV}^{(\ell)} (\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}) \mathcal{S}^{(\ell)}$. Therefore, $\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}$ won't introduce extra error in this stage, and we only need to consider $W_{OV}^{(\ell)} (\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}) \mathcal{S}^{(\ell)}$:

$$\begin{aligned}\left\|W_{OV}^{(\ell)} (\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}) \mathcal{S}^{(\ell)}\right\|_\infty &= \max_{s,(i,j)} \left|(W_{OV}^{(\ell)} (\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}))_s^\top \mathcal{S}_{(i,j)}^{(\ell)}\right| \\ &= \max_{s,(i,j)} \left|\sum_{p=1}^{kN} (W_{OV}^{(\ell)} (\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}))_{s,p} (\mathcal{S}_{(i,j)}^{(\ell)})_p\right| \\ &\leq \|\hat{X}_1^{(\ell-1)} - X_1^{(\ell-1)}\|_\infty \\ &\qquad \text{(Since } \textstyle\sum_p (\mathcal{S}_{(i,j)}^{(\ell)})_p = 1, (\mathcal{S}_{(i,j)}^{(\ell)})_p \geq 0.\text{)}\end{aligned}$$

Combine both parts of error, we have $\left\|\hat{X}_1^{(\ell)} - X_1^{(\ell)}\right\|_\infty \leq \frac{\epsilon}{(knL)^{6L}}$. ■

Thus after stage 1, the intermediate input sequences $\hat{X}_1^{(\ell)}$ are $\frac{\epsilon}{(knL)^{6L}}$-close to the ideal sequence.

D.3. Stage 2 for Mixed Training

After the first stage, $W_{KQ}^{(\ell)}$ for all $\ell \geq 2$ are not updated and remain zero. Given that $\left\|\hat{X}_1^{(\ell)} - X_1^{(\ell)}\right\|_\infty \leq \frac{\epsilon}{(knL)^{6L}}$, Lemma 19 still applies and we have

$$\mathcal{S}_{(i+1,\text{hop}_i^1(j))}^{(2)} := \mathcal{S}\Big((\hat{X}^{(1)})^\top W_{KQ}^{(2)} \hat{X}_{(i,j)}^{(1)}\Big)_{(i+1,\text{hop}_i^1(j))} \geq 1 - \frac{\epsilon}{2(knL)^{6L}}.$$

However, the first layer and the later layers with $\ell \geq 3$ are **all updated in the second stage** due to the perturbed inputs on the mixture of training data. The following lemma bounds the deviation of the gradient steps, making sure the empirical gradients stay close to the population dynamics.

Lemma 31 *In stage 2, given that* $\left\|\hat{X}_1^{(\ell)} - X_1^{(\ell)}\right\|_\infty \le \frac{\epsilon}{(knL)^{6L}}$ *for all gradients* $\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell')}$ *for* $\ell \neq 2$ *or* $\ell' \neq 2$*, the gradients can be upper bounded by*

$$\|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell')}\|_\infty \lesssim \frac{\beta_0 \epsilon}{(knL)^{6L-2.5}}.$$

Proof We first bound the possible signal term gradients $\nabla_{W_{KQ}^{(i)}} \mathcal{L}^{(i)}$, $i \neq 2$. We prove that the gradient norm for $W_{KQ}^{(\ell)}$ for $\ell \neq 2$ can be upper bounded by

$$\|\nabla_{W_{KQ}^{(1)}} \mathcal{L}^{(1)}\|_\infty \lesssim \frac{\beta_0 \epsilon}{(knL)^{6L-1}}, \quad \|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)}\|_\infty \lesssim \frac{\beta_0 L \epsilon}{(knL)^{6L}}.$$

Recall the gradient of ℓth layer on the sample X_m is:

$$\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)} = \sum_{s' \in [N]} (\mathbf{1}\{s' = \mathrm{hop}_{i_m}^{2^{\ell-1}}(j_m)\} - \mathcal{P}_{s'}^{(\ell)}) \hat{X}_m^{(\ell-1)} J^{(\ell)} (\hat{X}_m^{(\ell-1)})^\top \left(\Psi_\ell^\top W_{OV}^{(\ell)}\right)_{s'}^\top (\hat{X}_{(i_m,j_m)}^{(\ell-1)})^\top$$

For $\ell = 1$, since the softmax is close to one-hot, we have the Jacobian

$$\|J^{(1)}\|_2 = \|\operatorname{diag}(\mathcal{S}) - \mathcal{S}\mathcal{S}^\top\|_2 \lesssim \frac{\epsilon}{(knL)^{6L}}.$$

Therefore the first layer gradient has an infinity norm upper bound (since $\|\hat{X}_m^{(\ell)}\|_\infty \le 1$.)

$$\|\nabla_{W_{KQ}^{(1)}} \mathcal{L}^{(1)}\|_\infty \le 2d \left\|J^{(1)}\right\|_2 \cdot \beta_0 \cdot 1 \lesssim \frac{\beta_0 \epsilon}{(KNL)^{6L-1}}.$$

For $\ell \ge 3$, the ideal empirical gradient is zero. The perturbation satisfies $\left\|\hat{X}_1^{(\ell)} - X_1^{(\ell)}\right\|_\infty \le \frac{\epsilon}{(knL)^{6L}}$. With the same proof strategy in Lemma 25, we compare the idealized empirical gradient and actual empirical gradient. Here all key-query matrices are not updated, so the Jacobian is still $\frac{1}{kN}(I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top)$. The actual empirical gradient is (we ignore the layer number $^{(\ell)}$ here)

$$\begin{aligned}\hat{g}_\ell = &\frac{\beta_0}{kN} \hat{X}_1 (I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top) \hat{X}_1^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}) (\hat{X}_{1,(i,j)}^{(\ell-1)})^\top \\ &- \frac{\beta_0}{kN} \sum_{s' \in [N]} \mathcal{S}_{s'} \hat{X}_1 (I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top) \hat{X}_1^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'}) (\hat{X}_{1,(i,j)}^{(\ell-1)})^\top.\end{aligned}$$

Denote the following term as

$$\hat{\gamma}_{s';\ell} = \hat{X}_1 (I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top) \hat{X}_1^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'}) (\hat{X}_{1,(i,j)}^{(\ell-1)})^\top,$$

$$\gamma_{s';\ell} = X_1 (I - \frac{1}{kN}\mathbf{1}\mathbf{1}^\top) X_1^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'}) (X_{1,(i,j)}^{(\ell-1)})^\top.$$

and we define $\Delta\gamma_{s';\ell} = \hat{\gamma}_{s';\ell} - \gamma_{s';\ell}$. Then we can rewrite the empirical gradient into

$$\hat{g}_\ell - \hat{g}_\ell^{\mathrm{ideal}} = \Delta\gamma_{\mathrm{hop}^{2^{\ell-1}}(v_m),m} - \sum_{s' \in [N]} \mathcal{S}_{s'} \Delta\gamma_{s'}.$$

