Transformer Efficiently Learns Low-dimensional Target Functions In-context

Anonymous Authors[1]

Abstract

Transformers can efficiently learn in-context from example demonstrations. We study ICL of a nonlinear function class via transformer with a nonlinear MLP layer: given a class of *single-index* target functions $f_*(\boldsymbol{x}) = \sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle)$, where the index features $\boldsymbol{\beta} \in \mathbb{R}^d$ are drawn from a rank-r $\ll d$ subspace, we show that a nonlinear transformer optimized by gradient descent learns f_* in-context with a prompt length that only depends on the dimension of function class r. In contrast, an algorithm that directly learns f_* on the test prompt yields a statistical complexity that scales with the ambient dimension d. Our result highlights the adaptivity of ICL to low-dimensional structures of the function class.

1. Introduction

Transformers (Vaswani et al., 2017) possess the remarkable ability of *in-context learning (ICL)* (Brown et al., 2020), whereby the model constructs a predictor from a prompt consisting of pairs of labeled examples without updating any parameters. A common explanation is that the trained transformer can implement a learning algorithm, such as gradient descent on the in-context examples, in its forward pass (Dai et al., 2022; Von Oswald et al., 2023).

Many recent theoretical works focus on learning *linear functions* using linear transformers, and it can be shown that minima of the pretraining loss implements one (preconditioned) gradient descent step on the least squares objective computed on the test prompt (Zhang et al., 2023; Ahn et al., 2023; Mahankali et al., 2023).

The motivation of our work is the observation that the simple setting of learning linear models with linear transformers does not fully capture the statistical efficiency and adaptivity of ICL. Specifically,

- A linear transformer has limited expressivity: specifically, the pretrained linear transformer cannot outperform directly solving linear regression on the test prompt. Thus, we ask the following question: *With the aid of MLP layer, can a pretrained transformer learn a nonlinear function class in-context, and outperform simple baselines such as one gradient step on the test prompt?*

- A key feature of ICL is the *adaptivity* to structure of the function class; for example, prior empirical results have shown that transformers may match the performance of ridge regression or LASSO, depending on sparsity of the pretrained task distribution (Garg et al., 2022). Such adaptivity cannot be fully explained by the one gradient step algorithm on the test prompt, which does not take into account the "prior" distribution of target functions. Hence a natural question to ask is that, "*Can a pretrained transformer adapt to certain structures of the target function class, and how does such adaptivity contribute to the statistical efficiency of ICL?*"

1.1. Our Contributions

Gaussian single-index models. To address the above questions, we study the in-context learning of the *single-index* function class, where the t-th pretraining task is constructed as $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N, \boldsymbol{x} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \boldsymbol{I}_d)$, $y_i = \sigma_*^t(\langle \boldsymbol{x}_i, \boldsymbol{\beta}_t \rangle) + \varsigma_i$, where $\sigma_*^t : \mathbb{R} \to \mathbb{R}$ is the link function, and $\boldsymbol{\beta}_t \in \mathbb{R}^d$ is the index feature vector which is randomly drawn from some fixed *rank-r subspace* for some $r \leq d$. Thus, outputs only depend on the direction of $\boldsymbol{\beta}_t$ in the d-dimensional input space (See Section 2 for details). Due to the nonlinear link function, single-index targets cannot be learned by linear transformers.

For this function class, the statistical efficiency of simple algorithms that can be implemented on the in-context examples has been extensively studied: given a link function with degree P and information exponent k (defined as the index of the smallest non-zero coefficient in the Hermite expansion of σ_*), we know that kernel methods can learn the function with $n \gtrsim d^P$ samples (Ghorbani et al., 2021; Donhauser et al., 2021), whereas two-layer neural network trained by gradient descent can achieve a better sample complexity $n \gtrsim d^{\Theta(k)}$ (Ben Arous et al., 2021; Bietti et al., 2022). These serve as a baseline for comparing the statisti-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

cal efficiency of ICL.

Moreover, in our problem setting, there is a low-dimensional structure: the subspace from which $\boldsymbol{\beta}_t$ is drawn is low-dimensional. In particular, when $r \ll d$, we expect the transformer to efficiently extract the rank-r subspace during pretraining, and hence outperform baseline algorithms that directly learn the target function from the test prompt, which cannot make use of such information.

Transformer learns single-index models in-context. We characterize the sample complexity of learning the single-index model in-context, using a transformer with a nonlinear MLP block and linear self-attention module, optimized by gradient descent.

Informally speaking, our main theorem states that the length of the test prompt N^* required to achieve sufficiently small generalization error (See (2.1) for the definition) is $r^{\Theta(P)}$, where P is the highest degree of the link function. Most importantly, it does not depend on the ambient dimension d (up to polylogarithmic term), but only the dimension r of the feature subspace. When $r \ll d$, we see a separation between ICL and algorithms that directly learn the single-index function from the test prompt such as gradient descent (where the sample complexity scales with d). This highlights the benefit of ICL in adapting to low-dimensional structures of the target function class, by featuer extraction via pretraining.

1.2. Related Works

Recent works (Zhang et al., 2023; Ahn et al., 2023; Mahankali et al., 2023; Wu et al., 2023; Zhang et al., 2024) studied the training of linear transformer to learn linear target functions in-context. Similar theoretical works are also established for transformers with SoftMax attention (Huang et al., 2023; Nichani et al., 2024; Chen et al., 2024). Our setting closely resembles (Kim & Suzuki, 2024), where a nonlinear MLP block is followed by a linear attention layer; the main difference is that we establish learnability for a concrete nonlinear function class, whereas (Kim & Suzuki, 2024) focused on global convergence of optimization. Finally, (Cheng et al., 2023) showed that transformers learn nonlinear functions in-context via a functional gradient update, but no statistical guarantees or optimization complexity were given.

The statistical and computational complexity of learning low-dimensional functions has been extensively studied. Typical target functions include single-index models (Ben Arous et al., 2021; Ba et al., 2022; Bietti et al., 2022; Mousavi-Hosseini et al., 2023; Damian et al., 2023; Ba et al., 2023) and multi-index models (Damian et al., 2022; Abbe et al., 2022; 2023; Bietti et al., 2023).

2. Problem Setting

Notations. We use boldface to represent vectors, matrices, and tensors. Let N be a nonnegative integer. Then, $[N]$ denotes the set $\{n \in \mathbb{Z} \mid 1 \le n \le N\}$. For a nonnegative integer i, the i-th Hermite polynomial is defined as $\mathrm{He}_i(z) = \mathrm{e}^{\frac{z^2}{2}} \frac{\mathrm{d}^i}{\mathrm{d}z^i} \mathrm{e}^{\frac{-z^2}{2}}$. For a set S, $\mathrm{Unif}(S)$ denotes the uniform distribution over S. We denote the unit sphere $\{\boldsymbol{x} \in \mathbb{R}^d \mid \|\boldsymbol{x}\| = 1\}$ by $\mathbb{S}^{d-1}$. $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$ represent $O(\cdot)$ and $\Omega(\cdot)$ notations where polylogarithmic terms are hidden. If necessary, we specify the targeted variables in $O, \Omega, \tilde{O}$ and $\tilde{\Omega}$, as $O_d(\cdot)$ for example. We write $a \lesssim b$ when there exists a constant c such that $a \le cb$ holds.

2.1. Data Generating Process

First, we introduce the basic setting of in-context learning (Brown et al., 2020) of simple function classes as investigated in (Garg et al., 2022; Akyürek et al., 2022). In each inference (test) task, learners are fed a sequence of inputs and outputs $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_N, y_N, \boldsymbol{x})$ referred to as *prompt*, where $\boldsymbol{x}_i, \boldsymbol{x} \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$. The labeled examples $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_N \end{pmatrix} \in \mathbb{R}^{d \times N}$, $\boldsymbol{y} = \begin{pmatrix} y_1 & \cdots & y_N \end{pmatrix}^\top \in \mathbb{R}^N$ are called *context*, and $\boldsymbol{x}$ is the *query*. We assume that the output y_i can be expressed as $y_i = f_*(\boldsymbol{x}_i) + \varsigma_i$,where f_* is the true function describing input-output relation and ς_i is label noise. The task is to predict the *response* $y = f_*(\boldsymbol{x}) + \varsigma$ corresponding to the query $\boldsymbol{x}$ given the context, without updating model parameter. We specify the distribution of inputs and outputs as follows:

Assumption 1. *The prompt* $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_N, y_N, \boldsymbol{x})$ *and the response* y *is generated as* $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N, \boldsymbol{x} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \boldsymbol{I}_d)$, $y_i = f_*(\boldsymbol{x}_i) + \varsigma_i$, $y = f_*(\boldsymbol{x}) + \varsigma$, *where* $\varsigma_i, \varsigma \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(\{-\tau, \tau\})$, *and the true function* f_* *is generated from the following distribution.*

- *Let* $\mathcal{S}$ *be an* $r \le d$*-dimensional linear subspace of* $\mathbb{R}^d$*. Draw a vector* $\boldsymbol{\beta}$ *uniformly from the unit sphere in* $\mathcal{S}$*, i.e., from* $\{\boldsymbol{\beta} \mid \boldsymbol{\beta} \in \mathcal{S}, \|\boldsymbol{\beta}\| = 1\}$.
- *Draw Hermite coefficients* $\{c_i\}_{i=2}^P$ *from a distribution satisfying* $\mathbb{E}[c_2] \neq 0$, $\sum_{i=2}^P c_i^2 = \Theta_{d,r}(1)$ $(a.s.)$*, and* $(c_2, \ldots, c_P) \neq (0, \ldots, 0)$ $(a.s.)$*. Then, we define* $f_*(\boldsymbol{x}_i) = \sigma_*(\langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle)$*, where* $\sigma_*(z) = \sum_{i=2}^P c_i \mathrm{He}_i(z)$.

Throughout the paper, we assume that $P \ll d, r$ and $r \ll d$: specifically, we take $P = O_{d,r}(1)$. This entails that the class of target functions is *low-dimensional*, and such structure should be adapted by the transformer via pretraining. As mentioned in the introduction section, the difficulty to learn a single-index model is governed by the *information exponent* of the link function σ_*: when we conduct the Hermite expansion as $\sigma_*(z) = \sum_{i \ge 0} c_i \frac{\mathrm{He}_i(z)}{i!}$, then the information exponent is defined by $\min\{i \mid c_i \neq 0\}$. In

the case of Assumption 1, this is equal to the minimal i such that $c_i \neq 0$. Therefore, our case allows that exponent changes across the tasks, and models the situation where the difficulty to learn f_* varies across tasks.

Let $f(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x})$ be an estimator for y. We evaluate the model performance by the *expected ICL risk* defined as

$$\mathcal{R}_N(f) \coloneqq \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},y}[|f(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}) - y|], \tag{2.1}$$

where the expectation is taken over prompts with length N and responses.

2.2. Student Model: transformer with Nonlinear MLP Layer

As a learning model capable of in-context learning, we consider a transformer composed of a single-layer self-attention module preceded by an embedding module using a nonlinear two-layer perceptron. Let $\boldsymbol{E} \in \mathbb{R}^{d_e \times d_N}$ be an embedding matrix constructed using a prompt $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_N, y_N, \boldsymbol{x})$. Using single-layer SoftMax-based self-attention module (Vaswani et al., 2017), the predinction of y is constructed as the right-bottom entry of $\boldsymbol{E} + \boldsymbol{W}^P \boldsymbol{W}^V \boldsymbol{E} \cdot \mathrm{softmax}\Big(\frac{(\boldsymbol{W}^K \boldsymbol{E})^\top \boldsymbol{W}^Q \boldsymbol{E}}{\rho}\Big)$, where ρ is a normalization constant and $\boldsymbol{W}^K, \boldsymbol{W}^Q \in \mathrm{R}^{d_k \times d_e}$, $\boldsymbol{W}^V \in \mathrm{R}^{d_v \times d_e}$ and $\boldsymbol{W}^P \in \mathrm{R}^{d_e \times d_v}$ are parameters called key, query, value and projection matrix, respectively. In this paper, we take $\boldsymbol{E}$ as

$$\boldsymbol{E} = \begin{bmatrix} \sigma(\boldsymbol{W}\boldsymbol{X} + \boldsymbol{b}) & \sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \\ \boldsymbol{y}^\top & 0 \end{bmatrix}. \tag{2.2}$$

Now $\sigma(\boldsymbol{W}\boldsymbol{X} + \boldsymbol{b})$ is a $m \times N$ matrix whose (i,j)-th element is $\sigma(\boldsymbol{w}_i^\top \boldsymbol{x}_j + b_i)$ and $\sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$ is a m-dimensional vector whose i-th element is $\sigma(\boldsymbol{w}_i^\top \boldsymbol{x} + b_i)$, where $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m \in \mathbb{R}^d$ and $b_1, \ldots, b_m \in \mathbb{R}$ are parameters and $\sigma : \mathbb{R} \to \mathbb{R}$ is a nonlinear activation function. In this paper, we use $\sigma(z) = \mathrm{ReLU}(z) = \max\{z, 0\}$.

In other words, we consider a two-layer neural network whose width is m, and take the output of each neuron $\sigma(\boldsymbol{w}^\top \boldsymbol{x} + b)$ at the hidden layer as the embedding. Using the output of a neural network as an embedding is adopted in some recent works (Guo et al., 2023; Kim & Suzuki, 2024). This MLP layer can extract features of the ground truth efficiently.

We further simplify the original self-attention module following the same line as (Wu et al., 2023; Zhang et al., 2023): we omit the softmax activation, set $\rho = d_N - 1 = N$, merge some parameter matrices and let some entries in the merged matrices as zero. We can show that the prediction of the output for $\boldsymbol{x}$ by the simplified transformer can be written as

$$f(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}; \boldsymbol{W}, \boldsymbol{\Gamma}, \boldsymbol{b}) = \left\langle \frac{\boldsymbol{\Gamma}\sigma(\boldsymbol{W}\boldsymbol{X} + \boldsymbol{b})\boldsymbol{y}}{N}, \sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \right\rangle \tag{2.3}$$

where $\boldsymbol{\Gamma}$ is a parameter matrix. See Appendix E for the derivation of equation (2.3). We call $\boldsymbol{\Gamma}$ the *attention matrix*.

Algorithm 1 Pretraining of transformer with MLP layer

1: **Input:** Learning rate η_1, weight decay rate λ_1, λ_2, prompt length N_1, N_2 and number of tasks T_1, T_2.
2: Draw data $\{(\boldsymbol{x}_{t,1}, y_{t,1}, \ldots, \boldsymbol{x}_{t,N_1}, y_{t,N_1}, \boldsymbol{x}_t, y_t)\}_{t=1}^{T_1}$ with prompt length N_1 and $\{(\boldsymbol{x}_{t,1}, y_{t,1}, \ldots, \boldsymbol{x}_{t,N_2}, y_{t,N_2}, \boldsymbol{x}_t, y_t)\}_{t=T_1+1}^{T_1+T_2}$ with prompt length N_2.
3: Initialize MLP weights as $\boldsymbol{w}_j^{(0)} \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ $(j \leq m/2)$ and $\boldsymbol{w}_j^{(0)} = \boldsymbol{w}_{m-j}^{(0)}$ $(j > m/2)$, biases as $b_j = 0$ $(j \in [m])$, and the attention matrix diagonally as $\Gamma_{j,j}^{(0)} \sim \mathrm{Unif}(\{\pm 1\})$ $(j \leq m/2)$ and $\Gamma_{j,j}^{(0)} = -\Gamma_{m-j,m-j}$ $(j > m/2)$.
4: $\boldsymbol{w}_j^{(1)} \leftarrow \boldsymbol{w}_j^{(0)} - \eta_1 \Big[\nabla_{\boldsymbol{w}_j} \hat{\mathcal{R}}(f) + \lambda_1 \boldsymbol{w}_j^{(0)}\Big]$
5: Re-initialize $\boldsymbol{b}$ as $b_j \sim \mathrm{Unif}([-1, 1])$ $(j \in [m])$
6: Find the minimizer $\boldsymbol{\Gamma}^*$ of $\min_{\boldsymbol{\Gamma}} \frac{1}{T_2} \sum_{t=T_1+1}^{T_1+T_2} (y_t - f(\boldsymbol{X}_t, \boldsymbol{y}_t, \boldsymbol{x}_t; \boldsymbol{W}^{(1)}, \boldsymbol{\Gamma}, \boldsymbol{b}))^2 + \lambda_2 \|\boldsymbol{\Gamma}\|_F^2$
7: **Output:** parameters $(\boldsymbol{W}^{(1)}, \boldsymbol{\Gamma}^*, \boldsymbol{b})$

2.3. Pretraining: Empirical Risk Minimization via Gradient Descent

We pretrain parameters of the transformer (2.3) by the gradient-based algorithm, written in Algorithm 1. Throughout the paper, we assume that the width m is even.

