What and How does In-Context Learning Learn? Bayesian Model Averaging, Parameterization, and Generalization

Yufeng Zhang[*,†] OpenAI
Fengzhuo Zhang[*] National University of Singapore
Zhuoran Yang Yale University
Zhaoran Wang Northwestern University

Abstract

In-Context Learning (ICL) ability has been found efficient across a wide range of applications, where the Large Language Models (LLM) learn to complete the tasks from the examples in the prompt without tuning the parameters. In this work, we conduct a comprehensive study to understand ICL from a statistical perspective. First, we show that the perfectly pretrained LLMs perform Bayesian Model Averaging (BMA) for ICL under a dynamic model of examples in the prompt. The average error analysis for ICL is then built for the perfectly pretrained LLMs with the analysis of BMA. Second, we demonstrate how the attention structure boosts the BMA implementation. With sufficient examples in the prompt, attention is proven to perform BMA under the Gaussian linear ICL model, which also motivates the explicit construction of the hidden concepts from the attention heads values. Finally, we analyze the pretraining behavior of LLMs. The pretraining error is decomposed as the generalization error and the approximation error. The generalization error is upper bounded via PAC-Bayes framework. Then the ICL average error of the pretrained LLMs is shown to be the sum of $O(T^{-1})$ and the pretraining error. In addition, we analyze the ICL performance of the pretrained LLMs with misspecified examples.

[*]These authors contributed equally to this work.
[†]Yufeng performed this work when he was with Northwestern University.

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

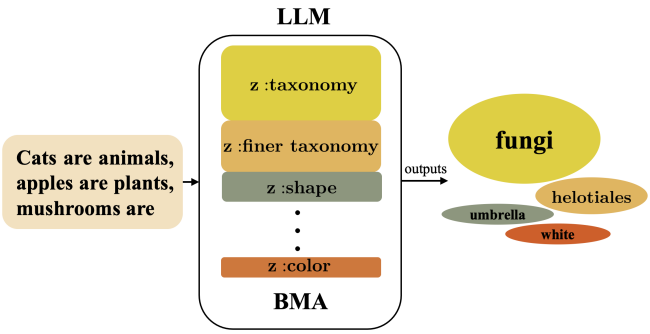

Figure 1: LLMs implement BMA for ICL. They estimate the posterior of hidden concept z from examples and use the posterior to mix the conditional probability of response on the query and hidden concept z.

1 Introduction

With the ever-increasing sizes of model capacity and corpus, Large Language Models (LLM) have achieved tremendous successes across a wide range of tasks (Dong et al., 2019; Wei et al., 2022c; Kojima et al., 2022; Ouyang et al., 2022). Recent studies have revealed that these LLMs possess immense potential, as their large capacity allows for a series of *emergent abilities* (Wei et al., 2022b; Liu et al., 2023). One such ability is ICL, which enables an LLM to learn from just a few examples, without changing the network parameters. Despite the tremendous empirical successes, theoretical understanding of ICL remains limited. Specifically, existing works fail to explain why LLMs have the ability for ICL, how the attention mechanism is related to the ICL ability, and how pretraining influences ICL. Although the optimality of ICL is investigated in Xie et al. (2021) and Wies et al. (2023), these works both make unrealistic assumptions on the pretrained models, and their results cannot demystify the particular role played by the attention mechanism in ICL.

In this work, we focus on the scenario where a transformer is first pretrained on a large dataset and then prompted to perform ICL. Our goal is to rigor-

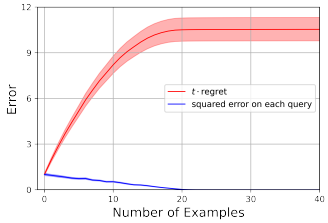

Figure 2: The cumulative squared error of LLMs trained for linear regression is bounded by a constant. It verifies Propostion 3.4, which is a result of Propostion 3.3.

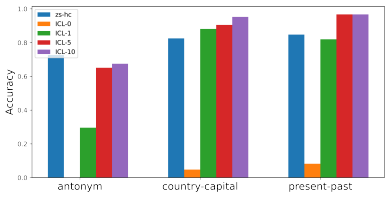

Figure 3: The constructed hidden concept provides sufficient information for LLM. Conditioned on it in a zero-shot setting, LLMs have comparable performance with ICL with several examples. It verifies Propostions 3.3 and 3.6.