We have the perturbation error upper bounded by:

$$\left\|\hat{g}_\ell - \hat{g}_\ell^{\text{ideal}}\right\|_\infty \le \|\Delta\gamma_{\text{hop};\ell}\| + \sum_{s'} \mathcal{S}_{s'}\|\Delta\gamma_{s';\ell}\|. \tag{10}$$

The error of the following difference

$$\|\Delta\gamma_{s';\ell}\|_\infty = \|\gamma_{s';\ell} - \hat{\gamma}_{s';\ell}\|_\infty \le Cd\left\|\hat{X}_1 - X_1\right\|_\infty.$$

with some absolute constant. Thus the gradient is upper bounded by

$$\|\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{(\ell)}\|_\infty \lesssim \frac{\beta_0}{kN}\cdot\frac{d\epsilon}{(kNL)^{6L}} \lesssim \frac{\beta_0 L\epsilon}{(kNL)^{6L}}.$$

Next, we bound the gradients $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ with $\ell' < 2$, i.e. $\ell' = 1$. By Lemma 29 and $\mathcal{S}^{(2)}_{(i+1,\text{hop}_i^1(j))} \ge 1 - \frac{\epsilon}{2(kNL)^{6L}}$, we have

$$\left\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\|_\infty \le 6\beta_0 k(kNL)^{3/2}\cdot\frac{\epsilon}{(kNL)^{6L}} \lesssim \beta_0\cdot\frac{\epsilon}{(kNL)^{6L-2.5}}.$$

Finally, we bound the gradients $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ with $\ell > \ell' \ge 2$. Similar to Lemma 29, we can derive a general upper bound for those gradients; this is deferred to Lemma 37 in Section D.5. We have the upper bound with $\ell_0 = 1$:

$$\left\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\|_\infty \le 6\beta_0\cdot\frac{\epsilon\log^{\ell_0-1}\frac{1}{\epsilon}}{(kNL)^{6L-6\ell_0+5}} \lesssim \beta_0\cdot\frac{\epsilon}{(kNL)^{6L-1}}.$$

Combining all three parts where the worst bound is $O\Big(\frac{\beta_0\cdot\epsilon}{(kNL)^{6L-2.5}}\Big)$, we conclude the proof. ■

With these bounds for the gradients, we can further upper bound the distance between the ideal $X_2^{(\ell)}$ and the actual intermediate sequence $\hat{X}_2^{(\ell)}$. Here, $\ell = 1$ suffer from the first kind of additional error (further training after the layer learns the correct pattern), while $\ell \ge 3$ have the second type of additional error (unwanted updates from zero before the effective training stage).

Lemma 32 (Perturbation analysis of stage 2 for $X_2^{(\ell)}$) *Under the condition of Theorem 8, we have for all $\ell \in [L]$,*

$$\|X_2^{(\ell)} - \hat{X}_2^{(\ell)}\| \lesssim \frac{\epsilon}{(kNL)^{6L-6}},$$

where $X_2^{(\ell)}$ is the ideal output with one-hot attention pattern, and $\hat{X}_2^{(\ell)}$ is the transformer output.

Proof The ideal output for the ℓ layer is

$$X_2^{(\ell)} = X_2^{(\ell-1)} + W_{KQ}^{(\ell)}X_2^{(\ell-1)}\mathcal{S}_{\text{ideal}}^{(\ell)},$$

where $\mathcal{S}^{(\ell)}_{\text{ideal}}$ is the ideal one-hot attention pattern. The empirical output $\hat{X}_2^{(\ell)}$ has the error introduced by the non-saturation of the softmax, together with those from the previous error in $\hat{X}_2^{(\ell-1)}$:

$$\begin{aligned}\hat{X}_2^{(\ell)} &= \hat{X}_2^{(\ell-1)} + W_{OV}^{(\ell)}\hat{X}_2^{(\ell-1)}\mathcal{S}^{(\ell)} \\ &= X_2^{(\ell)} + W_{OV}^{(\ell)}X_2^{(\ell-1)}\underbrace{(\mathcal{S}_2^{(\ell)} - \mathcal{S}_{2,\text{ideal}}^{(\ell)})}_{\Delta\mathcal{S}_2^{(\ell)}} + \underbrace{(\hat{X}_2^{(\ell-1)} - X_2^{(\ell-1)}) + W_{OV}^{(\ell)}(\hat{X}_2^{(\ell-1)} - X_2^{(\ell-1)})\mathcal{S}_2^{(\ell)}}_{\text{Accumulated perturbation error}}.\end{aligned}$$

Layer 1: We first prove the case with $\ell = 1$, where the accumulated perturbation error is 0 since $X^{(0)}$ is always the original input. Therefore, we just consider the $\Delta\mathcal{S}_2^{(\ell)}$ term.

According to the last step in Lemma 15, the separation between correct and wrong positions before the softmax $\eta\hat{\Delta}$ is greater than $\Omega(\log k \log \frac{kN}{\epsilon})$. While by Lemma 31, the update in the second stage should be upper bounded by the sum of all gradient norms of $W_{KQ}^{(1)}$ in the second stage:

$$\sum_{\ell=1}^{L}\left\|\eta\nabla_{W_{KQ}^{(1)}}\mathcal{L}^{(\ell)}\right\|_\infty \le L\cdot k^2N^3\cdot\frac{\epsilon}{(KNL)^{6L-2.5}}\log k\cdot\log\frac{kN}{\epsilon} \ll \log k\log\frac{kN}{\epsilon},$$

which is dominated by the current parameter since $\epsilon \le O(\frac{1}{k^6N^6})$. Therefore, the softmax $\mathcal{S}_2^{(1)}$ is still close to one-hot by

$$\|\mathcal{S}_2^{(1)} - \mathcal{S}_{2,\text{ideal}}^{(1)}\|_\infty \le \frac{\epsilon}{2(KNL)^{6L}}.$$

and thus we have $\|X_2^{(1)} - \hat{X}_2^{(1)}\|_\infty \le \frac{\epsilon}{2(kNL)^{6L}}$.

Layer 2: For $\ell = 2$, we already have $\|\Delta\mathcal{S}_2^{(2)}\| \le \frac{\epsilon}{2(KNL)^{6L}}$, so the softmax error is upper bounded by $\frac{\epsilon}{2(KNL)^{6L}}$. Now consider the accumulated perturbation error. By the perturbation analysis, we have $\left\|\hat{X}_2^{(1)} - X_2^{(1)}\right\|_\infty \le \frac{\epsilon}{2(kNL)^{6L}}$. Note that the error in $\hat{X}_2^{(1)} - X_2^{(1)}$ are in different rows of the matrices from $W_{OV}^{(2)}(\hat{X}_2^{(1)} - X_2^{(1)})\mathcal{S}^{(2)}$. Therefore, $\hat{X}_2^{(1)} - X_2^{(1)}$ won't introduce extra error in this stage. By similar arguments in stage 1 we have $\|W_{OV}^{(2)}(\hat{X}_2^{(1)} - X_2^{(1)})\mathcal{S}^{(2)}\|_\infty \le \|\hat{X}_2^{(1)} - X_2^{(1)}\|_\infty$. Combine both parts of error, we have $\|X_2^{(2)} - \hat{X}_2^{(2)}\|_\infty \le \frac{\epsilon}{(kNL)^{6L}}$.