In Algorithm 1, we first conduct one-step gradient descent for regularized empirical risk $\hat{\mathcal{R}}(f) \coloneqq \frac{1}{T_1} \sum_{t=1}^{T_1} (y_t - f(\boldsymbol{X}_t, \boldsymbol{y}_t, \boldsymbol{x}_t; \boldsymbol{W}, \boldsymbol{\Gamma}, \boldsymbol{b}))^2 + \lambda_1 \|\boldsymbol{w}\|^2$ and update the MLP weight $\boldsymbol{w}$. Secondly, we conduct standard ridge regression with respect to the attention matrix $\boldsymbol{\Gamma}$. Note that the minimizer can be efficiently found because the optimization problem is convex with respect to $\boldsymbol{\Gamma}$. The symmetric initialization (line 3 in Algorithm 1) ensures that the output of the transformer is zero at initialization and removes the interaction between neurons: it is used in some recent works (Chizat et al., 2019; Damian et al., 2022).

3. Main Result: Transformer Learns Single-index Models In-context

3.1. Main Theorem

We state our main theorem to describe ICL ability of transformers. See Appendix D.3 for the proof.

Theorem 1. *Assume that the data distribution is specified as Assumption 1. Consider pretraining transformer (2.3) via Algorithm 1 with* $m \gtrsim r^P$, $T_1 = \tilde{\Omega}(d^4)$, $N_1 = \tilde{\Omega}(d^2)$, $\eta_1 = \Omega\left(\sqrt{\frac{d^3}{r}}\frac{1}{\mathrm{poly}\log d}\right)$ *and* $\lambda_1 = \eta_1^{-1}$. *Then, with probability at least 0.99 over the data distribution and the random initialization, there exists* $\lambda_2 > 0$ *such that the ICL risk with prompt length* N^* *is upper bounded as*

$$\mathcal{R}_{N^*}(f) - \tau \lesssim \frac{r^{3P/2}}{\sqrt{m}} + \mathrm{polylog}(T_2)\frac{r^{5P/2}}{\sqrt{T_2}}d\sqrt{d} + \sqrt{r^{5P}}\sqrt{\frac{r^2}{N_2} + \frac{r^2}{N^*}},$$

where $f = f(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}; \boldsymbol{W}^{(1)}, \boldsymbol{\Gamma}^*, \boldsymbol{b})$ *and dependence on* $\mathrm{poly}\log(d)$ *in the right-hand side is ignored.*

We can evaluate the sample and prompt-length complexity of Algorithm 1 using Theorem 1: to achieve $\mathcal{R}_{N^*}(f) \leq \tau + \epsilon$ for given $\epsilon > 0$, it is sufficient to set $m = \tilde{\Omega}(r^{3P})$, $T_2 = \tilde{\Omega}(r^{5P}d^3)$ and $N_2, N^* = \tilde{\Omega}(r^{5P+2})$. Most importantly, at the test time, the required prompt length N^* only depends on the inner dimension r, up to polylogarithmic terms. We emphasize that the nonlinear link function σ_* and the true direction $\boldsymbol{\beta} \in \mathcal{S}$ varies across tasks, and then the difficulty to learn the input-output relation varies. Nevertheless, Theorem 1 shows that transformers can learn the relation on a short prompt which does not scale with d.

As we have nonlinearlity in our model, empirical risk minimization problem becomes nonconvex and establishing optimization guarantee via gradient descent becomes more difficult than the linear setting. We established the guarantee by making use of one-step gradient descent, which has been considered in literatures of feature learning (Ba et al., 2022; Damian et al., 2022) and which has successfully yielded end-to-end optimization guarantee. Now we have to extend these results utilizing discussions specific to ICL as we have an attention layer.

Comparison against baseline methods. Another aspect specific to our result is that we can say transformers can outperform learning algorithms that directly act on test prompts: a lot of works (Zhang et al., 2023; Ahn et al., 2023; Mahankali et al., 2023; Wu et al., 2023; Zhang et al., 2024) discussed linear transformers, but due to the linearity of the studied transformer, ICL cannot outperform linear estimators on the test prompt.

Let us concretely compare our result with algorithms acting on the test prompt: these algorithms read each context $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_N, y_N)$ and update their parameters, then make a prediction of the response y for the query $\boldsymbol{x}$. This is simply a regression problem to estimate a single-index model $f_*(\boldsymbol{x}) = \sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle)$ using samples $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_N, y_N)$. Thus, the required prompt length is equal to the number of samples needed to learn the single-index model. Sample complexities for various algorithms to learn single-index models were shown: for linear methods such as kernel methods, $d^{\Omega(P)}$ samples are necessary to achieve ϵ-error, that is, to achieve $\mathbb{E}_{\boldsymbol{x}}[|f(\boldsymbol{x}) - y|] \leq \tau + \epsilon$ for given $\epsilon > 0$ where f is the estimator (Ghorbani et al., 2021; Donhauser et al., 2021). On the other hand, neural networks can learn single-index models with $d^{\Omega(k^*)}$ samples (Ben Arous et al., 2021; Bietti et al., 2022) by gradient descent, where k^* is the information exponent of σ_*, i.e., the minimal i such that $c_i \neq 0$. However, for the easiest case where $c_2 \neq 0$ and thus $k^* = 2$, the sample complexity is at least linear in the ambient dimension d. Moreover, a lower bound is known for a general framework of algorithms called CSQ algorithm, which includes stochastic gradient descent on neural networks: any CSQ algorithm needs $\Omega(d^{k^*/2})$ samples to achieve ϵ-error (Damian et al., 2022). Therefore, if we run these learning algorithms on each test prompt, they require the prompt length which depends on $\mathrm{poly}(d)$. Thus when $r \ll d$, pretrained transformers can capture the input-output relation in-context with a significantly shorter prompt length.

Discussion on the Mechanism. The proof of Theorem 1 is composed of several parts. First, we show that after one-step gradient descent, $\boldsymbol{w}^{(1)}$ aligns with $\mathcal{S}$, i.e., $\boldsymbol{w}^{(1)}$ is almost contained in $\mathcal{S}$ (See Appendix B for details). Secondly, we show in Appendix C that there is an attention matrix $\boldsymbol{\Gamma}$ such that the entire transformer approximates the true function well, which is crucial in the generalization error analysis (Appendix D). Thus, we can say that the MLP layer succeeds in "memorizing" the low-dimensional feature space even with the single step gradient descent, and the attention matrix works to approximate the link function correctly.

4. Conclusion and Future Direction

In this work, we studied the ICL ability of transformers and showed that they can adapt to the intrinsic low-dimensional structure of nonlinear true functions, and then outperformed algorithms working directly on the test prompt, in that the required prompt length only scaled with the inner dimension $r \ll d$.

There are several important future challenges. It is intriguing to explore the regime $r \approx d$, where memorizing the feature space $\mathcal{S}$ by pretraining is no longer meaningful. Extending our result to multi-index models is also an interesting future direction. Finally, we used a nonlinear MLP layer with a linear self-attention module, but it is important to study the ICL on nonlinear functions via nonlinear self-attention such as softmax-based self-attention modules.

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abbe, E., Adsera, E. B., and Misiakiewicz, T. 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*, pp. 4782–4887. PMLR, 2022.

Abbe, E., Adsera, E. B., and Misiakiewicz, T. SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 2552–2623. PMLR, 2023.

Ahn, K., Cheng, X., Daneshmand, H., and Sra, S. Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2023.

Akyürek, E., Schuurmans, D., Andreas, J., Ma, T., and Zhou, D. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

Ba, J., Erdogdu, M. A., Suzuki, T., Wang, Z., Wu, D., and Yang, G. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. *Advances in Neural Information Processing Systems*, 35: 37932–37946, 2022.

Ba, J., Erdogdu, M. A., Suzuki, T., Wang, Z., and Wu, D. Learning in the presence of low-dimensional structure: A spiked random matrix perspective. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HlIAoCHDWW.

Ben Arous, G., Gheissari, R., and Jagannath, A. Online stochastic gradient descent on non-convex losses from high-dimensional inference. *The Journal of Machine Learning Research*, 22(1):4788–4838, 2021.

Bietti, A., Bruna, J., Sanford, C., and Song, M. J. Learning single-index models with shallow neural networks. *Advances in Neural Information Processing Systems*, 35: 9768–9783, 2022.

Bietti, A., Bruna, J., and Pillaud-Vivien, L. On learning Gaussian multi-index models with gradient flow. *arXiv preprint arXiv:2310.19793*, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Chen, S., Sheen, H., Wang, T., and Yang, Z. Training dynamics of multi-head softmax attention for in-context learning: Emergence, convergence, and optimality. *arXiv preprint arXiv:2402.19442*, 2024.

Cheng, X., Chen, Y., and Sra, S. Transformers implement functional gradient descent to learn non-linear functions in context. *arXiv preprint arXiv:2312.06528*, 2023.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. *Advances in neural information processing systems*, 32, 2019.

Dai, D., Sun, Y., Dong, L., Hao, Y., Ma, S., Sui, Z., and Wei, F. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. *arXiv preprint arXiv:2212.10559*, 2022.

Damian, A., Lee, J., and Soltanolkotabi, M. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pp. 5413–5452. PMLR, 2022.

Damian, A., Nichani, E., Ge, R., and Lee, J. D. Smoothing the landscape boosts the signal for SGD: Optimal sample complexity for learning single index models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=73XPopmbXH.

Donhauser, K., Wu, M., and Yang, F. How rotational invariance of common kernels prevents generalization in high dimensions. In *International Conference on Machine Learning*, pp. 2804–2814. PMLR, 2021.

Garg, S., Tsipras, D., Liang, P. S., and Valiant, G. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. Linearized two-layers neural networks in high dimension. *The Annals of Statistics*, 49(2):1029–1054, 2021.

Götze, F., Sambale, H., and Sinulis, A. Concentration inequalities for polynomials in α-sub-exponential random variables. 2021.

Guo, T., Hu, W., Mei, S., Wang, H., Xiong, C., Savarese, S., and Bai, Y. How do transformers learn in-context beyond simple functions? a case study on learning with representations. *arXiv preprint arXiv:2310.10616*, 2023.

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

Kim, J. and Suzuki, T. Transformers learn nonlinear features in context: Nonconvex mean-field dynamics on the attention landscape. *arXiv preprint arXiv:2402.01258*, 2024.

Mahankali, A., Hashimoto, T. B., and Ma, T. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. *arXiv preprint arXiv:2307.03576*, 2023.

Maurer, A. A vector-contraction inequality for rademacher complexities. In *Algorithmic Learning Theory: 27th International Conference, ALT 2016, Bari, Italy, October 19-21, 2016, Proceedings 27*, pp. 3–17. Springer, 2016.

Mousavi-Hosseini, A., Park, S., Girotti, M., Mitliagkas, I., and Erdogdu, M. A. Neural networks efficiently learn low-dimensional representations with SGD. In *The Eleventh International Conference on Learning Representations*, 2023.

Nichani, E., Damian, A., and Lee, J. D. How transformers learn causal structure with gradient descent. *arXiv preprint arXiv:2402.14735*, 2024.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov, M. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

Wainwright, M. J. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

Wu, J., Zou, D., Chen, Z., Braverman, V., Gu, Q., and Bartlett, P. L. How many pretraining tasks are needed for in-context learning of linear regression? *arXiv preprint arXiv:2310.08391*, 2023.

Zhang, R., Frei, S., and Bartlett, P. L. Trained transformers learn linear models in-context. *arXiv preprint arXiv:2306.09927*, 2023.

Zhang, R., Wu, J., and Bartlett, P. L. In-context learning of a linear transformer block: Benefits of the mlp component and one-step gd initialization. *arXiv preprint arXiv:2402.14951*, 2024.

A. Preliminaries

By coordinate transformation, without loss of generality we can assume that $\mathcal{S} = \{(x_1, \dots, x_r, 0, \dots, 0) \mid x_1, \dots, x_r \in \mathbb{R}\}$ and $\boldsymbol{\beta} \sim \mathrm{Unif}(\mathbb{S}^{r-1})$, i.e., $\boldsymbol{\beta} \sim \mathrm{Unif}(\{(\beta_1, \dots, \beta_r) \mid \beta_1^2 + \cdots + \beta_r^2 = 1\})$. Therefore, we assume this in the entire proof. For a vector $\boldsymbol{w}$, we use $\boldsymbol{w}_{a:b}$ for $a \le b$ to denote the vector $[w_a, w_{a+1}, \dots, w_b]^\top$.

A.1. Definition of the Term "With High Probability"

We sometimes use the term "an event A occurs with high probability". Now we explain the definition of this.

Definition 2. *We say that an event A occurs* with high probability *when there exists a sufficiently large constant C^* which does not depend on the ambient dimension d and*

$$Pr[A] \ge 1 - \mathrm{poly}(d) d^{-C^*}$$

holds.

Note that if $A_1, \dots, A_M$ occurs with high probability where $M = \mathrm{poly}(d)$, then $A_1 \cap \cdots \cap A_M$ occurs with high probability.

A.2. Tensors

In this paper, a k-tensor is a multidimensional array which has k indices: for example, matrices are 2-tensors. Let A be a k-tensor. $A_{i_1,\dots,i_k}$ denotes $(i_1, \dots, i_k)$-th entry of A. Let A be a k-tensor and B be a l-tensor where $k \ge l$. $A(B)$ denotes $k - l$ tensor whose $(i_1, \dots, i_{k-l})$-th entry is

$$A(B)_{i_1,\dots,i_{k-l}} = \sum_{j_1,\dots,j_l} A_{i_1,\dots,i_{k-l},j_1,\dots,j_l} B_{j_1,\dots,j_l},$$

and is defined only when sizes are compatible. If $k = l$, we sometimes write $A(B)$ as $A \circ B$. Let $\boldsymbol{v} \in \mathbb{R}^d$ be a vector and k be a positive integer. Then, $\boldsymbol{v}^{\otimes k} \in \mathbb{R}^{d \times \cdots \times d}$ denotes a k-tensor whose $(i_1, \dots, i_k)$-th entry is $v_{i_1} \cdots v_{i_k}$.