Figure 4: The ratios between $\texttt{attn}_\dagger$ and $\texttt{attn}$ converge to a constant that depends on the dimension when the example number T tends to infinity. It verifies Propostion 3.6.

ously understand why the practice of "pretraining + prompting" unleashes the power of ICL. To this end, we aim to answer the following three questions: **(a)** What type of ICL estimator is learned by LLMs? **(b)** What are suitable performance metrics to evaluate ICL accurately and what are the error rates? **(c)** What is the role played by the transformer architecture during the pretraining and prompting stages? The first and the third questions demand scrutinizing the transformer architecture to understand how ICL happens during transformer prompting. The second question then requires statistically analyzing the extracted ICL process. Moreover, the third question necessitates a holistic understanding beyond prompting — we also need to characterize the statistical error of *pretraining* and how this error affects prompting.

To address these questions, we adopt a Bayesian view and assume that the examples fed into a pretrained LLM are sampled from a hidden variable model parameterized by a hidden concept $z_* \in \mathfrak{Z}$. Moreover, the pretraining dataset contains sequences of examples from the same hidden variable model, but with the concept parameter $z \in \mathfrak{Z}$ itself randomly distributed according to a prior distribution. We mathematically formulate ICL as the problem of predicting the response of the current covariate, where the prompt contains t examples of covariate-response pairs and the current covariate.

Under such a setting, to answer **(a)**, we show that the perfectly pretrained LLMs perform ICL in the form of BMA under a dynamical data model. That is, LLM first computes a posterior distribution of $z_* \in \mathfrak{Z}$ given the first t examples, and then predicts the response of the $(t+1)$-th covariate by aggregating over the posterior (Proposition 3.3), which is empirically verified in Section 3.2.

In addition, to answer **(b)**, we adopt the online learning framework and define a notion called ICL average error, which is the averaged prediction error of ICL on a sequence of covariate-response examples. We prove that the ICL average error after prompting t examples is $\mathcal{O}(1/t)$ up to the statistical error of the pretrained model (Theorem 5.2), which is validated by experiments in Section 3.2.

Finally, to answer **(c)**, we elucidate the role played by the transformer architecture in prompting and pretraining respectively. In particular, we show that a variant of attention mechanism encodes BMA in its architecture under a linear Gaussian model, which enables the transformer to perform ICL via prompting. Such an attention mechanism can be viewed as an extension of linear attention and coincides with the standard softmax attention (Garnelo and Czarnecki, 2023) when the length of the prompt goes to infinity. Thus we show that softmax attention Vaswani et al. (2017) approximately encodes BMA (Proposition 3.6), which is empirically verified in Section 3.4. Besides, the transformer architecture enables a fine-grained statistical error analysis of pretraining. We prove that the error of the pretrained language model, measured via total variation, is bounded by a sum of approximation error and generalization error (Theorem 4.3). The generalization error decays to zero sublinearly in the number of tokens for pretraining. This features the first pretraining analysis of transformers in total variation distance that also takes the approximation error into account.

In sum, by addressing questions **(a)**–**(c)**, we provide a unified understanding of the ICL ability of LLMs and the particular role played by the attention mechanism. Our theory provides a holistic theoretical understanding of the ICL average error and pretraining errors of ICL.