Layer ℓ: Finally, we inductively prove that for layer $\ell \ge 3$, the distance $\|X_2^{(\ell)} - \hat{X}_2^{(\ell)}\| \le \frac{\epsilon\log\frac{1}{\epsilon}}{(kNL)^{6L-6}}$. For the base case $\ell = 3$, we first consider the second accumulation term, which is upper bounded by the second layer:

$$\|W_{OV}^{(3)}(\hat{X}_2^{(2)} - X_2^{(2)})\mathcal{S}^{(3)}\|_\infty \le \|\hat{X}_2^{(2)} - X_2^{(2)}\|_\infty \le \frac{\epsilon}{(kNL)^{6L}}.$$

Now we prove that the $\|\Delta\mathcal{S}_2^{(\ell)}\|_\infty \le \frac{\epsilon\log\frac{1}{\epsilon}}{(kNL)^{6L-6}L}$. The parameter norm after the update for each $W_{KQ}^{(\ell)}$, $\ell > 2$ is upper bounded by $\eta\sum_{\ell'=1}^{L}\|\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{(\ell')}\|_2 \le \frac{L\epsilon}{(kNL)^{6L-2.5}}$ using Lemma 31. Therefore we can expand the softmax:

$$\left\|\mathcal{S}_2^{(\ell)} - \frac{1}{kN}\mathbf{1}_{kN}\right\|_\infty \le \eta\|\tilde{J}\cdot(\hat{X}_2^{(\ell-1)})^\top\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{\mathrm{M}}\hat{X}_{2,(i,j)}^{(\ell-1)}\|_\infty$$

$$\leq C \cdot \eta \cdot \frac{1}{kN} \cdot \sum_{\ell'=1}^{L} \|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell')}\|_2 \cdot L^2 \quad (\|\tilde{J}\|_2 \lesssim \tfrac{1}{kN}, \|\hat{X}_{2,(i,j)}^{(\ell-1)}\|_2 \lesssim L.)$$

$$\lesssim \frac{k^2N^3}{\beta_0} \log k \log \frac{kN}{\epsilon} \cdot \frac{1}{kN} \cdot d \frac{\beta_0 L\epsilon}{(kNL)^{6L-2.5}} \cdot L^2$$

$$\leq \frac{\epsilon \log \frac{1}{\epsilon}}{(kNL)^{6L-5.5}}. \qquad (k > \log k, kN > \log kN)$$

Since for the ideal $W_{OV}^{(3)} X_2^{(2)}$ is one-hot/all zero for each row, $\left\|W_{OV}^{(3)} X_2^{(2)} \Delta \mathcal{S}_2^{(\ell)}\right\|_\infty \leq \frac{\epsilon \log \frac{1}{\epsilon}}{(kNL)^{6L-5.5}}$. Combine both part, we have $\|X_2^{(3)} - \hat{X}_2^{(3)}\| \leq \frac{\epsilon \log \frac{1}{\epsilon}}{(kNL)^{6L-5.5}}$.

For $\ell \geq 4$, we first consider the second accumulation term, which is upper bounded by the induction hypothesis:

$$\|W_{OV}^{(\ell)}(\hat{X}_2^{(\ell-1)} - X_2^{(\ell-1)})\mathcal{S}^{(\ell)}\|_\infty \leq \|\hat{X}_2^{(\ell-1)} - X_2^{(\ell-1)}\|_\infty \leq \frac{\epsilon \log \frac{1}{\epsilon}}{(kNL)^{6L-5.5}}.$$

And since each row of ideal $W_{OV}^{(\ell)} X_2^{(\ell-1)}$ is either all-one/all-zero, $\left\|W_{OV}^{(\ell)} X_2^{(\ell-1)} \Delta \mathcal{S}_2^{(\ell)}\right\|_\infty = 0$. By induction, we finish the proof. ■

To conclude, the intermediate sequence $\hat{X}_2^{(\ell)}$ is $O(\frac{\epsilon \log \frac{1}{\epsilon}}{(kNL)^{6L-6}})$ close to the ideal $X_2^{(\ell)}$ after Stage 2 for all layer ℓ. In the next section, we will continue the induction and prove that the final error is still $1/\operatorname{poly}(kN)$ small.

D.4. Stage $t \geq 3$ for Mixed Training

Finally, we prove the rest of the stages still satisfy that for all $\ell \in [L], t \in [L]$,

$$\left\|\hat{X}_t^{(\ell)} - X_t^{(\ell)}\right\|_\infty \leq \frac{\epsilon \log^{t-1} \frac{1}{\epsilon}}{(kNL)^{6L-6t+6}},$$

the softmax score of the first layer satisfies

$$\mathcal{S}_{(i,\pi_i(j))}^{(1)} = \mathcal{S}((X^{(0)})^\top W_{KQ}^{(1)} X_{(i,j)}^{(0)}) \geq 1 - \frac{\epsilon}{2(kNL)^{6L}},$$

and the softmax score of ℓth layer ($1 < \ell < t$) satisfies

$$\mathcal{S}_{(i+2^{\ell-2},\operatorname{hop}_i^{2^{\ell-2}}(j))}^{(\ell)} := \mathcal{S}\Big((\hat{X}^{(\ell-1)})^\top W_{KQ}^{(\ell)} \hat{X}_{(i,j)}^{(\ell-1)}\Big)_{(i+2^{\ell-2},\operatorname{hop}_i^{2^{\ell-2}}(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{6L}}.$$

We prove this by induction, and we already have $t = 2$ as our induction hypothesis.

First, we prove that for each stage t, $\nabla_{W_{KQ}^{(t)}} \mathcal{L}$ is close to the population gradient, while the other gradients can be upper bounded using the perturbation $\left\|\hat{X}_{t-1}^{(\ell)} - X_{t-1}^{(\ell)}\right\|_\infty$ from the previous timestep.

Similar to stage 2, we first control the deviation of gradients $\nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}$ from their idealized versions, where $\ell' \neq t$ or $\ell \neq t$. Since when $\ell' > \ell$ the gradient is zero by definition, we only consider $\ell \geq \ell'$. We first consider the cases when $\ell = \ell'$ and then $\ell > \ell'$.