Let $f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ be a d-variable function. A k-tensor $\nabla^k f(\boldsymbol{x})$ is defined as

$$\left(\nabla^k f(\boldsymbol{x})\right)_{i_1,\dots,i_k} = \frac{\partial}{\partial x_{i_1}} \cdots \frac{\partial}{\partial x_{i_k}} f(\boldsymbol{x}).$$

A.3. Hermite Polynomials

We frequently use (probablists') Hermite polynomials, which is defined by $\mathrm{He}_i(z) = \mathrm{e}^{\frac{z^2}{2}} \frac{\mathrm{d}^i}{\mathrm{d}z^i} \mathrm{e}^{\frac{-z^2}{2}}$, where i is a non-negative integer. Hermite polynomials have orthogonality, in that $\mathbb{E}_{z \sim \mathcal{N}(0,1)}[\mathrm{He}_i(z)\mathrm{He}_j(z)] = i! \delta_{i,j}$. The Hermite expansion for $\sigma : \mathbb{R} \to \mathbb{R}$ is defined as $\sigma(z) = \sum_{i \ge 0} \frac{a_i}{i!} \mathrm{He}_i(z)$ where $a_i = \mathbb{E}_{z \sim \mathcal{N}(0,1)}[\sigma(z)\mathrm{He}_i(z)]$. Similarly, the multivariate Hermite expansion for $f : \mathbb{R}^d \to \mathbb{R}$ is defined as $f(\boldsymbol{z}) = \sum_{i_1 \ge 0, \dots, i_d \ge 0} \frac{a_{i_1,\dots,i_d}}{(i_1)! \cdots (i_d)!} \mathrm{He}_{i_1}(z_1) \cdots \mathrm{He}_{i_d}(z_d)$, where $a_{i_1,\dots,i_d} = \mathbb{E}_{z_i,\dots,z_d \sim \mathcal{N}(0,1)}[f(\boldsymbol{z})\mathrm{He}_{i_1}(z_1) \cdots \mathrm{He}_{i_d}(z_d)]$. The coefficient $a_{i_1,\dots,i_d}$ can also be obtained by $a_{i_1,\dots,i_d} = \mathbb{E}_{z_i,\dots,z_d \sim \mathcal{N}(0,1)}\left[\frac{\partial^{i_1}}{\partial z_1^{i_1}} \cdots \frac{\partial^{i_d}}{\partial z_d^{i_d}} f(\boldsymbol{z})\right]$.

The lemma below is useful to find a basis of the set of true functions.

Lemma 3. *Suppose $\boldsymbol{\beta} \in \mathcal{S}$. Then,*

$$\mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle) = \sum_{p_1 \ge 0, \dots, p_r \ge 0}^{p_1 + \cdots + p_r = p} \frac{(p_1 + \cdots + p_r)!}{p_1! \cdots p_r!} \cdot \beta_1^{p_1} \cdots \beta_r^{p_r} \cdot \mathrm{He}_{p_1}(x_1) \cdots \mathrm{He}_{p_r}(x_r)$$

holds.

Proof. Note that $\mathbb{E}_{z_i,\dots,z_r \sim \mathcal{N}(0,1)}\left[\frac{\partial^{i_1}}{\partial z_1^{i_1}} \cdots \frac{\partial^{i_r}}{\partial z_r^{i_r}} \mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle)\right]$ is nonzero only when $i_1 + \cdots + i_r = p$. When $i_1 + \cdots + i_r = p$, then $\mathbb{E}_{z_1,\dots,z_r \sim \mathcal{N}(0,1)}\left[\frac{\partial^{i_1}}{\partial z_1^{i_1}} \cdots \frac{\partial^{i_r}}{\partial z_r^{i_r}} \mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle)\right] = p! \beta_1^{i_1} \cdots \beta_r^{i_r}$ holds. Then, from the multivariate Hermite expansion we obtain the claim. □

B. Proofs for MLP Layer

First, note that $f(\boldsymbol{X}_t, \boldsymbol{y}_t, \boldsymbol{x}_t; \boldsymbol{W}^{(0)}, \boldsymbol{\Gamma}^{(0)}, \boldsymbol{b}) = 0$ at initialization. Then, from line 4 of Algorithm 1, for each $j \in [m]$,

$$\begin{aligned}\boldsymbol{w}_j^{(1)} &= 2\eta_1 \frac{1}{T}\sum_{t=1}^{T} y_t \nabla_{\boldsymbol{w}_j^{(0)}} f(\boldsymbol{X}_t, \boldsymbol{y}_t, \boldsymbol{x}_t; \boldsymbol{W}^{(0)}, \boldsymbol{\Gamma}^{(0)}, \boldsymbol{b}) \\ &= 2\eta_1 \Gamma_{j,j}^{(0)} \Bigg(\sum_{t=1}^{T} \frac{1}{T} y_t \sigma({\boldsymbol{w}_j^{(0)}}^\top \boldsymbol{x}_t) \frac{1}{N}\sum_{i=1}^{N} y_{t,i} \sigma'({\boldsymbol{w}_j^{(0)}}^\top \boldsymbol{x}_{t,i}) \boldsymbol{x}_{t,i} \\ &\quad + \sum_{t=1}^{T} \frac{1}{T} y_t \sigma'({\boldsymbol{w}_j^{(0)}}^\top \boldsymbol{x}_t) \boldsymbol{x}_t \frac{1}{N}\sum_{i=1}^{N} y_{t,i} \sigma({\boldsymbol{w}_j^{(0)}}^\top \boldsymbol{x}_{t,i}) \Bigg)\end{aligned}$$

holds because we assumed $\eta_1 = \lambda_1^{-1}$ (for simplicity let $T \coloneqq T_1$ and $N \coloneqq N_1$ in this section). Here $\boldsymbol{x}_{t,i}$ is the i-th column of $\boldsymbol{X}_t$ and $y_{t,i}$ is the i-th element of $\boldsymbol{y}_t$. Now let

$$\begin{aligned}&\boldsymbol{g}_T(\boldsymbol{w}) \\ &\coloneqq \left(\sum_{i=1}^{T} \frac{1}{T} y_t \sigma(\boldsymbol{w}^\top \boldsymbol{x}_t) \frac{1}{N}\sum_{i=1}^{N} y_{t,i} \sigma'(\boldsymbol{w}^\top \boldsymbol{x}_{t,i}) \boldsymbol{x}_{t,i} + \sum_{i=1}^{T} \frac{1}{T} y_t \sigma'(\boldsymbol{w}^\top \boldsymbol{x}_t) \boldsymbol{x}_t \frac{1}{N}\sum_{i=1}^{N} y_{t,i} \sigma(\boldsymbol{w}^\top \boldsymbol{x}_{t,i}) \right)\end{aligned}$$

and $\boldsymbol{g}(\boldsymbol{w}) = \mathbb{E}[\boldsymbol{g}_T(\boldsymbol{w})]$, where the expectation is taken with respect to the data distribution. Note that $\boldsymbol{w}_j^{(1)} = 2\eta_1 \Gamma_{j,j}^{(0)} \boldsymbol{g}_T(\boldsymbol{w}_j^{(0)})$.

In this section we make

- asymptotic expansion of $\boldsymbol{g}(\boldsymbol{w})$, and
- uniform upper bound for the difference $\|\boldsymbol{g}(\boldsymbol{w}) - \boldsymbol{g}_T(\boldsymbol{w})\|$.

Asymptotic Expansion of $\boldsymbol{g}(\boldsymbol{w})$. First, note that

$$\boldsymbol{g}(\boldsymbol{w}) = 2\mathbb{E}_y[\mathbb{E}_{\boldsymbol{x}}[y\sigma'(\boldsymbol{w}^\top \boldsymbol{x})\boldsymbol{x}]\mathbb{E}_{\boldsymbol{x}}[y\sigma(\boldsymbol{w}^\top \boldsymbol{x})]],$$

where $\mathbb{E}_y[\cdot]$ means the expectation with respect to the distribution of $\boldsymbol{\beta}$ and $\{c_i\}$.

Now let $\sigma(z) = \sum_{i\geq 0} a_i \frac{\mathrm{He}_i(z)}{i!}$ be the Hermite expansion of student activation. The asymptotic expansions of $\mathbb{E}_{\boldsymbol{x}}[y\sigma'(\boldsymbol{w}^\top \boldsymbol{x})\boldsymbol{x}]$ and $\mathbb{E}_{\boldsymbol{x}}[y\sigma(\boldsymbol{w}^\top \boldsymbol{x})]$ are known as follows:

Lemma 4. *It holds that*

$$\begin{aligned}\mathbb{E}_{\boldsymbol{x}}[y\sigma'(\boldsymbol{w}^\top \boldsymbol{x})\boldsymbol{x}] &= \sum_{k=1}^{P} \frac{a_{k+1}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k+1} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!} + \boldsymbol{w}\sum_{k=2}^{P} \frac{a_{k+2}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!}, \\ \mathbb{E}_{\boldsymbol{x}}[y\sigma(\boldsymbol{w}^\top \boldsymbol{x})] &= \sum_{k=2}^{P} \frac{a_{k}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!}.\end{aligned}$$

Proof. It is obtained from the proof of Lemma 7 in (Damian et al., 2022). □

Then, $\boldsymbol{g}(\boldsymbol{w})$ can be expanded as

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{w}) = 2\mathbb{E}_y \Bigg[\Bigg(&\sum_{k=1}^{p} \frac{a_{k+1}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k+1} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!} \\ &+ \boldsymbol{w}\sum_{k=2}^{p} \frac{a_{k+2}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!} \Bigg) \Bigg(\sum_{k=2}^{p} \frac{a_{k}\mathbb{E}_{\boldsymbol{x}}[\nabla^{k} f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes k})}{k!} \Bigg) \Bigg]\end{aligned}$$

$$=: 2\mathbb{E}_y\left[a_2\mathbb{E}_{\boldsymbol{x}}[\nabla^2 f_*(\boldsymbol{x})](\boldsymbol{w})\frac{a_2\mathbb{E}_{\boldsymbol{x}}[\nabla^2 f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes 2})}{2} + \boldsymbol{s}(y,\boldsymbol{w})\right]$$
$$= a_2^2\mathbb{E}_y[\mathbb{E}_{\boldsymbol{x}}[\nabla^2 f_*(\boldsymbol{x})](\boldsymbol{w})\mathbb{E}_{\boldsymbol{x}}[\nabla^2 f_*(\boldsymbol{x})](\boldsymbol{w}^{\otimes 2})] + 2\mathbb{E}_y[\boldsymbol{s}(y,\boldsymbol{w})].$$

The main term is proportional to $\mathbb{E}_y\big[(\boldsymbol{H}_{f_*}\boldsymbol{w})(\boldsymbol{H}_{f_*}\circ\boldsymbol{w}^{\otimes 2})\big]$, where

$$\boldsymbol{H}_{f_*} = \mathbb{E}_{\boldsymbol{x}}[\nabla^2 f_*(\boldsymbol{x})] = 2c_2\boldsymbol{\beta}\boldsymbol{\beta}^\top$$

be the expected Hessian of f_*. Let us calculate this main term explicitly. Recall that we assumed that $\boldsymbol{\beta}\sim\mathrm{Unif}(\mathbb{S}^{r-1})$. Note that by letting $\boldsymbol{\beta}'\sim\mathcal{N}(0,\boldsymbol{\Sigma}_{\boldsymbol{\beta}})$ where

$$\boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \begin{pmatrix}\boldsymbol{I}_r & 0\\ 0^\top & \boldsymbol{O}_{d-r}\end{pmatrix},$$

and $z\sim\chi_r$ independent from $\boldsymbol{\beta}$, then $\boldsymbol{\beta}'\sim\boldsymbol{\beta}z$ holds.

Now it holds that

$$\mathbb{E}_y\big[(\boldsymbol{H}_{f_*}\boldsymbol{w})(\boldsymbol{H}_{f_*}\circ\boldsymbol{w}^{\otimes 2})\big] = \mathbb{E}_{\boldsymbol{\beta},c_2}\Big[4c_2^2(\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{w})(\boldsymbol{\beta}\boldsymbol{\beta}^\top\circ\boldsymbol{w}^{\otimes 2})\Big]$$
$$= 4\mathbb{E}[c_2^2]\mathbb{E}_{\boldsymbol{\beta}}\big[\boldsymbol{\beta}^{\otimes 4}\big]\boldsymbol{w}^{\otimes 3}.$$

Furthermore, as

$$\mathbb{E}_{\boldsymbol{\beta}}[\boldsymbol{\beta}_i\boldsymbol{\beta}_j\boldsymbol{\beta}_k\boldsymbol{\beta}_l] = \begin{cases}\dfrac{3}{\mathbb{E}_{z\sim\chi_r}[z^4]}\ (i=j=k=l\le r)\\ \dfrac{1}{\mathbb{E}_{z\sim\chi_r}[z^4]}\ (i=j\le r, k=l\le r, i\ne k \text{ or } i=k\le r, j=l\le r, i\ne j,\\ \qquad\qquad \text{or } i=l\le r, j=k\le r, i\ne j)\\ 0\ (\text{otherwise})\end{cases}$$

it follows that

$$\big(\mathbb{E}[c_2^2]\mathbb{E}_{\boldsymbol{\beta}}\big[\boldsymbol{\beta}^{\otimes 4}\big]\boldsymbol{w}^{\otimes 3}\big)_i = \begin{cases}\mathbb{E}[c_2^2]\dfrac{3}{r(r+2)}w_i\displaystyle\sum_{j=1}^r w_j^2\ (i\le r)\\ 0\ (i>r)\end{cases}.$$

Then, we arrive at

$$\mathbb{E}_y\big[(\boldsymbol{H}_{f_*}\boldsymbol{w})(\boldsymbol{H}_{f_*}\circ\boldsymbol{w}^{\otimes 2})\big] = \mathbb{E}[c_2^2]\frac{12}{r(r+2)}\|\boldsymbol{w}_{1:r}\|_2^2\boldsymbol{w}_{1:r}.$$

Next, we upper bound the residual term $\mathbb{E}_y[\boldsymbol{s}(y,\boldsymbol{w})]$.

Lemma 5.

$$\sup_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\|\mathbb{E}_y[\boldsymbol{s}(y,\boldsymbol{w})]\| = O\left(\sqrt{\frac{r^2}{d^4}}\right)$$

holds.

Proof. Note that

$$\sup_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\|\mathbb{E}_y[\boldsymbol{s}(y,\boldsymbol{w})]\| \le \sup_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\mathbb{E}_y[\|\boldsymbol{s}(y,\boldsymbol{w})\|^2]^{1/2}.$$

Then, by Minkowski's inequality, it suffices to show that

$$\mathbb{E}_y\left[\left\|\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^2\right]^{1/2} \lesssim \sqrt{\frac{r^2}{d^4}}\ (k\geq 1, l\geq 2, (k,l)\neq(1,2))$$

and

$$\mathbb{E}_y\left[\left\|\boldsymbol{w}\frac{a_{k+2}\mathbb{E}[\nabla^{k}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^2\right]^{1/2} \lesssim \sqrt{\frac{r^2}{d^4}}\ (k\geq 2, l\geq 2).$$

The former is obtained by

$$\begin{aligned}
&\mathbb{E}_y\left[\left\|\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^2\right]^{1/2}\\
\leq&\mathbb{E}_y\left[\left\|\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\right\|^4\right]^{1/4}\left[\left\|\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^4\right]^{1/4}\\
\lesssim&\sqrt{\frac{r^{\lfloor k/2\rfloor}}{d^k}}\sqrt{\frac{r^{\lfloor l/2\rfloor}}{d^l}}.
\end{aligned}$$

Now, for deriving the last line, we used Corollary 9 and Lemma 24 in (Damian et al., 2022). The latter can be derived by following the same line. □

Bounding the Difference between Empirical and Population Gradient. We upper bound $\|\boldsymbol{g}_T(\boldsymbol{w})-\boldsymbol{g}(\boldsymbol{w})\|$ by extending Lemma 19 in (Damian et al., 2022). In the paper they bound the difference between empirical and population gradient of a two-layer fully-connected neural network. However, in our case we have an attention module and nonlinear activation appears twice in the gradient. This yields the need for using concentration argument in a "nested" way.

Lemma 6. *Let*

$$\boldsymbol{z}_1(\boldsymbol{w}) = \sum_{t=1}^{T}\frac{1}{T}y_t\sigma(\boldsymbol{w}^\top\boldsymbol{x}_t)\frac{1}{N}\sum_{i=1}^{N}y_{t,i}\sigma'(\boldsymbol{w}^\top\boldsymbol{x}_{t,i})\boldsymbol{x}_{t,i}$$

and

$$\boldsymbol{z}_2(\boldsymbol{w}) = \sum_{t=1}^{T}\frac{1}{T}y_t\sigma'(\boldsymbol{w}^\top\boldsymbol{x}_t)\boldsymbol{x}_t\frac{1}{N}\sum_{i=1}^{N}y_{t,i}\sigma(\boldsymbol{w}^\top\boldsymbol{x}_{t,i}).$$

Then, with high probability over the data distribution,

$$\begin{aligned}
\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}}\|\boldsymbol{z}_1(\boldsymbol{w})-\mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})]\| &= \tilde{O}\left(\left(\sqrt{\frac{1}{d}}+\sqrt{\frac{d}{N}}\right)\sqrt{\frac{d}{T}}\right),\\
\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}}\|\boldsymbol{z}_2(\boldsymbol{w})-\mathbb{E}[\boldsymbol{z}_2(\boldsymbol{w})]\| &= \tilde{O}\left(\left(\frac{1}{d}+\sqrt{\frac{d}{N}}\right)\sqrt{\frac{d}{T}}\right)
\end{aligned}$$

holds.