2 Preliminary

Attention and Transformers. Attention mechanism has been the most powerful and popular neural network module in natural language processing, and it is the backbone of the LLMs (Devlin et al., 2018; Brown et al., 2020). Assume that we have a query vector $q \in \mathbb{R}^{d_k}$. With T key vectors in $K \in \mathbb{R}^{T\times d_k}$ and T value vectors in $V \in \mathbb{R}^{T\times d_v}$, the attention mechanism maps the query vector q to $\texttt{attn}(q, K, V) = V^\top \texttt{softmax}(Kq)$, where $\texttt{softmax}$ normalizes a vector via the exponential function, i.e., for $x \in \mathbb{R}^d$, $[\texttt{softmax}(x)]_i = \exp(x_i)/\sum_{j=1}^d \exp(x_j)$ for $i \in [d]$. The output is a weighted sum of V, and the weights reflect the closeness between W and q. For t query vectors, we stack them into $Q \in \mathbb{R}^{t\times d_k}$. Attention maps these queries using the function $\texttt{attn}(Q, K, V) = \texttt{softmax}(QK^\top)V \in \mathbb{R}^{t\times d_v}$, where $\texttt{softmax}$ is applied row-wisely. In the practical design of transformers, practitioners usually use Multi-Head Attention (MHA) instead of single head attention to express sophisticated functions, which forwards the inputs through h attention modules in parallel and outputs the sum of these sub-modules. Here $h \in \mathbb{N}$ is a hyperparameter. Taking $X \in \mathbb{R}^{T\times d}$ as the input, MHA outputs $\texttt{mha}(X, W) = \sum_{i=1}^h \texttt{attn}(XW_i^Q, XW_i^K, XW_i^V)$, where $W = (W_i^Q, W_i^K, W_i^V)_{i=1}^h$ is the parameters set of h attention modules, $W_i^Q \in \mathbb{R}^{d\times d_h}$, $W_i^K \in \mathbb{R}^{d\times d_h}$, and $W_i^V \in \mathbb{R}^{d\times d}$ for $i \in [h]$ are weight matrices for queries, keys, and values, and d_h is usually set to be d/h (Michel et al., 2019). The transformer is the concatenation of the attention modules and the fully-connected layers, which is widely adopted in LLMs (Brown et al., 2020).

Large Language Models and In-Context Learning. Many LLMs are *autoregressive*, such as GPT (Brown et al., 2020). It means that the model continuously predicts future tokens based on its own previous values. For example, starting from a token $x_1 \in \mathfrak{X}$, where $\mathfrak{X}$ is the alphabet of tokens, a LLM $\mathbb{P}_\theta$ with parameter $\theta \in \Theta$ continuously predicts the next token according to $x_{t+1} \sim \mathbb{P}_\theta(\cdot \,|\, S_t)$ based on the past $S_t = (x_1, \cdots, x_t)$ for $t \in \mathbb{N}$. Here, each token represents a word and the position of the word (Ke et al., 2020), and the token sequences S_t for $t \in \mathbb{N}$ live in the sequences space $\mathfrak{X}^*$. LLMs are first *pretrained* on a huge body of corpus, making the prediction $x_{t+1} \sim \mathbb{P}_\theta(\cdot \,|\, S_t)$ accurate, and then prompted to perform downstream tasks. During the pretraining phase, we aim to maximize the conditional probability $\mathbb{P}_\theta(x \,|\, S)$ over the nominal next token x (Brown et al., 2020).

After pretraining, LLMs are prompted to perform downstream tasks without tuning parameters. Different from the finetuned models that learn the task explicitly (Liu et al., 2023), LLMs can implicitly learn from the examples in the *prompt*, which is known as ICL (Brown et al., 2020). Concretely, pretrained LLMs are provided with a prompt $\texttt{pt}_t = (\widetilde{c}_1, r_1, \ldots, \widetilde{c}_t, r_t, \widetilde{c}_{t+1})$ with t examples and a query as inputs, where each pair $(\widetilde{c}_i, r_i) \in \mathfrak{X}^* \times \mathfrak{X}$ is an example of the task, and $\widetilde{c}_{t+1}$ is the query, as shown in Figure 6 in Appendix 2. For example, the $\texttt{pt}_t$ with $t = 2$ can be "Cats are animals, apples are plants, mushrooms are". Here $\widetilde{c}_1 \in \mathfrak{X}^*$ is a token sequence "Cats are", while r_1 is the response "animals". The query $\widetilde{c}_{t+1}$ is "mushrooms are", and the desired response is "fungi". The prompts are generated from a hidden concept $z_* \in \mathfrak{Z}$, e.g., z_* can be the classification of biological categories, where $\mathfrak{Z}$ is the concept space. The generation process is $\widetilde{c}_i \sim \mathbb{P}_{\text{q}}$ and $r_i \sim \mathbb{P}(\cdot \,|\, \texttt{pt}_{i-1}, z_*)$ for the nominal distribution $\mathbb{P}$ and $i \in [t]$, where $\mathbb{P}_{\text{q}}$ is the covariate distribution. Thus, when performing ICL, LLMs aim to estimate the conditional distribution $\mathbb{P}(r_{t+1}|\texttt{pt}_t, z_*)$. It is widely conjectured and experimentally found that the pretrained LLMs can implicitly identify the hidden concept $z_* \in \mathfrak{Z}$ from the examples, and then perform ICL by outputting from $\mathbb{P}(r_{t+1}|\texttt{pt}_t, z_*)$. In the following, we will provide theoretical justifications for this claim. Since LLMs are autoregressive, the definition of the notation $\mathbb{P}(\cdot \,|\, S)$ with $S \in \mathfrak{X}^*$ may be ambiguous because the length of the subsequent tokens is not specified. Unless explicitly specified, we let $\mathbb{P}(\cdot \,|\, S)$ denote the distribution of the next single token conditioned on S.