Lemma 33 *In stage $t \geq 3$, given $\left\|\hat{X}_{t-1}^{(\ell)} - X_{t-1}^{(\ell)}\right\|_\infty \leq \frac{\epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}}$, the gradient error of $W_{KQ}^{(\ell)}$ from the idealized gradient can be upper bounded by*

- *If $\ell < t$: $\|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)}\|_\infty \lesssim \frac{\beta_0 \epsilon}{(knL)^{6L-1}}$, where idealized gradient is zero.*
- *If $\ell > t$: $\|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)}\|_\infty \lesssim \frac{\beta_0 \epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}}$, where idealized gradient is zero.*
- *If $\ell = t$: $\left\|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)} - \nabla_{W_{KQ}^{(\ell)}} \mathcal{L}_{\mathcal{D}}^{(\ell)}(X_{t-1})\right\|_\infty \lesssim \frac{\beta_0 \epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}} + \frac{\log(d/\delta)}{\sqrt{M}}$, where $\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}_{\mathcal{D}}^{(\ell)}(X_{t-1})$ is the population gradient with the idealized input $X_{t-1}^{(\ell-1)}$ to the ℓth layer and the ℓth key query matrix set to zero: $W_{KQ}^{(\ell)} = 0$.*

Proof We consider $\ell < t$ first. Since the softmax is close to one-hot by $O\left(\frac{\epsilon}{(knL)^{6L}}\right)$, we have the Jacobian's spectral norm upper bounded by

$$\|J^{(\ell)}\|_2 = \|\operatorname{diag}(\mathcal{S}) - \mathcal{S}\mathcal{S}^\top\|_2 \lesssim \frac{\epsilon}{(knL)^{6L}}.$$

Therefore the first layer gradient has an infinity norm upper bound (since $\|\hat{X}_m^{(\ell)}\|_\infty \leq 1$.)

$$\|\nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell)}\|_\infty \leq 2d \left\|J^{(1)}\right\|_2 \cdot \beta_0 \cdot 1 \lesssim \frac{\beta_0 \epsilon}{(KNL)^{6L-1}}.$$

For $\ell = t$, we apply a similar strategy as in Lemma 25. The sample noise with idealized input sequences X_{t-1} can be upper bounded by $O\left(\frac{\log(d/\delta)}{\sqrt{M}}\right)$, and we need to further upper bound the perturbation error.

The actual empirical gradient is

$$\begin{aligned}\hat{g}_\ell = {} & \beta_0 \hat{X}_{t-1} J_{t-1}^{(\ell)} \hat{X}_{t-1}^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,\operatorname{hop}_i^{2^{\ell-1}}(j)}) (\hat{X}_{t-1,(i,j)}^{(\ell-1)})^\top \\ & - \beta_0 \sum_{s' \in [N]} \mathcal{S}_{s'} \hat{X}_{t-1} J_{t-1}^{(\ell)} \hat{X}_{t-1}^\top (e_{L+2,\ell+1} \otimes \mathbf{1}_k \otimes e_{N,s'}) (\hat{X}_{t-1,(i,j)}^{(\ell-1)})^\top.\end{aligned}$$

Given that $\left\|\hat{X}_{t-1}^{(\ell)} - X_{t-1}^{(\ell)}\right\|_\infty \leq \frac{\epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}}$, and by induction of the last stage, the softmax score at initialization of stage t is

$$\left\|\mathcal{S}_{t-1}^{(\ell)} - \frac{1}{kN} \mathbf{1}_{kN}\right\|_\infty \leq \frac{\epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}}.$$

The perturbation error that lies in the Jacobian $J_{t-1}^{(\ell)}$ and $\hat{X}_{t-1}^{(\ell)}$ can be upper bounded by $O\left(\frac{\beta_0 \epsilon \log^{t-2} \frac{1}{\epsilon}}{(knL)^{6L-6t+12}}\right)$. Combine the two error terms and we finished the proof for $\ell = t$.

For $\ell \geq t+1$, the proof is similar to $\ell = t$. Since the idealized empirical gradient is always zero, we just need to bound the perturbation error, which is the same as $\ell = t$. The upper bound is also $O\Big(\frac{\beta_0 \epsilon \log^{t-2} \frac{1}{\epsilon}}{(kNL)^{6L-6t+12}}\Big)$. ■

Next, we bound the non-signal gradients $\nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}$ with $\ell > \ell'$. We first apply the bounds Lemma 29 for $\nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}$ with $\ell' < t$. By the softmax lower bound $\mathcal{S}^{(\ell')}_{(i+2^{\ell'-1}, \mathrm{hop}_i^{2^{\ell'-2}}(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{6L}}$ by induction before the gradient in stage t applies, we have

$$\Big\| \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)} \Big\|_F \leq 6\beta_0 k (kNL)^{3/2} \cdot \frac{\epsilon}{(kNL)^{6L}} \lesssim \beta_0 \cdot \frac{\epsilon}{(kNL)^{6L-2.5}}.$$

Finally, to upper bound the terms with $\ell > \ell' \geq t$, we apply Lemma 37 (in Section D.5). The upper bound is $O\Big(\frac{\beta_0 \epsilon \log^{t-2} \frac{1}{\epsilon}}{(kNL)^{6L-6t+11}}\Big)$.

Combining all of the above results, for $t \geq 2$ we have that each gradient error with $\ell \neq t$ or $\ell' \neq t$ is in the worst-case upper bounded by

$$\Big\| \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)} \Big\|_F \lesssim \beta_0 \cdot \frac{\epsilon}{(kNL)^{\min\{6L-6t+11, 6L-2.5\}}} \leq \beta_0 \cdot \frac{\epsilon}{(kNL)^{6L-6t+9.5}}.$$

Given the above bounds on the empirical gradients, we next prove that after one gradient step: (1) the tth layer learns the correct signal and (2) the other layers' unwanted errors can be controlled as predicted in the induction hypothesis. Those are shown in the following two lemmas. The proof strategy resembles Lemma 32.

Lemma 34 *For $\ell \leq t$, for all query (i,j) and input X, we have after the tth gradient step*

$$\mathcal{S}^{(\ell)}_{(i+2^{\ell-2}, \mathrm{hop}_i^{2^{\ell-2}}(j))} := \mathcal{S}\Big((\hat{X}^{(\ell-1)})^\top W_{KQ}^{(\ell)} \hat{X}^{(\ell-1)}_{(i,j)} \Big)_{(i+2^{\ell-2}, \mathrm{hop}_i^{2^{\ell-2}}(j))} \geq 1 - \frac{\epsilon}{2(kNL)^{6L}}.$$

Furthermore, we have $\|X_t^{(\ell)} - \hat{X}_t^{(\ell)}\|_\infty \leq \frac{\epsilon}{2(kNL)^{6L}}$.

Proof For each layer $\ell < t$, recall that the separation between the correct position and the other positions before the one-step gradient in stage t is $\Omega(\log k \log \frac{kN}{\epsilon})$. By Lemma 33 and Lemma 29, the update in the second stage should be upper bounded by

$$\sum_{\ell'=1}^{L} \Big\| \eta \nabla_{W_{KQ}^{(\ell)}} \mathcal{L}^{(\ell')} \Big\|_\infty \leq L \cdot k^2 N^3 \cdot \frac{\epsilon}{(KNL)^{6L-2.5}} \log k \cdot \log \frac{kN}{\epsilon} \ll \log k \log \frac{kN}{\epsilon},$$

which is dominated by the current parameter since $\epsilon \leq O(\frac{1}{k^6 N^6})$. Therefore, the softmax $\mathcal{S}_2^{(1)}$ is still close to one-hot by

$$\|\mathcal{S}_t^{(\ell)} - \mathcal{S}_{t,\mathrm{ideal}}^{(\ell)}\|_\infty \leq \frac{\epsilon}{2(KNL)^{6L}}.$$

and thus we have $\|X_t^{(\ell)} - \hat{X}_t^{(\ell)}\|_\infty \le \frac{\epsilon}{2(kNL)^{6L}}$.