Proof. First, consider $\boldsymbol{z}_1(\boldsymbol{w})$. Let $\iota = C\log N$, where C is a sufficiently large constant. From Lemma 19 in (Damian et al., 2022), with probability at least $1-2N\mathrm{e}^{-\iota}$ it holds that

$$\frac{1}{N}\sum_{i=1}^{N}y_{t,i}\sigma'(\boldsymbol{w}^\top\boldsymbol{x}_{t,i})\boldsymbol{x}_{t,i} \leq \mathbb{E}[y_{t,i}\sigma'(\boldsymbol{w}^\top\boldsymbol{x}_{t,i})\boldsymbol{x}_{t,i}] + C_t\sqrt{\frac{d\iota^{P+1}}{N}} \text{ for all } w\in\mathbb{S}^{d-1} \tag{B.1}$$

with fixed t, where C_t is a constant which only depends on t and P. Let R_t be the right hand side of (B.1). Note that $R_t = \tilde{O}\left(\sqrt{\frac{1}{d}}+\sqrt{\frac{d\iota^{P+1}}{N}}\right)$ with high probability. Moreover, from Lemma 17 in (Damian et al., 2022), $y_t \leq (C_t'\iota)^{P/2} =: R_t'$ holds with high probability. In the following, we assume that

- (B.1) holds,
- $y_t \le R'_t$ holds with all t, and
- $\|\boldsymbol{x}_{t,i}\|, \|\boldsymbol{x}_t\| \le a\sqrt{d}$ holds for all t and i, where a is a sufficiently large constant.

which occurs with high probability. Let

$$\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) = \frac{1}{T}\sum_{i=1}^{T} y_t \sigma(\boldsymbol{w}^\top \boldsymbol{x}_t)\boldsymbol{v}_t(\boldsymbol{w})\mathbf{1}[\boldsymbol{v}_t(\boldsymbol{w}) \le R_t \wedge y_t \le R'_t],$$

where $\boldsymbol{v}_t(\boldsymbol{w}) \coloneqq \frac{1}{N}\sum_{i=1}^{N} y_{t,i}\sigma'(\boldsymbol{w}^\top \boldsymbol{x}_{t,i})\boldsymbol{x}_{t,i}$. From the assumption above, $\tilde{\boldsymbol{z}}_1 = \boldsymbol{z}_1$ holds uniformly over all $\boldsymbol{w} \in \mathbb{S}^{d-1}$. Let us bound $\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}} \|\boldsymbol{z}_1(\boldsymbol{w}) - \mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})]\|$ following the same line as the proof of Lemma 19 in (Damian et al., 2022). As

$$\boldsymbol{z}_1(\boldsymbol{w}) - \mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})] = (\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]) - (\mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})] - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]),$$

it suffices to bound $\sup_{\boldsymbol{w}} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\|$ and $\sup_{\boldsymbol{w}} \|\mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})] - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\|$. First,

$$\begin{aligned}
&\sup_{\boldsymbol{w}} \|\mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})] - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| \\
&= \sup_{\boldsymbol{w}} \big\|\mathbb{E}\big[y_t\sigma(\boldsymbol{w}^\top \boldsymbol{x}_t)\boldsymbol{v}_t(\boldsymbol{w})\mathbf{1}[\boldsymbol{v}_t(\boldsymbol{w}) > R_t \vee y_t > R'_t]\big]\big\| \\
&\le \sup_{\boldsymbol{w}} \sqrt{\mathbb{E}[\|\boldsymbol{v}_t(\boldsymbol{w})\|^2]} \cdot \sup_{\boldsymbol{w}} \sqrt{\mathbb{E}[y_t^2\sigma(\boldsymbol{w}^\top \boldsymbol{x}_t)^2\mathbf{1}[\boldsymbol{v}_t(\boldsymbol{w}) > R_t \vee y_t > R'_t]^2]} \\
&\le \sup_{\boldsymbol{w}} \mathbb{E}\big[\|\boldsymbol{v}_t(\boldsymbol{w})\|^2\big]^{1/2} \cdot \sup_{\boldsymbol{w}} \mathbb{E}[y_t^4]^{1/4}\mathbb{E}[\sigma(\boldsymbol{w}^\top \boldsymbol{x}_t)^8]^{1/8}\mathbb{P}[\boldsymbol{v}_t(\boldsymbol{w}) > R_t \vee y_t > R'_t]^{1/8}.
\end{aligned}$$

Then, as $\sup_{\boldsymbol{w}} \mathbb{E}\big[\|\boldsymbol{v}_t(\boldsymbol{w})\|^2\big]^{1/2}, \sup_{\boldsymbol{w}} \mathbb{E}[y_t^4]^{1/4}$ and $\sup_{\boldsymbol{w}} \mathbb{E}[\sigma(\boldsymbol{w}^\top \boldsymbol{x}_t)^8]^{1/8}$ are polynomial in d, by setting C sufficiently large, it follows that $\sup_{\boldsymbol{w}} \|\mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})] - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| = O(d^{-C^*})$ for any $C^* \ge 1$ with high probability.

Second, let us bound $\sup_{\boldsymbol{w}} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\|$. From Lemmas 18 and 19 in (Damian et al., 2022), there exists ϵ-covering $\mathcal{N}_\epsilon$ of $\mathbb{S}^{d-1}$ with $|\mathcal{N}_\epsilon| \le \mathrm{e}^{C_1 d\log(NT/\epsilon)}$ and $1/4$-covering $\mathcal{N}_{1/4}$ of $\mathbb{S}^{d-1}$ with $|\mathcal{N}_{1/4}| \le \mathrm{e}^{C_2 d}$ such that for all $\boldsymbol{w} \in \mathbb{S}^{d-1}$, there exists $\boldsymbol{\pi}(\boldsymbol{w}) \in \mathcal{N}_\epsilon$ such that $\|\boldsymbol{w} - \boldsymbol{\pi}(\boldsymbol{w})\| \le \epsilon$ and $\boldsymbol{v}_t(\boldsymbol{w}) = \boldsymbol{v}_t(\boldsymbol{\pi}(\boldsymbol{w}))$ holds for all t. Then

$$\begin{aligned}
&\sup_{\boldsymbol{w}} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| \\
&\le \sup_{\boldsymbol{w}\in\mathcal{N}_\epsilon} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| + \sup_{\boldsymbol{w}} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \tilde{\boldsymbol{z}}_1(\boldsymbol{\pi}(\boldsymbol{w}))\| + \sup_{\boldsymbol{w}} \|\mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})] - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{\pi}(\boldsymbol{w}))]\| \\
&\le \sup_{\boldsymbol{w}\in\mathcal{N}_\epsilon} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| + \sup_{\boldsymbol{w}} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \tilde{\boldsymbol{z}}_1(\boldsymbol{\pi}(\boldsymbol{w}))\| + O(d^{-C^*} + \epsilon d) \\
&\le \sup_{\boldsymbol{w}\in\mathcal{N}_\epsilon} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| + O(RR'\epsilon\sqrt{d}) + O(d^{-C^*} + \epsilon d)
\end{aligned}$$

Let us bound the first term in the right-hand side. Notice that

$$\sup_{\boldsymbol{w}\in\mathcal{N}_\epsilon} \|\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]\| \le 2 \sup_{\boldsymbol{w}\in\mathcal{N}_\epsilon} \sup_{\boldsymbol{u}\in\mathcal{N}_{1/4}} \boldsymbol{u}\cdot[\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]].$$

Now, $\tilde{\boldsymbol{z}}_1(\boldsymbol{w})$ is RR'-sub Gaussian and then with probability $1 - 2\mathrm{e}^{-z}$, it holds that

$$\boldsymbol{u}\cdot[\tilde{\boldsymbol{z}}_1(\boldsymbol{w}) - \mathbb{E}[\tilde{\boldsymbol{z}}_1(\boldsymbol{w})]] \le RR'\sqrt{2z/T},$$

where $R = \max_t R_t$ and $R' = \max_t R'_t$. Therefore, with probability at least $1 - 2\mathrm{e}^{C_3 d\log(NT/\epsilon) - z}$ we have

$$\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}} \|\boldsymbol{z}_1(\boldsymbol{w}) - \mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})]\| = O\left(RR'\sqrt{\frac{2z}{T}} + RR'\epsilon\sqrt{d} + \epsilon d + d^{-C^*}\right).$$

By taking $z = C_3 d \log(NT/\epsilon) + \iota$ and $\epsilon = O(d^{-C^*})$, we arrive at

$$\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}} \|\boldsymbol{z}_1(\boldsymbol{w}) - \mathbb{E}[\boldsymbol{z}_1(\boldsymbol{w})]\| = O\left(\iota^{P/2}\left(\sqrt{\frac{1}{d}} + \sqrt{\frac{d\iota^{P+1}}{N}}\right)\sqrt{\frac{2z}{T}} + \epsilon d + O(d^{-C^*})\right)$$
$$= \tilde{O}\left(\left(\sqrt{\frac{1}{d}} + \sqrt{\frac{d}{N}}\right)\sqrt{\frac{d}{T}}\right)$$

with high probability.

Bounds for $\sup_{\boldsymbol{w}\in\mathbb{S}^{d-1}} \|\boldsymbol{z}_2(\boldsymbol{w}) - \mathbb{E}[\boldsymbol{z}_2(\boldsymbol{w})]\|$ can be obtained by the same procedure, noting that $\mathbb{E}[y_{t,i}\sigma(\boldsymbol{w}^\top \boldsymbol{x}_{t,i})] = O(1/d)$. □

Combining Lemmas 5 and 6, we immediately obtain the following:

Corollary 7. *Asuume $T = \tilde{\Omega}(d^4)$ and $N = \tilde{\Omega}(d^2)$. Then, $\|\boldsymbol{g}_T(w)\| = \tilde{O}(\sqrt{\frac{r}{d^3}})$ and $\|\boldsymbol{g}_T(w)_{r+1:d}\| = \tilde{O}(\sqrt{\frac{r^2}{d^4}})$ holds with high probability.*

It is used to derive prompt length-free generalization result (Appendix D.2).

C. Construction of Attention Matrix

In this section, we construct $\boldsymbol{\Gamma}$ which satisfies the following approximation property:

Theorem 8. *With high probability (see Section A.1 for the definition), there exists $\boldsymbol{\Gamma}$ such that*

$$\left|\left\langle \frac{\boldsymbol{\Gamma}\sigma(\boldsymbol{W}^{(1)}\boldsymbol{X}^t + \boldsymbol{b})\boldsymbol{y}}{\dim(\boldsymbol{y}^t)}, \begin{bmatrix} \sigma({\boldsymbol{w}_1^{(1)}}^\top \boldsymbol{x}^t + b_1) \\ \vdots \\ \sigma({\boldsymbol{w}_m^{(1)}}^\top \boldsymbol{x}^t + b_m) \end{bmatrix} \right\rangle - y^t \right| - \tau \lesssim \mathrm{poly}\log(d)\left(\frac{r^{5P/2}}{\sqrt{N_2}} + \frac{r^{3P/2}}{\sqrt{m}}\right)$$

for all $t \in \{T_1+1, \dots, T_2\}$, where $\{\boldsymbol{w}_j^{(1)}\}_j$ are neurons obtained by line 4 of Algorithm 1 with $\eta_1 = \Omega\left(\sqrt{\frac{d^3}{r}}\frac{1}{\mathrm{poly}\log d}\right), m \gtrsim r^P$ and $\boldsymbol{b}$ is re-initialized in Line 5 of Algorithm 1. Moreover, $\|\boldsymbol{\Gamma}\| = \tilde{O}(\sqrt{r^{5P}/m^2})$ is satisfied.

We give an intuitive explanation of the way to construct such $\boldsymbol{\Gamma}$. First, recall that the true function satisfies $f_*(\boldsymbol{x}) = \sum_{i=2}^P c_i \mathrm{He}_i(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)$. Let $\mathcal{F}_* = \{f_* : \sum_{i=2}^P c_i \mathrm{He}_i(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle) \mid \boldsymbol{\beta} \in \mathcal{S}, c_i \in \mathbb{R}\}$. We can find an orthogonal basis of $\mathcal{F}_*$: let $\{\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_r\}$ be an orthonormal basis of $\mathcal{S}$. Then, we can show that $\mathcal{G} = \{\prod_{j=1}^r \mathrm{He}_{p_j}(\langle \boldsymbol{\beta}_j, \cdot\rangle) \mid 2 \le p_1 + \cdots + p_r \le P, p_1 \ge 0, \dots, p_r \ge 0\}$ forms a basis for $\mathcal{F}_*$ and $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(0,I_d)}[g(\boldsymbol{x})g'(\boldsymbol{x})] = 0$ is satisfied if $g \neq g'$.

Let $q = |\mathcal{G}| (= O(r^P))$. Assign numbers to the functions in $\mathcal{G}$ arbitrarily as $\mathcal{G} = \{g_1, \dots, g_q\}$. We can show that there exists $\boldsymbol{a}^1, \dots, \boldsymbol{a}^q \in \mathbb{R}^m$ such that $\sum_{j=1}^m \boldsymbol{a}_j^i \sigma(\boldsymbol{w}_j^\top \boldsymbol{x} + b_j) \simeq g_i(\boldsymbol{x})$ for each $i \in [q]$. This comes from the approximation property of two-layer fully-connected neural networks.

A key fact is that we can construct attention matrix using this $\boldsymbol{a}^1, \dots, \boldsymbol{a}^q$. Let $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{a}^1 & \cdots & \boldsymbol{a}^q \end{pmatrix} \in \mathbb{R}^{m\times q}$ and $\boldsymbol{D} = \mathrm{diag}\{\alpha_1, \dots, \alpha_q\}$ where $\alpha_i = \mathbb{E}[g_i(\boldsymbol{x})^2]^{-1}$. Then, letting $\dim(\boldsymbol{y}) = N$ and $\boldsymbol{\Gamma} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^\top$, we notice that

$$\left\langle \frac{\boldsymbol{\Gamma}\sigma(W\boldsymbol{X} + \boldsymbol{b})\boldsymbol{y}}{\dim(\boldsymbol{y})}, \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\rangle$$
$$= \sum_{i=1}^q \alpha_i \left(\frac{1}{N}\sum_{j=1}^N \left(\sum_{k=1}^m \boldsymbol{a}_k^i \sigma(\boldsymbol{w}_k^\top \boldsymbol{x}_j + b_k)\right) y_j\right)\left(\sum_{k=1}^m \boldsymbol{a}_k^i \sigma(\boldsymbol{w}_k^\top \boldsymbol{x} + b_k)\right)$$
$$\simeq \sum_{i=1}^q \alpha_i \mathbb{E}[g_i(\boldsymbol{x})\sigma_*(\langle \boldsymbol{\beta}, \boldsymbol{x}\rangle)]g_i(\boldsymbol{x}) = \sigma_*(\langle \boldsymbol{\beta}, \boldsymbol{x}\rangle). \tag{C.1}$$

The most essential point is that the self-attention architecture approximately calculates $\mathbb{E}[g_i(\boldsymbol{x})\sigma_*(\langle\boldsymbol{\beta},\boldsymbol{x}\rangle)]$, i.e., the inner product between the true function and a base g_i. If N is sufficiently large, then we can approximate the true label well. Recall that we defined $\mathcal{G} = \{\prod_{j=1}^r \mathrm{He}_{p_j}(\langle\boldsymbol{\beta}_j,\cdot\rangle) \mid 2 \le p_1+\cdots+p_r \le P, p_1 \ge 0,\dots,p_r \ge 0\}$ where $\{\boldsymbol{\beta}_j\}$ is a basis of $\mathcal{S}$ and this forms a basis of the set of true functions. We let $|\mathcal{G}| = q$ and number the elements of $\mathcal{G}$ arbitrarily as $\mathcal{G} = \{g_1,\dots,g_q\}$. Importantly, $q \lesssim r^P$ holds.