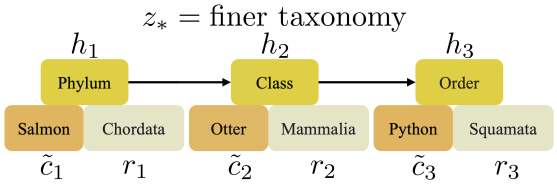

Figure 5: The hidden concept z_* is "finer taxonomy" makes the hidden variables h_t evolve to finer level according to (3.1). These hidden variables parameterize the relationship between $\widetilde{c}_t$ and r_t with (3.2).

3 In-Context Learning via Bayesian Model Averaging

3.1 ICL Statistical Models Enables Bayesian Model Averaging

Given a sequence $S = \{(\widetilde{c}_t, r_t)\}_{t=1}^T$ with T examples generated from a hidden concept $z_* \in \mathfrak{Z}$, we use $S_t = \{(\widetilde{c}_i, r_i)\}_{i=1}^t$ to represent the first t ICL examples in the sequence. Here $\widetilde{c}_t$ and r_t respectively denote the ICL covariate and response. During the ICL phase, a LLM is sequentially prompted with $\texttt{pt}_t = (S_t, \widetilde{c}_{t+1})$

for $t \in [T-1]$, i.e., the first t examples and the $(t+1)$-th covariate. The prompted LLM aims to predict the response r_{t+1} based on $\mathtt{pt}_t = (S_t, \widetilde{c}_{t+1})$ whose true distribution is $r_{t+1} \sim \mathbb{P}(\cdot \,|\, \mathtt{pt}_t, z_*)$. For the analysis of ICL, we focus on the following hidden variable model.

Assumption 3.1 (Dynamic Hidden Variable Model)**.** The hidden concept $z_* \in \mathfrak{Z}$ parameterizes the distributions of hidden variables $\{h_t\}_{t=1}^T \in \mathcal{H}^T$ as

$$h_t = g_{z_*}(h_1, \ldots, h_{t-1}, \zeta_t), \tag{3.1}$$

where g_{z_*} is a function parameterized by z_*, and $\{\zeta_t\}_{t=1}^T$ are exogenous noises. These hidden variables parameterize the covariates and responses for ICL as

$$r_t = f(\widetilde{c}_t, h_t, \xi_t), \quad \widetilde{c}_t \sim \mathbb{P}_{\mathrm{q}} \quad \forall t \in [T], \tag{3.2}$$

where $\mathbb{P}_{\mathrm{q}}$ is the distribution of query, the hidden variable $h_t \in \mathcal{H}$ determines the relation between c_t and r_t, $\xi_t \in \Xi$ for $t \in [T]$ are i.i.d. random noises, and $f : \mathcal{X}^* \times \mathcal{H} \times \Xi \to \mathfrak{X}$ is a function that relates response r_t to $\widetilde{c}_t, h_t$, and ξ_t.