For $\ell = t$, we upper bound the error of the gradient estimate by Lemma 33. The previous case with $\ell = t-1$ guarantees that the intermediate input $\hat{X}_t^{(t-1)}$ is close to the ideal input:

$$\|X_t^{(t-1)} - \hat{X}_t^{(t-1)}\|_\infty \le \frac{\epsilon}{2(kNL)^{6L}}.$$

We define the separation of the pre-softmax attention score between the correct position $(i + 2^{t-2}, \mathrm{hop}_i^{2^{t-2}}(j))$ and the others as $\hat{\Delta}_{i,j}^{(\ell)}$:

$$\hat{\Delta}_{i,j}^{(\ell)} = \left(\hat{X}^\top \hat{g}_\ell \hat{X}_{(i,j)}\right)_{(i+2^{t-2}, \mathrm{hop}_i^{2^{t-2}}(j))} - \max_{p \neq (i+2^{t-2}, \mathrm{hop}_i^{2^{t-2}}(j))} \left(\hat{X}^\top \hat{g}_\ell \hat{X}_{(i,j)}\right)_p.$$

By Lemma 24 and the upper bound for the perturbation, we have $\|\hat{\Delta}_{i,j}^{(\ell)}\|$ at least $\frac{\eta\beta_0}{k^2N^3}$. After one step gradient with learning rate $\eta \gtrsim \frac{k^2N^3}{\beta_0} \log k \log \frac{kN}{\epsilon}$, the softmax output of the correct position can be lower bounded by

$$\mathcal{S}(\hat{X}_m^\top W_{KQ}^{(\ell)} \hat{X}_{(i_m, j_m)})_{i_m, \mathrm{hop}_{i_m}^1(j_m)} \ge \frac{\exp\left(\frac{\eta\beta_0}{kN} \cdot \frac{1}{kN^2}\right)}{\exp\left(\frac{\eta\beta_0}{kN} \cdot \frac{1}{kN^2}\right) + kN - 1} \ge 1 - \frac{\epsilon}{2(kNL)^{6L}}.$$

Thus we have $\|X_t^{(t)} - \hat{X}_t^{(t)}\|_\infty \le \frac{\epsilon}{2(kNL)^{6L}}$, which concludes the proof. ■

The second lemma tracks the perturbation error for later layers.

Lemma 35 (Perturbation analysis of stage t for $X_t^{(\ell)}$) *Under the condition of Theorem 8, we have for all $\ell \ge t+1$,*

$$\|X_t^{(\ell)} - \hat{X}_t^{(\ell)}\| \lesssim \frac{\epsilon \log^{t-1} \frac{1}{\epsilon}}{(kNL)^{6L-6t+6}},$$

where $X_2^{(\ell)}$ is the ideal output with one-hot attention pattern, and $\hat{X}_2^{(\ell)}$ is the transformer output.

Proof Recall the decomposition of the transformer intermediate sequence

$$\begin{aligned}\hat{X}_t^{(\ell)} &= \hat{X}_t^{(\ell-1)} + W_{OV}^{(\ell)} \hat{X}_t^{(\ell-1)} \mathcal{S}^{(\ell)} \\ &= X_t^{(\ell)} + W_{OV}^{(\ell)} X_t^{(\ell-1)} \underbrace{(\mathcal{S}_t^{(\ell)} - \mathcal{S}_{t,\mathrm{ideal}}^{(\ell)})}_{\Delta\mathcal{S}_t^{(\ell)}} + \underbrace{(\hat{X}_t^{(\ell-1)} - X_t^{(\ell-1)}) + W_{OV}^{(\ell)}(\hat{X}_t^{(\ell-1)} - X_t^{(\ell-1)})\mathcal{S}_t^{(\ell)}}_{\text{Accumulated perturbation error}}.\end{aligned}$$

We inductively prove that for layer $\ell \ge t+1$, the distance $\|X_t^{(\ell)} - \hat{X}_t^{(\ell)}\| \le \frac{\epsilon \log^{t-1} \frac{1}{\epsilon}}{(kNL)^{6L-6t+6}}$. For the base case $\ell = t+1$, we first consider the second accumulation term, which is upper bounded by:

$$\|W_{OV}^{(t+1)}(\hat{X}_t^{(t)} - X_t^{(t)})\mathcal{S}^{(t+1)}\|_\infty \le \|\hat{X}_t^{(t)} - X_t^{(t)}\|_\infty \le \frac{\epsilon}{(kNL)^{6L}}.$$

Now we prove that the $\|\Delta\mathcal{S}_t^{(\ell)}\|_\infty \le \frac{\epsilon \log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}L}$. We expand the softmax:

$$\begin{aligned}
\left\|\mathcal{S}_t^{(\ell)} - \frac{1}{kN}\mathbf{1}_{kN}\right\|_\infty &\le \|\tilde{J}_t \cdot (\hat{X}_t^{(\ell-1)})^\top \left(\eta\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{\mathbf{M}} + W_{KQ}^{(\ell)(t)}\right)\hat{X}_{t,(i,j)}^{(\ell-1)}\|_\infty \\
&\le C\cdot\frac{1}{kN}\cdot\left(\sum_{\ell'=1}^{L}\eta\|\nabla_{W_{KQ}^{(\ell)}}\mathcal{L}^{(\ell)}\|_2 + \left\|W_{KQ}^{(\ell)}(t)\right\|_2\right)\cdot L^2 \\
&\qquad\qquad (\|\tilde{J}\|_2 \lesssim \tfrac{1}{kN}, \|\hat{X}_{t,(i,j)}^{(\ell-1)}\|_2 \lesssim L.) \\
&\lesssim \frac{k^2N^3}{\beta_0}\log k\log\frac{kN}{\epsilon}\cdot\frac{1}{kN}\cdot d\frac{\beta_0\epsilon\log^{t-2}\frac{1}{\epsilon}}{(knL)^{6L-6t+9.5}}\cdot L^2 \\
&\qquad\qquad (\left\|W_{KQ}^{(\ell)}(t)\right\|_2 \le \tfrac{\epsilon\log^{t-2}\frac{1}{\epsilon}}{(knL)^{6L-6t+9.5}}) \\
&\le \frac{\epsilon\log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}L}. \qquad (k>\log k, kN>\log kN)
\end{aligned}$$