Note that, to derive (C.1), there are two types of "approximation error": one is the error by a two-layer neural network $\sum_{j=1}^m \boldsymbol{a}_j^i \sigma(\boldsymbol{w}_j^\top \boldsymbol{x} + b_j)$ to approximate each basis $g_i(\boldsymbol{x})$. The other is discrepancy between average $\frac{1}{N}\sum_{j=1}^N g_i(\boldsymbol{x}_j)y_j$ and its expectation $\mathbb{E}[g_i(\boldsymbol{x})\sigma_*(\langle\boldsymbol{\beta},\boldsymbol{x}\rangle)]$. In Appendix C.1 we bound the former error, and the latter is evaluated in Appendix C.2. Finally, we prove Theorem 8 in Appendix C.3.

C.1. Approximation Result for Two-layer NN

We use the following proposition stating approximation ability of two-layer fully-connected neural networks:

Proposition 9. *Let $\{\boldsymbol{w}_j^{(1)}\}_j$ be neurons obtained by line 4 of Algorithm 1 with $m \gtrsim r^P, \eta_1 = \Omega\left(\sqrt{\frac{d^3}{r}}\frac{1}{\mathrm{poly}\log d}\right)$ and assume $\boldsymbol{b}$ is initialized as in Line 5 of Algorithm 1. Then, with high probability over the data distribution, there exists $\boldsymbol{a}^1,\dots,\boldsymbol{a}^q \in \mathbb{R}^m$ such that for each $i \in [q]$,*

$$\left(\sum_{j=1}^m \boldsymbol{a}_j^i \sigma({\boldsymbol{w}_j^{(1)}}^\top \boldsymbol{x} + b_j) - g_i(\boldsymbol{x})\right)^2 = \tilde{O}\left(\frac{r^P}{m} + \frac{1}{N_2}\right)$$

holds for all $\boldsymbol{x} = \boldsymbol{x}_t$ ($t \in \{T_1+1,\dots,T_2\}$) and $\boldsymbol{x} = \boldsymbol{x}_{t,n}$ ($t \in \{T_1+1,\dots,T_2\}, n \in [N_2]$). Moreover, $\|\boldsymbol{a}^i\|^2 = \tilde{O}\left(\frac{r^P}{m}\right)$ ($i \in [q]$) holds.

This is almost a counterpart of Lemma 13 in (Damian et al., 2022) and the proof is almost the same. However, there are slight differences as the property of $\boldsymbol{w}_j^{(1)}$ differs: specifically, the statement of Lemmas 10 and 11 in (Damian et al., 2022) to derive Lemma 13 in the paper should be slightly changed. Here we describe the counterparts of these lemmas.

Lemma 10 (counterpart of Lemma 10 in (Damian et al., 2022))**.** *Let $\boldsymbol{r}(w) = \boldsymbol{g}_T(w) - a_2^2\mathbb{E}[c_2^2]\frac{12}{r(r+2)}\|\boldsymbol{w}_{1:r}\|_2^2\boldsymbol{w}_{1:r}$. Then,*

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\|\Pi_{\mathcal{S}}\boldsymbol{r}(w)\|^j\right]^{1/j} \lesssim \tilde{O}\left(\left(\sqrt{\frac{1}{d}}+\sqrt{\frac{d}{N_1}}\right)\sqrt{\frac{d}{T_1}}\right) + O\left(\sqrt{\frac{r^2}{d^4}}\right)$$

holds with high probability, where $\Pi_{\mathcal{S}}$ be the orthogonal projection onto $\mathcal{S}$ (recall that we assumed that $\mathcal{S}$ be the subspace spanned by first r standard basis: that is, $\Pi_{\mathcal{S}}\boldsymbol{v} = \boldsymbol{v}_{1:r}$).

Proof. Since $\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\|\Pi_{\mathcal{S}}[\boldsymbol{g}_T(w) - \boldsymbol{g}(w)]\|^j\right]^{1/j} = \tilde{O}\left(\left(\sqrt{\frac{1}{d}}+\sqrt{\frac{d}{N_1}}\right)\sqrt{\frac{d}{T_1}}\right)$ from Lemma 6, it suffices to upper bound

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\Pi_{\mathcal{S}}[\boldsymbol{g}(w) - a_2^2\mathbb{E}[c_2^2]\frac{12}{r(r+2)}\|\boldsymbol{w}_{1:r}\|_2^2\boldsymbol{w}_{1:r}\right\|^j\right]^{1/j} = \mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}[\|\mathbb{E}_y[\Pi_{\mathcal{S}}\boldsymbol{s}(y,\boldsymbol{w})]\|^j]^{1/j}.$$

Then it is sufficient to show that

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\mathbb{E}_y\left[\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right]\right\|^j\right]^{1/j}$$
$$\lesssim \sqrt{\frac{r^2}{d^4}}\ (k \ge 1, l \ge 2, (k,l) \ne (1,2))$$

and

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\mathbb{E}_y\left[\Pi_{\mathcal{S}}\boldsymbol{w}\frac{a_{k+2}\mathbb{E}[\nabla^k f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right]\right\|^j\right]^{1/j} \lesssim \sqrt{\frac{r^2}{d^4}}\ (k \ge 2, l \ge 2).$$

First,

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\mathbb{E}_y\left[\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right]\right\|^j\right]^{1/j}\\
\leq&\mathbb{E}_y\left[\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^j\right]\right]^{1/j}\\
\leq&\mathbb{E}_y\left[\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\frac{a_{k+1}\mathbb{E}[\nabla^{k+1}f^t](\boldsymbol{w}^{\otimes k})}{k!}\right\|^{2j}\right]^{1/2}\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^{2j}\right]^{1/2}\right]^{1/j}\\
\lesssim&\sqrt{\frac{r^{\lfloor k/2\rfloor}}{d^k}}\sqrt{\frac{r^{\lfloor l/2\rfloor}}{d^l}}\lesssim\sqrt{\frac{r^2}{d^4}}.
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\mathbb{E}_y\left[\Pi_{\mathcal{S}}\boldsymbol{w}\frac{a_{k+2}\mathbb{E}[\nabla^k f^t](\boldsymbol{w}^{\otimes k})}{k!}\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right]\right\|^j\right]^{1/j}\\
\leq&\mathbb{E}_y\left[\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\|\Pi_{\mathcal{S}}\boldsymbol{w}\|^{2j}\right]^{1/2}\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\frac{a_{k+2}\mathbb{E}[\nabla^k f^t](\boldsymbol{w}^{\otimes k})}{k!}\right\|^{4j}\right]^{1/4}\right.\\
&\left.\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\left\|\frac{a_l\mathbb{E}[\nabla^l f^t](\boldsymbol{w}^{\otimes l})}{l!}\right\|^{4j}\right]^{1/4}\right]^{1/j}\\
\lesssim&\sqrt{\frac{r}{d}}\sqrt{\frac{r^{\lfloor k/2\rfloor}}{d^k}}\sqrt{\frac{r^{\lfloor l/2\rfloor}}{d^l}}\lesssim\sqrt{\frac{r^2}{d^4}}.
\end{aligned}$$

Then we obtain $\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}\left[\|\Pi_{\mathcal{S}}\boldsymbol{r}(w)\|^j\right]^{1/j}\lesssim\tilde{O}\Big(\Big(\sqrt{\frac{1}{d}}+\sqrt{\frac{d}{N_1}}\Big)\sqrt{\frac{d}{T_1}}\Big)+O\Big(\sqrt{\frac{r^2}{d^4}}\Big)$ as desired. □

To derive the counterpart of Lemma 11 in (Damian et al., 2022) we need the following statement:

Lemma 11 (Tensor expectation lower bound)**.** *Let $\boldsymbol{T}$ be a $k < P$-symmetric tensor which has support on $\mathcal{S}$ and let $\bar{\boldsymbol{w}} = \boldsymbol{w}_{1:r}\|\boldsymbol{w}_{1:r}\|^2$. Then*

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}[\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k})^2]\gtrsim\frac{r^{2i}}{d^{3i}}\mathbb{E}[\|\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k-i})\|_F^2].$$

Proof. *Let $\boldsymbol{u}\sim\mathcal{N}(0,I_d)$, $z\sim\chi(d)$ and $\bar{\boldsymbol{u}} = \boldsymbol{u}_{1:r}\|\boldsymbol{u}_{1:r}\|^2$. Then we can decompose as $\bar{\boldsymbol{u}} = z^3\bar{\boldsymbol{w}}$. Therefore*

$$\mathbb{E}[\boldsymbol{T}(\bar{\boldsymbol{u}}^{\otimes k})^2] = \mathbb{E}[z^{6k}]\mathbb{E}[\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k})^2]$$

holds.

On the other hand, let $\boldsymbol{x}\sim\mathbb{S}^{r-1}$ and $z'\sim\chi(r)$. Then we can decompose as $\bar{\boldsymbol{u}} = (z')^3\boldsymbol{x}$. Thus

$$\mathbb{E}[\boldsymbol{T}(\bar{\boldsymbol{u}}^{\otimes k})^2] = \mathbb{E}[(z')^{6k}]\mathbb{E}[\boldsymbol{T}(\boldsymbol{x}^{\otimes k})^2]$$

holds. It implies that

$$\mathbb{E}[\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k})^2] = \frac{\mathbb{E}[(z')^{6k}]}{\mathbb{E}[z^{6k}]}\mathbb{E}[\boldsymbol{T}(\boldsymbol{x}^{\otimes k})^2].$$

Similarly,

$$\mathbb{E}[\|\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k-i})\|_F^2] = \frac{\mathbb{E}[(z')^{6(k-i)}]}{\mathbb{E}[z^{6(k-i)}]}\mathbb{E}[\|\boldsymbol{T}(\boldsymbol{x}^{\otimes(k-i)})\|_F^2]$$

is satisfied. Now from Corollary 13 in (Damian et al., 2022),

$$\mathbb{E}[\|\boldsymbol{T}(\boldsymbol{x}^{\otimes(k-i)})\|_F^2]\lesssim r^i\mathbb{E}[\boldsymbol{T}(\boldsymbol{x}^{\otimes k})^2]$$

holds. Therefore we obtain

$$\begin{aligned}\mathbb{E}[\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k})^2] &= \frac{\mathbb{E}[(z')^{6k}]}{\mathbb{E}[z^{6k}]}\mathbb{E}[\boldsymbol{T}(\boldsymbol{x}^{\otimes k})^2] \\ &\gtrsim r^{-i}\frac{\mathbb{E}[(z')^{6k}]}{\mathbb{E}[z^{6k}]}\mathbb{E}[\|\boldsymbol{T}(\boldsymbol{x}^{\otimes (k-i)})\|_F^2] \\ &= r^{-i}\frac{\mathbb{E}[(z')^{6k}]}{\mathbb{E}[z^{6k}]}\frac{\mathbb{E}[z^{6(k-i)}]}{\mathbb{E}[(z')^{6(k-i)}]}\mathbb{E}[\|\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k-i})\|_F^2] \\ &= \frac{r^{2i}}{d^{3i}}\mathbb{E}[\|\boldsymbol{T}(\bar{\boldsymbol{w}}^{\otimes k-i})\|_F^2].\end{aligned}$$

□

Corollary 12. *Let $\boldsymbol{T}$ be a $k < P$-symmetric tensor which has support on $\mathcal{S}$ and let $\hat{\boldsymbol{w}} = \frac{1}{r(r+2)}\boldsymbol{w}_{1:r}\|\boldsymbol{w}_{1:r}\|^2$. Then*

$$\mathbb{E}_{\boldsymbol{w}\sim\mathbb{S}^{d-1}}[\boldsymbol{T}(\hat{\boldsymbol{w}}^{\otimes k})^2] \gtrsim \frac{1}{d^{3i}}\mathbb{E}[\|\boldsymbol{T}(\hat{\boldsymbol{w}}^{\otimes k-i})\|_F^2].$$

Using Lemma 10 and Corollary 12 in the proof of Lemma 11 in (Damian et al., 2022) yields the following Lemma:

Lemma 13 (counterpart of Lemma 11 in (Damian et al., 2022))**.** *Suppose $r^2 \lesssim d$, $T_1 \gtrsim d^3$ and $N_1 \gtrsim d^2$. Let $\boldsymbol{T}$ be $k < P$-symmetric tensor with $\|\boldsymbol{T}\|_F = 1$ and assume $\boldsymbol{T}$ has support on $\mathcal{S}$. Then,*

$$\mathbb{E}_{\boldsymbol{w}\sim\mathrm{Unif}\mathbb{S}^{d-1}}[(\Pi_{\mathcal{S}}\boldsymbol{g}_T(w))^{\otimes 2k}](\boldsymbol{T},\boldsymbol{T}) \gtrsim d^{-3k}$$

holds.

Now we have Lemmas 10 and 13. By letting $\eta = \Omega\left(\sqrt{\frac{d^3}{r}}\frac{1}{\mathrm{poly}\log d}\right)$ and following the same line towards the proof of Lemma 13 in (Damian et al., 2022), we obtain Proposition 9.

C.2. Concentration of Correlation between a Label and a Base

In this subsection, we give an upper bound for

$$\left|\frac{1}{N}\sum_{j=1}^{N} y_j g(\boldsymbol{x}_j) - \mathbb{E}[yg(\boldsymbol{x})]\right|$$

as follows:

Proposition 14. *Let $g \in \mathcal{G}$. With high probability,*

$$\left|\frac{1}{N}\sum_{j=1}^{N} y_j g(\boldsymbol{x}_j) - \mathbb{E}[yg(\boldsymbol{x})]\right| \lesssim \frac{r^{3P/2}}{\sqrt{N}}(\log d)^{P/2}$$

holds.

Without loss of generality, we assume that $\mathcal{S} = \{(\beta_1,\dots,\beta_r,0,\dots,0) \mid \beta_1,\dots,\beta_r \in \mathbb{R}\}$ and $g(\boldsymbol{x})$ can be written in the form as $g(\boldsymbol{x}) = \mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r)$, satisfying $q_1+\cdots+q_r \le P$. Note that

$$\frac{1}{N}\sum_{j=1}^{N} y_j g(\boldsymbol{x}_j) - \mathbb{E}[yg(\boldsymbol{x})] = \frac{1}{N}\sum_{j=1}^{N}\varsigma_j g(\boldsymbol{x}_j) + \frac{1}{N}\sum_{j=1}^{N}\sigma_*(\langle\boldsymbol{x}_j,\boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j) - \mathbb{E}[\sigma_*(\langle\boldsymbol{x},\boldsymbol{\beta}\rangle)g(\boldsymbol{x})],$$

where $\sigma_*(z) = \sum_{i=2}^{P} c_p\mathrm{He}_i(z)$. First we give an upper bound for $\frac{1}{N}\sum_{j=1}^{N}\sigma_*(\langle\boldsymbol{x}_j,\boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j) - \mathbb{E}[\sigma_*(\langle\boldsymbol{x},\boldsymbol{\beta}\rangle)g(\boldsymbol{x})]$.