In the data generation process, a hidden concept $z_* \in \mathfrak{Z}$ is first generated from $\mathbb{P}(z)$. The covariates $\widetilde{c}_t$ are generated from $\mathbb{P}_{\mathrm{q}}$ in an i.i.d. manner. Then the hidden variables $\{h_t\}_{t=1}^T$ and responses $\{r_t\}_{t=1}^T$ are generated according to (3.1) and (3.2). The model in Assumption 3.1 essentially assumes that the hidden concept z_* implicitly determines the transition of the conditional distribution $\mathbb{P}(r_t = \cdot \,|\, \widetilde{c}_t)$ by affecting the evolution of the hidden variables $\{h_t\}_{t\in[T]}$. These hidden variables capture the relationship between examples $\{(\widetilde{c}_t, r_t)\}_{t=1}^T$, since the examples may have some inferent relationship and share some similarity when they are come up with by humans (Elman, 1995; Niyogi et al., 1997). In other words, this assumption assumes that the examples convey a *main semantic* z_*, e.g., "finer taxonomy" in Figure. 5. Then the response is generated from the hidden variable h, e.g., "class", and covariate, e.g., "otter" in Figure. 5. This model is quite general, and it subsumes the models in previous works.

Comparision with existing models The models in the existing works all assume that the examples are i.i.d., i.e., $h_t = g_{z_*}(\zeta_t)$ in the model of Assumption 3.1. For example, the Hiddn Markov Model (HMM) model in Xie et al. (2021) assumes that the hidden variables for each example are independently generated. In contrast, we allow them to depend on each other via (3.1). When the hidden variables $h_t = z_*$ for $t \in [T]$ degenerate to the hidden concept, the model in Assumption 3.1 recovers the topic model in Wang et al. (2023) and the ICL model in Jiang (2023).

Assuming that the tokens follow the statistical model given in (3.2), during pretraining, we collect N_p independent trajectories by sampling from (3.2) with concept z randomly sampled from $\mathbb{P}(z)$. Intuitively, by training in an autoregressive manner, the LLM approximates the conditional distribution $\mathbb{P}(r_{t+1} \,|\, \mathtt{pt}_t) = \mathbb{E}_{z\sim\mathbb{P}(z\,|\,\mathtt{pt}_t)}[\mathbb{P}(r_{t+1} \,|\, \mathtt{pt}_t, z)]$, which is the conditional distribution of r_{t+1} given $\mathtt{pt}_t$, aggregated over the randomness of the concept z_*. Given an infinite number of samples and the sufficiently large function class Θ, the pretrained LLMs can perfectly match the pretraining distribution.

Assumption 3.2 (Perfect Pretraining)**.** A LLM $\mathbb{P}_\theta$ is called perfectly pretrained if for all $\mathtt{pt} \in \mathcal{X}^*$ and $r \in \mathcal{X}$, we have $\mathbb{P}_\theta(r \,|\, \mathtt{pt}) = \mathbb{P}(r \,|\, \mathtt{pt})$, where $\mathbb{P}$ is the distribution induced by the model in Assumption 3.1.

We will relax this assumption in Section 4 by analyzing the pretraining error.

Proposition 3.3 (LLMs Perform BMA)**.** Under Assumptions 3.1 and 3.2, LLMs perform BMA for ICL, i.e.,

$$\mathbb{P}_\theta(r_{t+1} \,|\, \mathtt{pt}_t) = \int \mathbb{P}(r_{t+1} \,|\, \widetilde{c}_{t+1}, S_t, z)\mathbb{P}(z \,|\, S_t)\mathrm{d}z, \tag{3.3}$$

where $\mathbb{P}$ is the distribution induced by Assumption 3.1.