Since for the ideal $W_{OV}^{(\ell)}X_t^{(\ell-1)}$ is one-hot/all zero for each row, $\left\|W_{OV}^{(\ell)}X_t^{(\ell-1)}\Delta\mathcal{S}_t^{(\ell)}\right\|_\infty \le \frac{\epsilon\log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}L}$. Combine both part, we have $\|X_t^{(\ell)} - \hat{X}_t^{(\ell)}\| \le \frac{\epsilon\log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}}$. Further, this also indicates that the 2-norm of $\left\|W_{KQ}^{(\ell)}(t)\right\|_2$ is upper bounded by

$$\left\|W_{KQ}^{(\ell)}(t)\right\|_2 \le \frac{\epsilon\log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}}.$$

For $\ell \ge t+2$, we first consider the second accumulation term, which is upper bounded by the induction hypothesis:

$$\|W_{OV}^{(\ell)}(\hat{X}_t^{(\ell-1)} - X_t^{(\ell-1)})\mathcal{S}^{(\ell)}\|_\infty \le \|\hat{X}_t^{(\ell-1)} - X_t^{(\ell-1)}\|_\infty \le \frac{\epsilon\log^{t-2}\frac{1}{\epsilon}}{(knL)^{6L-6t+12}}.$$

And since each row of ideal $W_{OV}^{(\ell)}X_t^{(\ell-1)}$ is either all-one/all-zero, $\left\|W_{OV}^{(\ell)}X_t^{(\ell-1)}\Delta\mathcal{S}_t^{(\ell)}\right\|_\infty = 0$. Therefore, the upper bound should be $O\Big(\frac{\epsilon\log^{t-1}\frac{1}{\epsilon}}{(knL)^{6L-6t+6}}\Big)$. By induction, we finish the proof. ■

When the induction comes to $\ell = L$, we correctly have all the $2^{\ell-1}$-hops encoded in the pre-readout output $X^{(L)}$. At the end of the mix training, we further train the readout layer with one gradient step to output all the correct $2^{\ell-1}$-hops for each position.

Lemma 36 *After one step gradient on θ_Ψ we have for all $\ell \in [L]$,*

$$\sup_{\sigma,(i,j)}\left\|\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X^{(\ell-1)})_{(i,j)}) - e_{\mathrm{hop}_i^{2^{\ell-1}}(j)}\right\|_\infty \le \epsilon.$$

Proof We calculate the population gradient for Ψ_ℓ, and then do the finite sample analysis. Note that when training on the mixture of data, we can learn all the readout layers at once.

The population gradient for the ℓth readout layer is

$$\nabla_{\Psi_\ell}\mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) = -\mathbb{E}\left[\left(e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} - \mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)})\right)(f^{(\ell)}(X)_{(i,j)})^\top\right]$$

By Lemma 34, the output of the second layer is

$$\begin{aligned} f^{(\ell)}(X)_{(i,j)} &= W_{OV}^{(\ell)}\hat{X}^{(\ell-1)}\mathcal{S}^{(\ell)} = W_{OV}^{(\ell)}X^{(\ell-1)}\mathcal{S}_{\text{ideal}}^{(\ell)} + W_{OV}^{(\ell)}(\hat{X}^{(\ell-1)} - X^{(\ell-1)})\mathcal{S}_{\text{ideal}}^{(\ell)} + W_{OV}^{(\ell)}X^{(\ell-1)}\Delta\mathcal{S}^{(\ell)} \\ &= e_{L+2,\ell+2}\otimes e_{k,i}\otimes e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \Delta_\ell. \end{aligned}$$

where $\|\Delta_\ell\|_\infty \le \frac{\epsilon}{(kNL)^{6L}}$ by Lemma 34. Since $\Psi_\ell(0) = \beta_0 e_{L+2,\ell+2}\otimes \mathbf{1}_k \otimes I_{N\times N}$, we have the expansion

$$\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)}) = \mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \Psi_\ell^\top\Delta_\ell) = \mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}) + \tilde{J}\Psi_\ell^\top\Delta_\ell.$$

Since $\|\Delta_\ell\|_\infty \le \epsilon$, we have $\left\|\tilde{J}\Psi_\ell^\top\Delta_\ell\right\|_\infty \le \beta_0\epsilon$. The signal term

$$\mathcal{S}(\beta_0 e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)}) = \frac{\exp(\beta_0)-1}{\exp(\beta_0)+N-1}e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \frac{1}{\exp(\beta_0)+N-1}\mathbf{1}_N.$$

The population gradient is thus

$$\begin{aligned} \nabla_{\Psi_\ell}\mathcal{L}_{\mathcal{D}}^{(\ell)}(\theta) &= -\mathbb{E}\left[\left(\frac{N}{\exp(\beta_0)+N-1}e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} - \frac{1}{\exp(\beta_0)+N-1}\mathbf{1}_N - \tilde{J}\Psi_\ell^\top\Delta_\ell\right)(f^{(\ell)}(X)_{(i,j)})^\top\right] \\ &= -\frac{1}{(\exp(\beta_0)+N-1)k}\left(e_{L+2,\ell+2}\otimes\mathbf{1}_k\otimes(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)\right) + O(\epsilon). \end{aligned}$$

where $O(\epsilon)$ denotes the terms with infinity norm smaller than ϵ with $\epsilon \lesssim \frac{1}{k^6N^6}$. The error altogether is upper bounded by $O(\frac{1}{k^6N^6})$ in infinity norm.

After one step of gradient, we have the softmax score (Δ is the error term with $\|\Delta\|_\infty \le \epsilon$.)

$$\mathcal{S}(\Psi_\ell(1)^\top f^{(\ell)}(X)_{(i,j)})) = \mathcal{S}\left(\frac{\eta}{(\exp(\beta_0)+N-1)k}(I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \beta_0 e_{N,\mathrm{hop}_i^{2^{\ell-1}}(j)} + \eta\Delta\right)$$

The separation between the correct entry and the others are lower bounded by:

$$\frac{\eta}{(\exp(\beta_0)+N-1)k}\frac{N-1}{N} - \eta\|\Delta\|_\infty \gtrsim \frac{\eta}{kN}.$$

By $\eta \gtrsim k^2N^3\log\frac{kN}{\epsilon}$, we have $\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)})_{\mathrm{hop}_i^{2^{\ell-1}}(j)} \ge 1-\epsilon$ and thus

$$\sup_{\sigma,(i,j)}\left\|\mathcal{S}(\Psi_\ell^\top f^{(\ell)}(X)_{(i,j)}) - e_{\mathrm{hop}_i^{2^{\ell-1}}(j)}\right\|_\infty \le \epsilon.$$

■

D.5. Gradient Upper Bounds for Mixed Training

In the analysis of the curriculum learning algorithm, we already proved a gradient upper bound for $\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}$ when $\ell' < \ell, \ell \le t$ in the tth stage (Lemma 29). However, those are just half of the non-signal terms in the case of mix training. The following lemma addresses the cases with $\ell > \ell' \ge t$.