Lemma 15. *Let* $D = \deg \frac{1}{N}\sum_{j=1}^N \sigma_*(\langle \boldsymbol{x}_j, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j) \le 2P$. *For all* $t > 0$,

$$\mathbb{P}(|\frac{1}{N}\sum_{j=1}^N \sigma_*(\langle \boldsymbol{x}_j, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j) - \mathbb{E}[\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x})]| \ge t)$$
$$\le 2\exp\left(-\frac{1}{C_D}\min_{1\le k\le D}\left(\frac{t\sqrt{N}}{M^k\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F}\right)^{2/k}\right)$$

holds, where C_D *and* M *are absolute constants.*

Proof. Let $F(\boldsymbol{x}_1, \dots, \boldsymbol{x}_N) = \frac{1}{N}\sum_{j=1}^N \sigma_*(\langle \boldsymbol{x}_j, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j)$. It is a polynomial for $x_{11}, \dots, x_{1d}, \dots, x_{Nd}$, which are standard Gaussian variables. For the standard Gaussian $X \sim \mathcal{N}(0,1)$, its Orlicz norm $\|X\|_{\Psi_2}$ is bounded; there exists M such that $\|X\|_{\Psi_2} \le M$. Moreover,

$$\|\mathbb{E}[\nabla^k F(\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)]\|_F = \sqrt{N}^{-1}\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F$$

is satisfied. Then, Theorem 1.2 in (Götze et al., 2021) yields the lemma. □

Then, our goal is to bound $\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F$.

Lemma 16.

$$\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F = O(r^{3P})$$

holds.

Proof. First,

$$\begin{aligned}
&\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2 \\
=&\|\sum_{p=2}^P c_p\mathbb{E}[\nabla^k(\mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2 \\
\le&\sum_{p=2}^P c_p\|\mathbb{E}[\nabla^k(\mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2 \\
\le&(c_2^2 + \cdots + c_P^2)\left(\sum_{p=2}^P \|\mathbb{E}[\nabla^k(\mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2\right) \\
\lesssim&\sum_{p=2}^P \|\mathbb{E}[\nabla^k(\mathrm{He}_p(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2 \\
\le&\sum_{p=2}^P\left(\sum_{p_1+\cdots+p_r=p}^{p_1,\dots,p_r\ge 0}\left(\frac{(p_1+\cdots+p_r)!}{p_1!\cdots p_r!}\cdot\beta_1^{p_1}\cdots\beta_r^{p_r}\right)^2\right) \\
&\cdot\sum_{p_1+\cdots+p_r=p}^{p_1,\dots,p_r\ge 0}\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2
\end{aligned}$$

holds. Noting that $\left(\frac{(p_1+\cdots+p_r)!}{p_1!\cdots p_r!}\cdot\beta_1^{p_1}\cdots\beta_r^{p_r}\right)^2 \le p!$ under $p_1+\cdots+p_r = p$ and the number of the combination of $(p_1, \dots, p_r) \ge 0$ such that $p_1+\cdots+p_r = p$ is at most r^p, there exists a constant C_P depending only on P such that

$$\|\mathbb{E}[\nabla^k(\sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2 \le C_P r^P\sum_{p=2}^P\sum_{p_1+\cdots+p_r=p}^{p_1,\dots,p_r\ge 0}\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2. \tag{C.2}$$

Next, let us bound $\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2$, where $g(\boldsymbol{x})=\mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r)$. It holds that

$$\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))=\sum_{i=0}^{k}\binom{k}{i}\nabla^i\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)\nabla^{k-i}\mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r). \tag{C.3}$$

Let $s=|p_1-q_1|+\cdots+|p_r-q_r|$ and $t=\sum_{i=1}^r|p_i-q_i|\mathbf{1}_{p_i>q_i}$. Let us first consider the case where $k=s$ to simplify the explanation; in this case, each element of $\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]$ can be decomposed into the terms as

$$\mathbb{E}[\partial_{x_1}^{u_1}\cdots\partial_{x_r}^{u_r}\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)\partial_{x_1}^{v_1}\cdots\partial_{x_r}^{v_r}\mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r)], \tag{C.4}$$

where $u_1+\cdots+u_r+v_1+\cdots+v_r=s$. However, this expectation becomes nonzero only when $u_i=|p_i-q_i|\mathbf{1}_{p_i>q_i}$ and $v_i=|p_i-q_i|\mathbf{1}_{p_i<q_i}$. Therefore, by seeing (C.3), we can notice that

$$\mathbb{E}[\nabla^i\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)\nabla^{k-i}\mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r)] \tag{C.5}$$

is nonzero tensor only when $i=t$ and $\mathbb{E}[\nabla^t\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)\nabla^{k-t}\mathrm{He}_{q_1}(x_1)\cdots\mathrm{He}_{q_r}(x_r)]$ has only one nonzero element. It implies that $\|\mathbb{E}[\nabla^s(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2\lesssim\binom{s}{t}^2$.

Let us consider the other cases: obviously, $\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2=0$ if $k<s$. Moreover, if $k-s$ is odd, one can confirm that $\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2=0$. Consider the case where $k-s=2l>0$. The expectation (C.4) under $u_1+\cdots+u_r+v_1+\cdots+v_r=l=s+2l$ is nonzero when $u_i=|p_i-q_i|\mathbf{1}_{p_i>q_i}+l_i$ and $v_i=|p_i-q_i|\mathbf{1}_{p_i<q_i}+l_i$, where $l_1+\cdots+l_r=l$. It implies that (C.5) is nonzero only when $i=t+l$, and in this case, the tensor (C.5) has at most r^l nonzero entries. As a consequence,

$$\|\mathbb{E}[\nabla^s(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2\lesssim r^l\binom{s+l}{t+l}^2$$

holds. Overall, $\|\mathbb{E}[\nabla^k(\mathrm{He}_{p_1}(x_1)\cdots\mathrm{He}_{p_r}(x_r)g(\boldsymbol{x}))]\|_F^2\le C_P'r^P$ is satisfied where C_P' depends only on P. Plugging this bound into (C.2), we arrive at

$$\|\mathbb{E}[\nabla^k(\sigma_*(\langle\boldsymbol{x},\boldsymbol{\beta}\rangle)g(\boldsymbol{x}))]\|_F^2\le C_PC_P'r^P\sum_{p=2}^{P}\sum_{p_1\ge0,\ldots,p_r\ge0}^{p_1+\cdots+p_r=p}r^P\lesssim r^{3P},$$

as desired. □

Plugging the result above and $t=\Theta(\frac{r^{3P/2}}{\sqrt{N}}(\log d)^{P/2})$ into Lemma 15 yields the following corollary.

Corollary 17. *With high probability,*

$$\left|\frac{1}{N}\sum_{j=1}^{N}\sigma_*(\langle\boldsymbol{x}_j,\boldsymbol{\beta}\rangle)g(\boldsymbol{x}_j)-\mathbb{E}[\sigma_*(\langle\boldsymbol{x},\boldsymbol{\beta}\rangle)g(\boldsymbol{x})]\right|\lesssim\frac{r^{3P/2}}{\sqrt{N}}(\log d)^{P/2}$$

holds.

Proof. [Proof of Proposition 14] Now we have Corollary 17, then it remains to show that $\frac{1}{N}\sum_{j=1}^N\varsigma_jg(\boldsymbol{x}_j)\lesssim\frac{r^{3P/2}}{\sqrt{N}}(\log d)^{P/2}$ with high probability. This is obvious from Lemma 18. □

C.3. Proof of Theorem 8

We prove Theorem 8 using the preparations above. It suffices to show the theorem in the case where $T=1$, as long as T is polynomial in d. Then, we drop the subscript t.

Note that, from Lemma 3, we can expand $f_*(\boldsymbol{x})$ as $f_*(\boldsymbol{x})=\sum_{i=1}^q\alpha_i\mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(0,I_d)}[yg_i(\boldsymbol{x})]g_i(\boldsymbol{x})$, where $\alpha_i=\mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(0,I_d)}[g_i(\boldsymbol{x})^2]^{-1}$. Note that $\alpha_i=O_{d,r}(1)$ and $\alpha_i^{-1}=O_{d,r}(1)$. Now let $\boldsymbol{A}=\begin{pmatrix}\boldsymbol{a}^1&\cdots&\boldsymbol{a}^q\end{pmatrix}\in\mathbb{R}^{m\times q}$ and

$\boldsymbol{D} = \mathrm{diag}\{\alpha_1, \dots, \alpha_q\}$. First,

$$
\begin{aligned}
&\left\langle \frac{\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^\top \sigma(\boldsymbol{W}\boldsymbol{X} + \boldsymbol{b})\boldsymbol{y}}{\dim(\boldsymbol{y})}, \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\rangle - y \\
&= \sum_{i=1}^{q} \alpha_i (g_i(\boldsymbol{x}) + \epsilon_i(\boldsymbol{x})) \left(\mathbb{E}[y g_i(\boldsymbol{x})] + \left(\frac{1}{N_2} \sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[y g_i(\boldsymbol{x})] \right) + \frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j) \right) \\
&\quad - \sigma_*(\langle \boldsymbol{x}, \boldsymbol{\beta} \rangle) - \varsigma \\
&= \sum_{i=1}^{q} \alpha_i (g_i(\boldsymbol{x}) + \epsilon_i(\boldsymbol{x})) \left(\mathbb{E}[y g_i(\boldsymbol{x})] + \left(\frac{1}{N_2} \sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[y g_i(\boldsymbol{x})] \right) + \frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j) \right) \\
&\quad - \sum_{i=1}^{q} g_i(\boldsymbol{x}) \alpha_i \mathbb{E}[y g_i(\boldsymbol{x})] - \varsigma
\end{aligned}
$$

holds, where $\epsilon_i(\boldsymbol{x}) = \sum_{j=1}^{m} \boldsymbol{a}_j^i \sigma({\boldsymbol{w}_j^{(1)}}^\top \boldsymbol{x} + b_j) - g_i(\boldsymbol{x})$.

Then, it follows that

$$
\begin{aligned}
&\left| \left\langle \frac{\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^\top \sigma(W\boldsymbol{X} + \boldsymbol{b})\boldsymbol{y}}{\dim(\boldsymbol{y})}, \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\rangle - y \right| \\
&\le \sum_{i=1}^{q} \alpha_i |\epsilon_i(\boldsymbol{x}) \mathbb{E}[y g_i(\boldsymbol{x})]| + \sum_{i=1}^{q} \alpha_i \left| g_i(\boldsymbol{x}) \left(\frac{1}{N_2} \sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[y g_i(\boldsymbol{x})] \right) + g_i(\boldsymbol{x}) \frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j) \right| \\
&+ \sum_{i=1}^{q} \alpha_i \left| \epsilon_i(\boldsymbol{x}) \left(\frac{1}{N_2} \sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[y g_i(\boldsymbol{x})] \right) + \epsilon_i(\boldsymbol{x}) \frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j) \right| + \tau.
\end{aligned}
$$

To bound these terms, we need to bound $g_i(\boldsymbol{x})$ and $\frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j)$, whose expectations are zero.

Lemma 18. *Let $s = \deg g_i$. Then*

$$
|g_i(\boldsymbol{x})| \lesssim (\log d)^{s/2}
$$

with high probability for each $i \in [q]$.

Proof. Note that $\mathbb{E}[\nabla^d g_i(\boldsymbol{x})]$ has only one nonzero element only when $d = s$. Then, from Theorem 1.2 in (Götze et al., 2021),

$$
\mathbb{P}(|g_i(\boldsymbol{x})| \ge t) \le 2 \exp\left(-\frac{1}{C} \left(\frac{t}{M^s} \right)^{2/s} \right)
$$

holds. Plugging $t = \Omega((\log d)^{s/2})$ yields the result. □

Lemma 19.

$$
\left| \frac{1}{N_2} \sum_{j=1}^{N_2} y_j \epsilon_i(\boldsymbol{x}_j) \right| \lesssim \frac{r^{P/2} (\log d)^{P/2}}{\sqrt{m}} + \frac{(\log d)^{P/2}}{\sqrt{N_2}}
$$

with high probability.

Proof. From Lemma 17 in (Damian et al., 2022), $\sigma_*(\langle \boldsymbol{\beta}, \boldsymbol{x}_j \rangle) \lesssim (\log d)^{P/2}$ holds with high probability. Then, the lemma follows immediately from Proposition 9. □

In addition to the lemmas above, we know that $\mathbb{E}[yg_i(\boldsymbol{x})] = O(1)$ from Lemma 3 and $q = O(r^P)$. Note that as we assumed $m \gtrsim r^P$, $\epsilon_i(\boldsymbol{x}) = O(\mathrm{poly}\log(d))$ holds. Then, we arrive at

$$\begin{aligned}
&\left|\left\langle \frac{\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^\top\sigma(W\boldsymbol{X}+\boldsymbol{b})\boldsymbol{y}}{\dim(\boldsymbol{y})}, \begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}+b_1)\\ \vdots \\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}+b_m)\end{bmatrix}\right\rangle - y\right| \\
\le& \sum_{i=1}^q |\epsilon_i(\boldsymbol{x})\mathbb{E}[yg_i(\boldsymbol{x})]| + \sum_{i=1}^q \left| g_i(\boldsymbol{x})\left(\frac{1}{N_2}\sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[yg_i(\boldsymbol{x})]\right) + g_i(\boldsymbol{x})\frac{1}{N_2}\sum_{j=1}^{N_2} y_j\epsilon_i(\boldsymbol{x}_j)\right| \\
&+ \sum_{i=1}^q \left|\epsilon_i(\boldsymbol{x})\left(\frac{1}{N_2}\sum_{j=1}^{N_2} y_j g_i(\boldsymbol{x}_j) - \mathbb{E}[yg_i(\boldsymbol{x})]\right) + \epsilon_i(\boldsymbol{x})\frac{1}{N_2}\sum_{j=1}^{N_2} y_j\epsilon_i(\boldsymbol{x}_j)\right| + \tau \\
\lesssim& r^P\left(\frac{r^{P/2}}{\sqrt{m}} + \frac{1}{\sqrt{N_2}}\right) + \mathrm{poly}\log(d)\left(r^P\cdot\frac{r^{3P/2}}{\sqrt{N_2}} + r^P\cdot\left(\frac{r^{P/2}}{\sqrt{m}} + \frac{1}{\sqrt{N_2}}\right)\right) + \tau \\
\lesssim& \mathrm{poly}\log(d)\left(\frac{r^{5P/2}}{\sqrt{N_2}} + \frac{r^{3P/2}}{\sqrt{m}}\right) + \tau.
\end{aligned}$$

For $\|\boldsymbol{\Gamma}\|$, we obtain

$$\begin{aligned}
\|\boldsymbol{\Gamma}\| &\le \|\boldsymbol{A}\|\|\boldsymbol{D}\|\|\boldsymbol{A}\| \\
&\le \sum_{i=1}^q \|\boldsymbol{a}^i\|^2\cdot\sqrt{q} \\
&= \tilde{O}\left(\sqrt{r^{5P}/m^2}\right).
\end{aligned}$$

D. Generalization Error Analysis and Proof of Theorem 1

D.1. Rademacher Complexity Bound

Let

$$\mathcal{F}_{N,G,W,B} = \left\{(\boldsymbol{X},\boldsymbol{y},\boldsymbol{x}) \mapsto \left|\left\langle \frac{\boldsymbol{\Gamma}\sigma(\boldsymbol{W}\boldsymbol{X}+\boldsymbol{b})\boldsymbol{y}}{N}, \begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}+b_1)\\ \vdots \\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}+b_m)\end{bmatrix}\right\rangle\right| \right.$$
$$\left. \|\boldsymbol{\Gamma}\|_F \le G, \|\boldsymbol{w}_1\|_2,\cdots,\|\boldsymbol{w}_m\|_2 \le W, |b_1|\cdots|b_m| \le B\right\}$$

be the set of transformers whose parameter norms are constrained, and let

$$\mathrm{Rad}_T(\mathcal{F}_{N,G,W,B}) = \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{f\in\mathcal{F}}\frac{1}{T}\sum_{t=1}^T \epsilon^t f(\boldsymbol{X}^t,\boldsymbol{y}^t,\boldsymbol{x}^t)\right]$$

be its Rademacher complexity, where $\epsilon_i \sim \mathrm{Unif}(\{\pm 1\})$.

We evaluate the Rademacher complexity as follows:

Proposition 20.

$$\mathrm{Rad}_T(\mathcal{F}_{N,G,W,B}) = O\left(\mathrm{polylog}(T)\frac{mG}{\sqrt{T}}\left(r^{P/2}\sqrt{d} + d\sqrt{d}\right)\right).$$

holds, when $B = O_{d,r}(1)$ and $W = O_{d,r}(1)$.