We note that the left-hand side of (3.3) is the prediction of the pretrained LLM given a prompt $\mathtt{pt}_t$. Meanwhile, the right-hand side is exactly the prediction given by the BMA algorithm that infers the posterior belief of the concept z_* based on S_t and predicts r_{t+1} by aggregating the likelihood in (3.2) with respect to the posterior $\mathbb{P}(z_* = \cdot \,|\, S_t)$, as shown in Figure 1. Thus, this proposition shows that perfectly pretrained LLMs are able to perform ICL because they **implement BMA during prompting**. As mentioned, Proposition 3.3 is proved under a more general model than the previous works and thus serves as a generalized result of some claims in the previous works. We note that the claim of Proposition 3.3 is independent of the network structure. This partially explains why LSTMs demonstrate ICL ability in Xie et al. (2021). In Section 3.3, we will demonstrate how the attention mechanism helps to implement BMA. The proof of Proposition 3.3 is in Appendix 5.2.

Next, we study the performance of ICL from an online learning perspective. Recall that LLMs are continuously prompted with S_t and aim to predict the $(t+1)$-th covariate r_{t+1} for $t \in [T-1]$. This can be viewed as an online learning problem. For any algorithm that generates a sequence of density estimators $\{\widehat{\mathbb{P}}(r_t)\}_{t=1}^T$ for predicting $\{r_t\}_{t\in[T]}$, we consider the following ICL average error as its performance metric:

$$\mathtt{AE}_t = \frac{1}{t}\sup_z \sum_{i=1}^t \Big(\log \mathbb{P}(r_i \,|\, \mathtt{pt}_{i-1}, z) - \log \widehat{\mathbb{P}}(r_i)\Big). \tag{3.4}$$

This ICL average error measures the performance of the estimator $\widehat{\mathbb{P}}$ compared with the best hidden concept in hindsight. For the perfectly trained LLMs, the estimator is exactly $\widehat{\mathbb{P}}(r_t) = \mathbb{P}(r_{t+1} \,|\, \mathtt{pt}_t)$. By building the equivalence of pretrained LLM and BMA, we have the following proposition, which shows that predicting $\{r_t\}_{t\in[T]}$ by iteratively prompting the LLM incurs a $\mathcal{O}(1/T)$ average error.

Proposition 3.4 (ICL Average Error of Perfectly Pretrained Model)**.** Under Assumptions 3.1 and 3.2, we have for any $t \in [T]$ that

$$\frac{1}{t}\sum_{i=1}^{t} \log \mathbb{P}_\theta(r_i \,|\, \mathtt{pt}_{i-1}) \geq \sup_{z\in\mathcal{Z}} \Big(\frac{1}{t}\sum_{i=1}^{t} \log \mathbb{P}(r_i \,|\, \mathtt{pt}_{i-1}, z) + \frac{\log \mathbb{P}_{\mathcal{Z}}(z)}{t} \Big).$$

Here $\mathbb{P}_{\mathcal{Z}}$ is the prior of the hidden concept $z \in \mathcal{Z}$. When the hidden concept space $\mathfrak{Z}$ is finite and the prior $\mathbb{P}_{\mathcal{Z}}(z)$ is the uniform distribution on $\mathfrak{Z}$, we have that $\mathtt{AE}_t \leq \log|\mathfrak{Z}|/t$. When the nominal concept z_* satisfies that $\sup_z \sum_{i=1}^t \log \mathbb{P}(r_i \,|\, z, \mathtt{pt}_{i-1}) = \sum_{i=1}^t \log \mathbb{P}(r_i \,|\, z_*, \mathtt{pt}_{i-1})$ for any $t \in [T]$, the average error is bounded as $\mathtt{AE}_t \leq \log(1/\mathbb{P}_{\mathcal{Z}}(z_*))/t$.

This theorem states that the ICL average error of the perfectly pretrained model is bounded by $\log(1/\mathbb{P}_{\mathcal{Z}}(z_*))/t$. This is intuitive since the average error is relatively large if the concept z_* rarely appears according to the prior distribution. This proposition shows that, when given sufficiently many examples, predicting $\{r_t\}_{t\in[T]}$ via ICL is almost as good as the oracle method which knows true concept z_* and the likelihood function $\mathbb{P}(r_i \,|\, \mathtt{pt}_{i-1}, z_*)$. We state the result for the case where $\mathfrak{Z}$ is finite, and these results can be generalized to uncountable $\mathfrak{Z}$ with continuity assumptions. The proof of Proposition 3.4 and the extension is in Appendix 5.3. In Section 4, we characterize the deviation between the learned model and the underlying true model.