Lemma 37 *Given a single sample (X, i, j) and suppose the training is in stage ℓ_0 (before the gradient step). If for all $\ell \ge \ell_0$,*

$$\left\|\mathcal{S}_{\ell_0}^{(\ell)} - \frac{1}{kN}\mathbf{1}_{kN}\right\|_\infty \le \frac{\epsilon \log^{\ell_0-1}\frac{1}{\epsilon}}{(kNL)^{6L-6\ell_0+6}}$$

and for all $\ell' < \ell_0$,

$$\mathcal{S}((\hat{X}^{(\ell'-1)})^\top W_{KQ}^{(\ell')}\hat{X}_{(i,j)}^{(\ell'-1)})_{(i+2^{\ell'-2},\mathrm{hop}_i^{2^{\ell'-2}}(j))} \ge 1 - \frac{\epsilon}{2(kNL)^{6L}}.$$

Then we have for any $\ell > \ell_0, \ell_0 \le \ell' < \ell$, the infinity norm of the gradient over $\mathcal{L}^{(\ell)}$ is upper bounded by

$$\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\|_\infty \le 6\beta_0 \cdot \frac{\epsilon \log^{\ell_0-1}\frac{1}{\epsilon}}{(kNL)^{6L-6\ell_0+5}}.$$

Proof Recall the gradient for a single sample (X, i, j)

$$\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)} = -\left[\sum_{s'\in[N]}\left(\mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} - \mathcal{S}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right)\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right]$$

So the norm of the gradient is upper bounded by

$$\begin{aligned}\left\|\nabla_{W_{KQ}^{(\ell')}}\mathcal{L}^{(\ell)}\right\| &\le \left\|\sum_{s'\in[N]}\left(\mathbf{1}\{s' = \mathrm{hop}_i^{2^{\ell-1}}(j)\} - \mathcal{S}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right)\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right\| \\ &\le 2\left\|\nabla_{W_{KQ}^{(\ell')}}(\Psi_\ell^\top f_{(i,j)}^{(\ell)})_{s'}\right\| = 2\left\|\nabla_{W_{KQ}^{(\ell')}}(e_{N,s'}^\top\Psi_\ell^\top f_{(i,j)}^{(\ell)})\right\|.\end{aligned}$$

Now we calculate the gradient recursively through Taylor expansion: for all ΔW with $\|\Delta W\| \to 0$,

$$f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')} + \Delta W) = f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')}) + \langle\nabla_{W_{KQ}^{(\ell')}}(f_{(i,j)}^{(\ell)}), \Delta W\rangle + O(\|\Delta W\|_F^2).$$

While by Taylor expansion, we have (we ignore $(W_{KQ}^{(\ell')})$ when there is no perturbation ΔW.)

$$\begin{aligned}f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')} + \Delta W) = f_{(i,j)}^{(\ell)}(W_{KQ}^{(\ell')}) &+ W_{OV}^{(\ell)}\nabla_{W_{KQ}^{(\ell')}}X^{(\ell-1)}(\Delta W)\mathcal{S}((X^{(\ell-1)})^\top W_{KQ}^{(\ell)}X_{(i,j)}^{(\ell-1)}) \\ &+ W_{OV}^{(\ell)}X^{(\ell-1)}J^{(\ell)}\nabla\mathcal{S}((X^{(\ell-1)})^\top W_{KQ}^{(\ell)}X_{(i,j)}^{(\ell-1)})(\Delta W) + O(\|\Delta W\|_F^2).\end{aligned}$$

Here the gradient $\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)\in\mathbb{R}^{d\times kN\times d\times d}}$ is a 4-th order tensor, and $\nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W) \in \mathbb{R}^{d\times kN}$. We upper bound the infinity norm of the matrix by induction from layer ℓ' to ℓ. We prove that

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\|_\infty \le \left(1+\frac{1}{k}\right)^{t-\ell'} \frac{L\cdot\epsilon\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}\|\Delta W\|_F$$

with $t\in[\ell',\ell-1]$.

Base case: $t=\ell'$ By Taylor expansion on $X^{(\ell')}$, we have

$$X^{(\ell')}(W_{KQ}^{(\ell')}+\Delta W) = X^{(\ell')} + W_{OV}^{(\ell')}X^{(\ell'-1)}J^{(\ell')}(X^{(\ell'-1)})^\top \Delta W X^{(\ell'-1)}) + O(\|\Delta W\|_F^2).$$

since previous layers are independent of $W_{KQ}^{(\ell')}$. Therefore, the first order term is

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\| = \left\|W_{OV}^{(\ell')}X^{(\ell'-1)}J^{(\ell')}(X^{(\ell'-1)})^\top \Delta W X^{(\ell'-1)})\right\|$$

Note that $\left\|\mathcal{S}_{\ell_0}^{(\ell)} - \frac{1}{kN}\mathbf{1}_{kN}\right\|_\infty \le \frac{\epsilon\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}$, we have the Jacobian

$$\left\|J^{(\ell)} - \frac{1}{kN}\left(I-\frac{1}{kN}\mathbf{1}\mathbf{1}^\top\right)\right\| \lesssim \frac{\epsilon}{kN}\cdot\frac{\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}.$$

Note that $W_{OV}^{(\ell')}X_{\text{ideal}}^{(\ell'-1)} = \frac{1}{kN}(e_{L+2,\ell'+2}\otimes\mathbf{1}_{kN})\mathbf{1}_{kN}^\top$, which cancels with the ideal Jacobian. The excess error is also upper bounded by $\frac{\epsilon}{kN}\cdot\frac{\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}$.

Therefore, we can upper bound the first order term by

$$\begin{aligned}
\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_2 &\le \left\|W_{OV}^{(\ell')}X^{(\ell'-1)}J^{(\ell')}\right\|_2\left\|(X^{(\ell'-1)})^\top\right\|_F\|\Delta W\|\left\|X^{(\ell'-1)})\right\|_F \\
&\lesssim knL\cdot\frac{\epsilon}{kN}\cdot\frac{\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}\|\Delta W\|_F. \\
(\left\|X^{(\ell'-1)}\right\|_F^2 &\le O(knL) \text{ since each embedding is either } \epsilon\text{-close to one-hot or all 0.}) \\
&\lesssim \frac{L\cdot\epsilon\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}\|\Delta W\|_F.
\end{aligned}$$

Since $\|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_\infty \le \|\nabla_{W_{KQ}^{(\ell')}} X^{(\ell')}(\Delta W)\|_2$, we finish the proof for base case.