Proof.

First,

$$
\begin{aligned}
&\mathrm{Rad}_T(\mathcal{F}_{N,G,W,B}) \\
&= \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{f\in\mathcal{F}} \frac{1}{T}\sum_{t=1}^{T}\epsilon^t f(\boldsymbol{X}^t,\boldsymbol{y}^t,\boldsymbol{x}^t)\right] \\
&= \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b}} \frac{1}{T}\sum_{t=1}^{T}\epsilon^t\left\langle \frac{\boldsymbol{\Gamma}\sigma(\boldsymbol{W}\boldsymbol{X}^t+\boldsymbol{b})\boldsymbol{y}^t}{N}, \begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}^t+b_1)\\ \vdots \\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}^t+b_m)\end{bmatrix}\right\rangle\right] \\
&\le \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b}} \frac{1}{T}\|\boldsymbol{\Gamma}\|_F\left\|\sum_{t=1}^{T}\epsilon^t\frac{\sigma(\boldsymbol{W}\boldsymbol{X}^t+\boldsymbol{b})\boldsymbol{y}^t}{N}\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}^t+b_1)\\ \vdots \\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}^t+b_m)\end{bmatrix}^\top\right\|_F\right] \\
&\le \frac{Gm}{T}\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{\boldsymbol{W},\boldsymbol{b}}\left\|\sum_{t=1}^{T}\epsilon^t\frac{\sigma(\boldsymbol{W}\boldsymbol{X}^t+\boldsymbol{b})\boldsymbol{y}^t}{N}\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}^t+b_1)\\ \vdots \\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}^t+b_m)\end{bmatrix}^\top\right\|_\infty\right] \\
&\le \frac{Gm}{TN}\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\left|\sum_{t=1}^{T}\epsilon^t\sum_{i=1}^{N}\sigma(\boldsymbol{w}^\top\boldsymbol{x}_i^t+b)y_i^t\sigma({\boldsymbol{w}'}^\top\boldsymbol{x}^t+b')\right|\right] \\
&\le \frac{Gm}{TN}\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\epsilon^t\sum_{i=1}^{N}\sigma(\boldsymbol{w}^\top\boldsymbol{x}_i^t+b)y_i^t\sigma({\boldsymbol{w}'}^\top\boldsymbol{x}^t+b') \\
&+ \sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} -\sum_{t=1}^{T}\epsilon^t\sum_{i=1}^{N}\sigma(\boldsymbol{w}^\top\boldsymbol{x}_i^t+b)y_i^t\sigma({\boldsymbol{w}'}^\top\boldsymbol{x}^t+b')] \\
&\le \frac{2Gm}{TN}\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\epsilon^t\sum_{i=1}^{N}\sigma(\boldsymbol{w}^\top\boldsymbol{x}_i^t+b)y_i^t\sigma({\boldsymbol{w}'}^\top\boldsymbol{x}^t+b')\right] \\
&\le \frac{2Gm}{T}\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\epsilon^t\sigma(\boldsymbol{w}^\top\boldsymbol{x}^t+b)y^t\sigma({\boldsymbol{w}'}^\top{\boldsymbol{x}'}^t+b')\right]. \\
&\le \frac{2Gm}{T}\mathbb{E}_{R,R'}\left[\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\epsilon^t\sigma(\boldsymbol{w}^\top\boldsymbol{x}^t+b)y^t\sigma({\boldsymbol{w}'}^\top{\boldsymbol{x}'}^t+b')\middle| A_{R,R'}\right]\right],
\end{aligned}
$$

where

$$A_{R,R'} \coloneqq \{\max\{\|\boldsymbol{x}_{1:r}^t\|^2, \|{\boldsymbol{x}'}_{1:r}^t\|^2\}_{t=1}^T = R^2, \max\{\|\boldsymbol{x}_{r+1:d}^t\|^2, \|{\boldsymbol{x}'}_{r+1:d}^t\|^2\}_{t=1}^T = {R'}^2\}.$$

We utilize multivariate contraction inequality (Maurer, 2016) to bound the last line.

Lemma 21. *Let $f(x,y,z) = \sigma(x)y\sigma(z)$, whose domain is restricted to $|x| \le R_1, |y| \le R_2, |z| \le R_3$. Then, f is $\sqrt{R_1^2R_2^2 + R_2^2R_3^2 + R_3^2R_1^2} \le R_1R_2 + R_2R_3 + R_3R_1$-Lipschitz continuous.*

Lemma 22. *There exists a polynomial g such that*

$$
\begin{aligned}
&\frac{2Gm}{T}\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\epsilon^t\sigma(\boldsymbol{w}^\top\boldsymbol{x}^t+b)y^t\sigma({\boldsymbol{w}'}^\top{\boldsymbol{x}'}^t+b')\middle| A_{R,R'}\right] \\
\le &\frac{2\sqrt{2}GmL}{T}\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'}\sum_{t=1}^{T}\Big(\epsilon^{t,1}(\boldsymbol{w}^\top\boldsymbol{x}^t+b)+\epsilon^{t,2}y^t+\epsilon^{t,3}({\boldsymbol{w}'}^\top{\boldsymbol{x}'}^t+b')\Big)\middle| A_{R,R'}\right],
\end{aligned}
$$

where $L = 2(W(R+R')+B)(g(R)+\tau) + (W(R+R')+B)^2$. Moreover, $g(z)$ is at most of degree P, increasing in the region $z \ge 0$, and its coefficient is $O(\sqrt{\sum_i c_i^2})$.

Proof. First, as $\|\boldsymbol{x}^t\| \leq \sqrt{R^2 + (R')^2}$, $|\boldsymbol{w}^\top \boldsymbol{x}^t + b| \leq W\sqrt{R^2 + (R')^2} + B \leq W(R + R') + B$ holds. Secondly,

$$\begin{aligned}|y^t| &\leq |\sum_{i=2}^{P} c_i \mathrm{He}_i(\boldsymbol{\beta}^\top \boldsymbol{x}^t_{1:r})| + \tau \\ &\leq \sqrt{\sum_{i=2}^{P} c_i^2}\sqrt{\sum_{i=2}^{P} \mathrm{He}_i(\boldsymbol{\beta}^\top \boldsymbol{x}^t_{1:r})^2} + \tau\end{aligned}$$

holds. Let $\mathrm{He}_i(z) = \sum_{j=0}^{P} h_{ij} z^j$. Define $f(z)$ as $f(z) = \sum_{j=0}^{P} c_j z^j$ where $c_j = \max_{i=2}^{P} |h_{ij}|$. Then, one can show that $\mathrm{He}_i(z)^2 \leq f(|z|)^2$ for all i and z. Moreover, $f(z)$ is increasing in the region $z \geq 0$. Using this f, we can obtain

$$\begin{aligned}|y^t| &\leq \sqrt{\sum_{i=2}^{P} c_i^2}\sqrt{\sum_{i=2}^{P} f(R)^2} + \tau \\ &= \sqrt{\sum_{i=2}^{P} c_i^2}\sqrt{P-1} f(R) + \tau\end{aligned}$$

Then, the lemma immediately follows from vector-value contraction inequality (Maurer, 2016), by letting $g(z) = \sqrt{\sum_{i=2}^{P} c_i^2}\sqrt{P-1} f(z)$. □

Moreover, we can observe that

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} \sum_{t=1}^{T}\Big(\epsilon^{t,1}(\boldsymbol{w}^\top \boldsymbol{x}^t + b) + \epsilon^{t,2} y^t + \epsilon^{t,3}(\boldsymbol{w}'^\top \boldsymbol{x}'^t + b')\Big)\middle| A_{R,R'}\right] \\ \leq &\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} \sum_{t=1}^{T}\epsilon^{t,1}(\boldsymbol{w}^\top \boldsymbol{x}^t + b) + \sum_{t=1}^{T}\epsilon^{t,2} y^t + \sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} \sum_{t=1}^{T}\epsilon^{t,3}(\boldsymbol{w}'^\top \boldsymbol{x}'^t + b')\middle| A_{R,R'}\right] \\ = &\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} \sum_{t=1}^{T}\epsilon^{t,1}(\boldsymbol{w}^\top \boldsymbol{x}^t + b) + \sup_{\boldsymbol{w},b,\boldsymbol{w}',b'} \sum_{t=1}^{T}\epsilon^{t,3}(\boldsymbol{w}'^\top \boldsymbol{x}'^t + b')\middle| A_{R,R'}\right] \\ = &2\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{\boldsymbol{w}} \boldsymbol{w}^\top \sum_{t=1}^{T}\epsilon^t \boldsymbol{x}^t\middle| A_{R,R'}\right] + 2\mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{x}',\epsilon}\left[\sup_{b} \sum_{t=1}^{T}\epsilon^t b\middle| A_{R,R'}\right] \\ \leq &2W\mathbb{E}_{\boldsymbol{x}}\left[\sqrt{\left\|\sum_{t=1}^{T}\epsilon^t \boldsymbol{x}^t\right\|^2}\middle| A_{R,R'}\right] + 2B\mathbb{E}_{\epsilon}\left[\sqrt{\left(\sum_{t=1}^{T}\epsilon^t\right)^2}\right] \\ \leq &2W\mathbb{E}_{\boldsymbol{x}}\sqrt{\left[\left\|\sum_{t=1}^{T}\epsilon^t \boldsymbol{x}^t\right\|^2\middle| A_{R,R'}\right]} + 2B\sqrt{\mathbb{E}_{\epsilon}\left[\left(\sum_{t=1}^{T}\epsilon^t\right)^2\right]} \\ \leq &2W\sqrt{T}(R + R') + 2B\sqrt{T}.\end{aligned}$$

Finally we should evaluate

$$\begin{aligned}&\mathbb{E}_{R,R'}\left[\frac{2\sqrt{2}GmL}{T}\Big(2W\sqrt{T}(R + R') + 2B\sqrt{T}\Big)\right] \\ = &\frac{mGC'_{W,B}}{\sqrt{T}}\mathbb{E}_{R,R'}\left[2(R + R')(W(R + R') + B)(g(R) + \tau) + (R + R')(W(R + R') + B)^2\right] \\ + &\frac{mGC'_{W,B}}{\sqrt{T}}\mathbb{E}_{R,R'}\left[2(W(R + R') + B)(g(R) + \tau) + (W(R + R') + B)^2\right]\end{aligned}$$

$$= \frac{mG}{\sqrt{T}} O\big(\mathbb{E}_R\big[R^{P+1}\big] + \mathbb{E}_{R'}[R']\mathbb{E}_R\big[R^P\big] + \mathbb{E}_{R,R'}[(R+R')^3]\big),$$

where $C'_{W,B}$ is a constant which only depends on W and B. in order to do so, it suffices to evaluate $\mathbb{E}_R\big[R^k\big]$ and where $\mathbb{E}_{R'}\big[(R')^k\big]$.

Lemma 23.

$$\mathbb{E}_{z_1,\dots,z_{2T}\sim\chi(r)}\left[\max_{t\in[2T]} z_t^k\right] = O_{r,T}\left(\left(r+\sqrt{8r}\right)^{k/2}(1+\log T)^{k/4}\right)$$

holds. Note that we regard k as $O_{r,T}(1)$.

Proof. From the concentration inequalty for chi-squared distribution (Wainwright, 2019)[Example 2.11],

$$z_t^2 \le r + \sqrt{8r\log(1/\delta)} \tag{D.1}$$

holds with probability at least $1-\delta$ for each t. Then, equation (D.1) holds uniformly over all t with probability at least $1-2T\delta$. Let $A_i = (r+\sqrt{8r\log(2T\cdot 2^i)})^{k/2}$. Then, with probability at least $1-2^{-i}$, $\max_{t\in[2T]} z_t^k \le A_i$ holds. Therefore we can divide the certain event into events with probability $1/2, 1/4, \dots$, where $\max_{t\in[2T]} z_t^k \le A_1, \max_{t\in[2T]} z_t^k \le A_2, \dots$ holds. It implies that

$$\begin{aligned}
\mathbb{E}_{z_1,\dots,z_{2T}\sim\chi(r)}\left[\max_{t\in[2T]} z_t^k\right] &\le \sum_{i=1}^{\infty} 2^{-i}(r+\sqrt{8r\log(2T\cdot 2^i)})^{k/2} \\
&\le 2\int_0^{1/2}\left(r+\sqrt{8r\log\frac{2T}{t}}\right)^{k/2}\mathrm{d}t \\
&\le \left(r+\sqrt{8r}\right)^{k/2}\int_0^{1/2}\left(\sqrt{\log\frac{2T}{t}}\right)^{k/2}\mathrm{d}t \\
&\le \left(r+\sqrt{8r}\right)^{k/2}(1+\log T)^{k/4}\int_0^{1/2}\left(\sqrt{\log\frac{2}{t}}\right)^{k/2}\mathrm{d}t.
\end{aligned}$$

□

From Lemma 23, we arrive at

$$\mathrm{Rad}_T(\mathcal{F}_{N,G,W,B}) = O\left(\mathrm{polylog}(T)\frac{mG}{\sqrt{T}}\left(r^{P/2}\sqrt{d}+d\sqrt{d}\right)\right).$$

□

D.2. Prompt Length-free Generalization Bound

Let the ICL risk for prompt length N be

$$\mathcal{R}_N(\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b}) = \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},y}[|f(\boldsymbol{X}_{1:N},\boldsymbol{y}_{1:N},\boldsymbol{x};\boldsymbol{W},\boldsymbol{\Gamma},\boldsymbol{b})-y|],$$

where the length of $\boldsymbol{X}_{1:N}$ and $\boldsymbol{y}_{1:N}$ is fixed to N. In this section, we upper bound $|\mathcal{R}_N(\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b}) - \mathcal{R}_M(\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b})|$ under the condition $N, M \gtrsim r^{\Theta(P)}$.

Proposition 24. *Assume that $\|\boldsymbol{w}_{j,1:r}\| = O(1), \|\boldsymbol{w}_{j,1:r}\| = O(\sqrt{r/d})$ and $|b_j| = O(1)$ for each $j\in[m]$. Then,*

$$|\mathcal{R}_N(\boldsymbol{\Gamma},\boldsymbol{W},\boldsymbol{b}) - \mathcal{R}_M(\boldsymbol{\Gamma},\boldsymbol{w},\boldsymbol{b})| = \tilde{O}\left(\|\boldsymbol{\Gamma}\|_F\sqrt{r^2m^2/N + r^2m^2/M}\right)$$

holds.

Proof. Note that

$$
\begin{aligned}
&|\mathcal{R}_N(\mathbf{\Gamma}, \boldsymbol{W}, \boldsymbol{b}) - \mathcal{R}_M(\mathbf{\Gamma}, \boldsymbol{w}, \boldsymbol{b})| \\
\leq &\mathbb{E}[|f(\boldsymbol{X}_{1:N}, \boldsymbol{y}_{1:N}, \boldsymbol{x}; \boldsymbol{W}, \mathbf{\Gamma}, \boldsymbol{b}) - f(\boldsymbol{X}_{1:M}, \boldsymbol{y}_{1:M}, \boldsymbol{x}; \boldsymbol{W}, \mathbf{\Gamma}, \boldsymbol{b})|] \\
= &\mathbb{E}\left[\left|\left\langle \mathbf{\Gamma}\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right), \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\rangle \right|\right] \\
\leq &\|\mathbf{\Gamma}\|_F \mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\| \left\| \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\|\right] \\
\leq &\|\mathbf{\Gamma}\|_F \mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\|^2\right]^{1/2} \mathbb{E}\left[\left\| \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m) \end{bmatrix} \right\|^2\right]^{1/2}
\end{aligned}
$$

First let us bound $\mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\|^2\right]$.