3.2 Validations for Propositions 3.3 and 3.4

To verify Propostion 3.3, we empirically construct the hidden concept and condition on it for inference. We construct the hidden concept vector as the average sum over prompts of the values of twenty selected attention heads, i.e., we compress the hidden concept into a 4096−dimension vector. To demonstrate the effectiveness of the constructed hidden concepts, we add these hidden concept vectors at a layer of LLMs when the model resolves the prompt with zero-shot. In Figure. 3, "zs-hc" refers to the results of LLMs that infers with learned hidden concept vectors and zero-shot prompt, and "ICL-i" refers to the results of LLMs prompted with i examples. We consider the tasks of finding antonyms, the capitals of countries, and the past tense of words, i.e., $h_t = z_*$ in Proposition 3.3. The results indicate that the LLMs conditioned on the learned hidden concept vectors $P_\theta(r_{t+1}|\widetilde{c}_{t+1}, z_*)$ have comparable performance with the LLMs prompted with several examples $P_\theta(r_{t+1}|\mathtt{pt}_t)$. This indicates that the learned hidden concept vectors are indeed efficient compression of the hidden concepts, which proves that LLMs deduce hidden concepts for ICL and corroborates with Proposition 3.3.

To verify Proposition 3.4, we will empirically show that $t\times\mathtt{AE}_t$, i.e., the cumulative error, is upper bounded by a constant. LLMs is trained for the linear regression task from scratch, which is a representative setting studied in Garg et al. (2022); Akyürek et al. (2022). The examples in the prompt are $\{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R}^d$, $d = 20$ and $y_i = w^T x_i$ for some w sampled from Gaussian distribution. Given the Gaussian model, we adopt the squared error to approximate the logarithm of the probability. Then $t \times \mathtt{AE}_t$ of the LLMs can be well approximated by the sum of the squared error till time t. We note that there is pretraining error in our experiments, which can be bounded by Theorems 4.3 and 5.2. The results in Figure 2 strongly corroborate our theoretical findings. First, the results verify our claim in Proposition 3.4 that $t\cdot\mathtt{AE}_t$ can be upper bounded by a constant. Second, the line of squared error indicates that the ICL of LLMs only has a significant error when $T \leq d$, i.e., the cumulative error only increases in this region. Thus, the cumulative error of the ICL by LLMs is at most linear in $O(d/T)$. From the view of our theoretical result, discretizing the set $\{z \in \mathbb{R}^d \,|\, \|z\|_2 \leq d\}$ with approximation error $\delta > 0$ will result in a set with $(C/\delta)^d$ elements, where $C > 0$ is an absolute constant. Proposition 3.4 implies that the cumulative error is the sum of the $\log|\mathfrak{Z}|/T = d\log(C/\delta)/T$ and the pretraining error, which matches the simulation results. The experiment details can be found in Appendix 4.

3.3 Attention Parameterizes BMA

In the following, we explore the role played by the attention mechanism in ICL. To simplify the presentation, we consider the case where the covariate $\widetilde{c}_t \in \mathfrak{X}^*$ is a single token $c_t \in \mathfrak{X}$ in this subsection. During the ICL phase, pretrained LLMs are prompted with $\mathtt{pt}_t = (S_t, c_{t+1})$ and tasked with predicting the $(t+1)$-th response r_{t+1}. The transformers first separately map the covariates $\widetilde{c}_i$ and responses r_i for $i \in [t]$ to the corresponding feature spaces, which are usually realized by the fully connected layers. We denote these two learnable mappings as $k : \mathbb{R}^d \to \mathbb{R}^{d_k}$ and $v : \mathbb{R}^d \to \mathbb{R}^{d_v}$. Their nominal values are denoted as k_* and v_*, respectively. The pretraining of the transformer essen-

tially learns the nominal mappings v_* and k_* with sufficiently many data points. After these transformations, the attention module will take $v_i = v_*(r_i)$ and $k_i = k_*(c_i)$ for $i \in [t]$ as the value and key vectors to predict the result for the query $q_{t+1} = k_{t+1} = k_*(c_{t+1})$. To elucidate the role played by attention, we consider a Gaussian linear simplification of (3.2).