Induction: $t\in[\ell',\ell-1]$**.** Suppose the induction hypothesis holds:

$$\|\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)\|_\infty \le \left(1+\frac{1}{k}\right)^{t-\ell'} \frac{L\cdot\epsilon\log^{\ell_0-1}\frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}}\|\Delta W\|_F.$$

We consider expanding $X^{(t+1)}(W_{KQ}^{(\ell')}+\Delta W)$ like the base case:

$$X^{(t+1)}(W_{KQ}^{(\ell')}+\Delta W) = X^{(t)}(W_{KQ}^{(\ell')}+\Delta W) + f^{(t+1)}(W_{KQ}^{(\ell')}+\Delta W)$$

$$
\begin{aligned}
&= X^{(t+1)} + \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) && \text{(From } X^{(t)}(W_{KQ}^{(\ell')} + \Delta W)\text{.)} \\
&+ W_{OV}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \mathcal{S}\bigg(\Big(X^{(t)}\Big)^\top W_{KQ}^{(t+1)} X^{(t)} \bigg) \\
&+ W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \bigg(\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \bigg)^\top W_{KQ}^{(t+1)} X^{(t)} \\
&+ W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \Big(X^{(t)}\Big)^\top W_{KQ}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) + O(\|\Delta W\|_F^2) \\
&&& \text{(The last 3 terms are from } f^{(t+1)}(W_{KQ}^{(\ell')} + \Delta W)\text{.)}
\end{aligned}
$$

Note that $\nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)$ and the rest three terms are in different rows since they uses different $W_{OV}^{(t)}$, so the infinity norm of the first order term should be the maximum of those two.

We first upper bound the last three terms. Since $W_{OV}^{(t+1)}$ is partial identity, and softmax is some weighted average, we have

$$
\bigg\| W_{OV}^{(t+1)} \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \mathcal{S}\bigg(\Big(X^{(t)}\Big)^\top W_{KQ}^{(t+1)} X^{(t)} \bigg) \bigg\| \le \bigg\| \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W) \bigg\|_\infty .
$$

The rest two terms can be directly upper bounded by

$$
\Big\| W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \Big\|_2 \bigg\| \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)) \bigg\|_2 \Big\| W_{KQ}^{(t+1)} \Big\|_2 \Big\| X^{(t)} \Big\|_2 .
$$

In previous stages, we have $\|X\|_F^2 \le O(kNL)$, $\Big\| W_{OV}^{(t+1)} X^{(t)} J^{(t+1)} \Big\|_2 \le \frac{\epsilon}{kN} \cdot \frac{\log^{\ell_0 - 1} \frac{1}{\epsilon}}{(kNL)^{6L - 6\ell_0 + 6}}$ and $\|W_{KQ}^{(t+1)}\|_2 \lesssim O(1)$. So these two terms can be both upper bounded by $\frac{L\epsilon \log^{\ell_0 - 1} \frac{1}{\epsilon}}{(kNL)^{6L - 6\ell_0 + 6}} \bigg\| \nabla_{W_{KQ}^{(\ell')}} X^{(t)}(\Delta W)) \bigg\|_2$
Combining two error terms, we have

$$
\bigg\| \nabla_{W_{KQ}^{(\ell')}} X^{(t+1)}(\Delta W) \bigg\|_\infty \le (1 + \frac{1}{k})^{t - \ell' + 1} \frac{L\epsilon \log^{\ell_0 - 1} \frac{1}{\epsilon}}{(kNL)^{6L - 6\ell_0 + 6}} \|\Delta W\|_F .
$$

By induction, when $t = \ell - 1$ we have (since $\ell - \ell' < k$.)

$$
\bigg\| \nabla_{W_{KQ}^{(\ell')}} X^{(\ell - 1)}(\Delta W) \bigg\|_\infty \le (1 + \frac{1}{k})^{\ell - \ell'} \frac{L\epsilon \log^{\ell_0 - 1} \frac{1}{\epsilon}}{(kNL)^{6L - 6\ell_0 + 6}} \|\Delta W\|_F \le 3 \frac{L\epsilon \log^{\ell_0 - 1} \frac{1}{\epsilon}}{(kNL)^{6L - 6\ell_0 + 6}} \|\Delta W\|_F .
$$

Finally, we can upper bound $\|\nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}\|$ by picking $\Delta W = \alpha \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}$ with $\alpha \to 0$:

$$
\begin{aligned}
\alpha \bigg\| \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)} \bigg\|_F^2 &\le 2 \Big\langle \nabla_{W_{KQ}^{(\ell')}} (e_{N,s'}^\top \Psi_\ell^\top f_{(i,j)}^{(\ell)}), \Delta W \Big\rangle \\
&\le 2 e_{N,s'}^\top \Psi_\ell^\top W_{OV}^{(\ell)} \nabla_{W_{KQ}^{(\ell')}} X^{(\ell - 1)}(\Delta W) \mathcal{S}((X^{(\ell - 1)})^\top W_{KQ}^{(\ell)} X_{(i,j)}^{(\ell - 1)}) + O(\alpha^2)
\end{aligned}
$$

$$\leq 2\beta_0 k \left\| \nabla_{W_{KQ}^{(\ell')}} X^{(\ell-1)}(\Delta W) \right\|_\infty \leq 6\beta_0 k \cdot \frac{L\epsilon \log^{\ell_0-1} \frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+6}} \|\Delta W\|_F.$$

Plug in $\Delta W = \alpha \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)}$, we have $\left\| \nabla_{W_{KQ}^{(\ell')}} \mathcal{L}^{(\ell)} \right\|_F \leq 6\beta_0 \cdot \frac{\epsilon \log^{\ell_0-1} \frac{1}{\epsilon}}{(knL)^{6L-6\ell_0+5}}$. ■

Appendix E. Learning the Value Matrix

We now show that the population gradient with respect to the value matrix vanishes at zero initialization. For simplicity, we focus on the first hop:

$$\begin{aligned}
\nabla_{W_{OV}^{(1)}} L_{\mathcal{D}}^{(1)}(\theta) &= -\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \left(\mathbf{1}\{s' = \mathrm{hop}_i^1(j)\} - \frac{1}{N} \right) \Psi_{s'} \left(\frac{1}{kN} X \mathbf{1}_{kN} \right)^\top \right] \\
&= -\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \left(\mathbf{1}\{s' = \mathrm{hop}_i^1(j)\} - \frac{1}{N} \right) \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right] \\
&= -\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \mathbf{1}\{s' = \mathrm{hop}_i^1(j)\} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right] + \mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \frac{1}{N} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right]
\end{aligned}$$

The second expectation does not depend on σ, so the expectation is

$$\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \frac{1}{N} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right] = \mathbb{E}_{(i,j)} \left[\sum_{s' \in [N]} \frac{1}{N} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right]$$

In the first term, only the indicator $\mathbf{1}\{s' = \mathrm{hop}_i^1(j)\}$ depend on σ. By symmetry, there is $\frac{1}{N}$ probability that $s' = \mathrm{hop}_i^1(j)$. Therefore, we have the expectation of this term

$$-\mathbb{E}_{\sigma,(i,j)} \left[\sum_{s' \in [N]} \mathbf{1}\{s' = \mathrm{hop}_i^1(j)\} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right] = -\mathbb{E}_{(i,j)} \left[\sum_{s' \in [N]} \frac{1}{N} \Psi_{s'} \left(\frac{1}{kN} \mathbf{1}_{kN} \right)^\top \right]$$

These two terms are identical and cancel out, showing $\nabla_{W_{OV}^{(1)}} L_{\mathcal{D}}^{(1)}(\theta) = 0$.