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\|^2\right] \\
\leq &\sum_{j=1}^m \mathbb{E}\left[\left(\frac{1}{N}\sum_{i=1}^N \sigma(\boldsymbol{w}_j^\top \boldsymbol{x}_i + b_j) y_i - \frac{1}{M}\sum_{i=1}^M \sigma(\boldsymbol{w}_j^\top \boldsymbol{x}_i + b_j) y_i\right)^2\right]
\end{aligned}
$$

For simplicity, we drop the subscript j and obtain

$$
\begin{aligned}
&\mathbb{E}\left[\left(\frac{1}{N}\sum_{i=1}^N \sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i - \frac{1}{M}\sum_{i=1}^M \sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\right)^2\right] \\
\leq &2\mathbb{E}\left[\left(\frac{1}{N}\sum_{i=1}^N \sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i - \mathbb{E}\big[\sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big]\right)^2\right] \\
&+2\mathbb{E}\left[\left(\frac{1}{M}\sum_{i=1}^M \sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i - \mathbb{E}\big[\sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big]\right)^2\right] \\
\leq &\frac{2}{N}\mathbb{E}\Big[\big(\sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big)^2\Big] + \frac{2}{M}\mathbb{E}\Big[\big(\sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big)^2\Big] \\
\leq &\frac{2}{N}\mathbb{E}\Big[\big((\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big)^2\Big] + \frac{2}{M}\mathbb{E}\Big[\big((\boldsymbol{w}^\top \boldsymbol{x}_i + b) y_i\big)^2\Big] \\
\leq &\frac{4}{N}\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2 + b^2 y_i^2\big] + \frac{4}{M}\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2 + b^2 y_i^2\big] \\
\leq &\frac{4}{N}\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2\big] + \frac{4}{M}\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2\big] + O(1/N + 1/M).
\end{aligned}
$$

Then, we need to evaluate $\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2\big]$; we need to be careful to obtain a tight bound for this value. Using the fact that pretrained $\boldsymbol{w}$ almost aligns to the true subspace $\mathcal{S}$, we obtain

$$
\begin{aligned}
\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2\big] &\leq \mathbb{E}\big[(\boldsymbol{w}_{1:r}^\top \boldsymbol{x}_{i,1:r} + \boldsymbol{w}_{r+1:d}^\top \boldsymbol{x}_{i,r+1:d})^2 y_i^2\big] \\
&\leq 2\|\boldsymbol{w}_{1:r}\|^2 \mathbb{E}\big[(\boldsymbol{x}_{i,1:r})^2 y_i^2\big] + 2\|\boldsymbol{w}_{r+1:d}\|^2 \mathbb{E}\big[(\boldsymbol{x}_{i,r+1:d})^2 y_i^2\big]
\end{aligned}
$$

$$= O(1) \cdot \mathbb{E}\big[(\boldsymbol{x}_{i,1:r})^2 y_i^2\big] + \tilde{O}(r/d) \cdot \mathbb{E}\big[(\boldsymbol{x}_{i,r+1:d})^2 y_i^2\big].$$

Moreover,

$$\begin{aligned}
&\mathbb{E}\big[x_1^2 y^2\big] \\
\leq& 2\mathbb{E}_{c,\beta,\varsigma}\left[x_1^2 (\sum_{i=2}^P c_i^2)(\sum_{i=2}^P \mathrm{He}_i(\boldsymbol{\beta}^\top \boldsymbol{x})^2)\right] + 2\mathbb{E}_{c,\beta,\varsigma}\big[x_1^2 \varsigma^2\big] \\
\lesssim& \sum_{i=2}^P \mathbb{E}_{c,\beta,\varsigma}\Big[x_1^2 \mathrm{He}_i(\boldsymbol{\beta}^\top \boldsymbol{x})^2\Big] + O(1) \\
\leq& \sum_{i=2}^P \mathbb{E}\big[x_1^4\big]^{1/2} \mathbb{E}\Big[\mathrm{He}_i(\boldsymbol{\beta}^\top \boldsymbol{x})^4\Big]^{1/2} + O(1) = O(1).
\end{aligned}$$

Then, we obtain $\mathbb{E}\big[(\boldsymbol{w}^\top \boldsymbol{x}_i)^2 y_i^2\big] = \tilde{O}(r)$. Thus, an upper bound

$$\mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\|^2\right] = \tilde{O}(rm/N + rm/M)$$

is obtained.

Second, we bound $\mathbb{E}\left[\left\|\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m)\end{bmatrix}\right\|^2\right]$ as

$$\begin{aligned}
\mathbb{E}\left[\left\|\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m)\end{bmatrix}\right\|^2\right] &\leq 2\sum_{j=1}^m \mathbb{E}[(\boldsymbol{w}_j^\top \boldsymbol{x})^2 + b_j^2] \\
&\leq 2\sum_{j=1}^m (b_j^2 + 4\|\boldsymbol{w}_{j,1:r}\|^2 \mathbb{E}[\|\boldsymbol{x}_{1:r}\|^2] + 4\|\boldsymbol{w}_{j,r+1:d}\|^2 \mathbb{E}[\|\boldsymbol{x}_{r+1:d}\|^2]) \\
&= \tilde{O}(mr).
\end{aligned}$$

Putting all things together, we arrive at

$$\begin{aligned}
&|\mathcal{R}_N(\boldsymbol{\Gamma}, \boldsymbol{W}, \boldsymbol{b}) - \mathcal{R}_M(\boldsymbol{\Gamma}, \boldsymbol{w}, \boldsymbol{b})| \\
\leq& \|\boldsymbol{\Gamma}\|_F \mathbb{E}\left[\left\|\left(\frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:N} + \boldsymbol{b})\boldsymbol{y}_{1:N}}{N} - \frac{\sigma(\boldsymbol{W}\boldsymbol{X}_{1:M} + \boldsymbol{b})\boldsymbol{y}_{1:M}}{M}\right)\right\|^2\right]^{1/2} \mathbb{E}\left[\left\|\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top \boldsymbol{x} + b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x} + b_m)\end{bmatrix}\right\|^2\right]^{1/2} \\
=& \tilde{O}\Big(\|\boldsymbol{\Gamma}\|_F \sqrt{r^2 m^2/N + r^2 m^2/M}\Big).
\end{aligned}$$

□

D.3. Proof of Theorem 1

Finally we are ready to prove our main theorem.

Proof. [Proof of Theorem 1] Let $\bar{\boldsymbol{\Gamma}}$ be the attention matrix constructed in Theorem 8 and let $\boldsymbol{\Gamma}^*$ be the minimizer of the ridge regression problem (line 6 in Algorithm 1). By the equivalence between optimization with L_2 regularization and norm-constrained optimization, there exists $\lambda_2 > 0$ such that

$$\|\boldsymbol{\Gamma}^*\|_F \leq \|\bar{\boldsymbol{\Gamma}}\|_F = O(\sqrt{r^{5P}}/m),$$

$$\left(\frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right)^2 \le \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}(y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b}))^2$$
$$\le \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}(y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\bar{\boldsymbol{\Gamma}},\boldsymbol{b}))^2.$$

Then, from Theorem 8,

$$\frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})| - \tau \lesssim \mathrm{poly}\log(d)\left(\frac{r^{5P/2}}{\sqrt{N_2}} + \frac{r^{3P/2}}{\sqrt{m}}\right)$$

holds.

We first evaluate $\mathcal{R}_{N_2}(f) - \tau$ where $f = f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})$: First,

$$\begin{aligned}
&\mathcal{R}_{N_2}(f) - \tau \\
&= \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})| \\
&\quad + \left(\mathcal{R}_{N_2}(f) - \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right) - \tau \\
&\lesssim \mathrm{poly}\log(d)\left(\frac{r^{5P/2}}{\sqrt{N_2}} + \frac{r^{3P/2}}{\sqrt{m}}\right) \\
&\quad + \sup_{f\in\mathcal{F}_{N,G,W,B}}\left(\mathcal{R}_{N_2}(f) - \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right)
\end{aligned}$$

holds, noting that $f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b}) \in \mathcal{F}_{N,G,W,B}$ with $W = O(1), B = O(1)$ and $G = \tilde{O}(\sqrt{r^{5P}/m^2})$. We can evaluate the expectation value of the second term of the last line using the Rademacher complexity as

$$\begin{aligned}
&\mathbb{E}\left[\sup_{f\in\mathcal{F}_{N,G,W,B}}\left(\mathcal{R}_{N_2}(f) - \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right)\right] \\
&= \mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},y}\left[\sup_{f\in\mathcal{F}_{N,G,W,B}}\left(\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},y}[|y - f(\boldsymbol{X},\boldsymbol{y},\boldsymbol{x};\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|]\right.\right. \\
&\qquad \left.\left. - \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right)\right] \\
&\le 2\mathbb{E}_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{x},y,\boldsymbol{\epsilon}}\left[\sup_{f\in\mathcal{F}_{N,G,W,B}}\frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}\epsilon_t|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right] \ (\epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}\{\pm 1\}) \\
&\lesssim \mathrm{Rad}_{T_2}(\mathcal{F}_{N,G,W,B}) + \mathbb{E}_{y,\boldsymbol{\epsilon}}\left[\frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}\epsilon_t y_t\right] \ (\because \text{Eq.(1) in } (Maurer,\ 2016)) \\
&\lesssim \mathrm{Rad}_{T_2}(\mathcal{F}_{N,G,W,B}) + \mathbb{E}_{y,\boldsymbol{\epsilon}}\left[\left(\frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}\epsilon_t y_t\right)^2\right]^{1/2} \\
&\le \mathrm{Rad}_{T_2}(\mathcal{F}_{N,G,W,B}) + \frac{1}{\sqrt{T_2}}\mathbb{E}[y^2]^{1/2} = \tilde{O}\left(\mathrm{polylog}(T_2)\frac{r^{5P/2}}{\sqrt{T_2}}\left(r^{P/2}\sqrt{d} + d\sqrt{d}\right)\right)
\end{aligned}$$

by Proposition 20 and $\mathbb{E}[y^2] = O(1)$ from Assumption 1. Then, from Markov's inequality, we have $\sup_{f\in\mathcal{F}_{N,G,W,B}}\left(\mathcal{R}_{N_2}(f) - \frac{1}{T_2}\sum_{t=T_1+1}^{T_1+T_2}|y_t - f(\boldsymbol{X}_t,\boldsymbol{y}_t,\boldsymbol{x}_t;\boldsymbol{W}^{(1)},\boldsymbol{\Gamma}^*,\boldsymbol{b})|\right) =$

$\tilde{O}\Big(\mathrm{polylog}(T_2)\frac{r^{5P/2}}{\sqrt{T_2}}\Big(r^{P/2}\sqrt{d}+d\sqrt{d}\Big)\Big)$ with probability at least $1-\delta$ where δ is a sufficiently small constant. Then, we obtain

$$\mathcal{R}_{N_2}(f)-\tau=\mathrm{poly}\log(d)\bigg(\frac{r^{5P/2}}{\sqrt{N_2}}+\frac{r^{3P/2}}{\sqrt{m}}+\mathrm{polylog}(T_2)\frac{r^{5P/2}}{\sqrt{T_2}}\Big(r^{P/2}\sqrt{d}+d\sqrt{d}\Big)\bigg).$$

Now we have done the upper bound for $\mathcal{R}_{N_2}(f)$. For $\mathcal{R}_{N^*}(f)$, we can use Proposition 24 because we have Corollary 7 and can ensure that the assumptions of Proposition 24 are satisfied. □

E. Derivation of Simplified Self-attention Module

We derive equation (2.3), following the same line as (Zhang et al., 2023). Recall that the prediction of y by the original self-attention module is defined as the right-bottom entry of

$$f_{\mathrm{Attn}}=\boldsymbol{E}+\boldsymbol{W}^P\boldsymbol{W}^V\boldsymbol{E}\cdot\mathrm{softmax}\bigg(\frac{(\boldsymbol{W}^K\boldsymbol{E})^\top\boldsymbol{W}^Q\boldsymbol{E}}{\rho}\bigg),$$

where the embedding matrix is given as (2.2). As mentioned in Section 2.2, we set $\rho=N$, omit softmax and merge $\boldsymbol{W}^P\boldsymbol{W}^V$ as $\boldsymbol{W}^{PV}\in\mathbb{R}^{(m+1)\times(m+1)}$ and $(\boldsymbol{W}^K)^\top\boldsymbol{W}^Q$ as $\boldsymbol{W}^{KQ}\in\mathbb{R}^{(m+1)\times(m+1)}$.

Now, we further assume that $\boldsymbol{W}^{PV}$ $\boldsymbol{W}^{KQ}$ are in the form as

$$\boldsymbol{W}^{PV}=\begin{bmatrix}* & *\\ 0_{1\times m} & v\end{bmatrix},\boldsymbol{W}^{KQ}=\begin{bmatrix}\boldsymbol{K} & *\\ 0_{1\times m} & *\end{bmatrix}.$$

Then, we obtain the simplified form

$$\tilde{f}_{\mathrm{Attn}}(\boldsymbol{E};\boldsymbol{W}^K,\boldsymbol{W}^Q,\boldsymbol{W}^V,\boldsymbol{W}^P)=\boldsymbol{E}+\begin{bmatrix}* & *\\ 0_{1\times m} & v\end{bmatrix}\boldsymbol{E}\cdot\frac{\boldsymbol{E}^\top\begin{bmatrix}\boldsymbol{K} & *\\ 0_{1\times m} & *\end{bmatrix}\boldsymbol{E}}{N}.\tag{E.1}$$

Note that we adopt the right-bottom entry $\Big(\tilde{f}_{\mathrm{Attn}}(\boldsymbol{E};\boldsymbol{W}^K,\boldsymbol{W}^Q,\boldsymbol{W}^V,\boldsymbol{W}^P)\Big)_{m+1,N+1}$ as prediction for a response of a query. Then, by (E.1), we obtain

$$\begin{aligned}
&\Big(\tilde{f}_{\mathrm{Attn}}(\boldsymbol{E};\boldsymbol{W}^K,\boldsymbol{W}^Q,\boldsymbol{W}^V,\boldsymbol{W}^P)\Big)_{m+1,N+1}\\
=&\left(\begin{bmatrix}0_{1\times m} & v\end{bmatrix}\boldsymbol{E}\cdot\frac{\boldsymbol{E}^\top\begin{bmatrix}\boldsymbol{K} & *\\ 0_{1\times m} & *\end{bmatrix}\boldsymbol{E}}{N}\right)_{m+1,N+1}\\
=&\begin{bmatrix}0_{1\times m} & v\end{bmatrix}\boldsymbol{E}\cdot\frac{\boldsymbol{E}^\top\begin{bmatrix}\boldsymbol{K} & *\\ 0_{1\times m} & *\end{bmatrix}\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}+b_1)\\ \vdots\\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}+b_m)\\ 0\end{bmatrix}}{N}\\
=&v\begin{bmatrix}y_1 & \cdots & y_N & 0\end{bmatrix}\cdot\frac{\boldsymbol{E}^\top\begin{bmatrix}\boldsymbol{K}\begin{bmatrix}\sigma(\boldsymbol{w}_1^\top\boldsymbol{x}+b_1)\\ \vdots\\ \sigma(\boldsymbol{w}_m^\top\boldsymbol{x}+b_m)\end{bmatrix}\\ 0\end{bmatrix}}{N}
\end{aligned}$$

$$= v\begin{bmatrix} y_1 & \cdots & y_N \end{bmatrix} \cdot \frac{\sigma(\boldsymbol{W}\boldsymbol{X}+\boldsymbol{b})^\top \boldsymbol{K} \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x}+b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x}+b_m) \end{bmatrix}}{N}$$

$$= \left\langle \frac{v\boldsymbol{K}^\top \sigma(\boldsymbol{W}\boldsymbol{X}+\boldsymbol{b})\boldsymbol{y}}{N}, \begin{bmatrix} \sigma(\boldsymbol{w}_1^\top \boldsymbol{x}+b_1) \\ \vdots \\ \sigma(\boldsymbol{w}_m^\top \boldsymbol{x}+b_m) \end{bmatrix} \right\rangle.$$

Letting $\boldsymbol{\Gamma} = v\boldsymbol{K}^\top$ yields equation (2.3).