Assumption 3.5 (Gaussian linear ICL model)**.** For the features $v_t = v_*(r_t)$ and $k_t = k_*(c_t)$ for $t \in [t]$, we have

$$v_t = z_*\phi(k_t) + \xi_t, \quad \forall t \in [T], \tag{3.5}$$

where $\phi : \mathbb{R}^{d_k} \to \mathbb{R}^{d_\phi}$ refers to the feature mapping in some Reproducing Kernel Hilbert Space (RKHS), $z_* \in \mathbb{R}^{d_v \times d_\phi}$ corresponds to the hidden concept, and $\xi_t \sim N(0, \sigma^2 I), t \in [T]$ are i.i.d. Gaussian noises with covariance $\sigma^2 I$. The prior of z_* is $\mathbb{P}(z)$ is a Gaussian distribution $N(0, \lambda I)$.

The function in (3.5) is general. The generality comes from: (i) the feature mapping ϕ and the corresponding RKHS make Kernel Mean Embedding (KME) have sufficient expressiveness since all the operations on distribution can be captured by some operations in KME space. (ii) Two learnable mappings $v_*(\cdot)$ and $k_*(\cdot)$ are neural networks and are general enough to represent continuous functions. To specify the relationship between Assumptions 3.1 and 3.5, note that (3.5) can be written as

$$r_t = v_*^{-1}\big(z_*\phi\big(k_*(c_t)\big) + \xi_t\big) \tag{3.6}$$

if v_*^{-1} is reversible, which is a realization of (3.2) with $h_t = z$, $\xi_t = \epsilon_t$, and $f(c, h, \xi) = v_*^{-1}(h\phi(k_*(c)) + \xi)$. In other words, (3.5), or equivalently (3.6), specifies a specialization of (3.2) where In other words, (3.5), or equivalently (3.6), specifies a specialization of (3.2) where in the feature space, the hidden concept z_* represents a transformation between the value v and the key k. Here, we simply take this as the transformation by a matrix, which can be easily generalized by building a bijection between concepts z and complex transformations. In the following, to simplify the notation, let $\mathfrak{K} : \mathbb{R}^{d_k} \times \mathbb{R}^{d_k} \to \mathbb{R}$. denote the kernel function of the RKHS induced by ϕ. The stacks of the values and keys are denoted as $K_t = (k_1, \ldots, k_t)^\top \in \mathbb{R}^{t \times d_k}$ and $V_t = (v_1, \ldots, v_t)^\top \in \mathbb{R}^{t \times d_v}$, respectively. Consequently, the model in (3.5) implies that

$$\begin{aligned}\mathbb{P}(v_{t+1} \mid \mathtt{pt}_t) &= \int \mathbb{P}(v_{t+1} \mid z, q_{t+1})\mathbb{P}(z \mid S_t)\mathrm{d}z \\ &\propto \exp\big(-\big\|v_{t+1} - \bar{z}_t\phi(q_{t+1})\big\|^2_{\Sigma_t^{-1}}/2\big),\end{aligned} \tag{3.7}$$

where we denote by Σ_t the covariance of $v_{t+1} \sim \mathbb{P}(\cdot \mid S_t, q_{t+1})$, and the mean concept $\bar{z}_t$ is

$$\bar{z}_t = V_t\big(\mathfrak{K}(K_t, K_t) + \lambda I\big)^{-1}\phi(K_t). \tag{3.8}$$

Combining (3.7) and (3.8), we can see that $\bar{z}_t\phi(q_{t+1})$ essentially measures the similarity between the query and keys, which is quite similar to the attention mechanism defined in Section 2. However, here the similarity is normalized according to (3.8), not by softmax. This motivates us to define a new structure of attention and explore the relationship between the newly defined attention and the original one. For any $q \in \mathbb{R}^{d_k}$, $K \in \mathbb{R}^{t \times d_k}$, and $V \in \mathbb{R}^{t \times d_v}$, we define a variant of the attention mechanism as follows,

$$\mathtt{attn}_\dagger(q, K, V) = V^\top\big(\mathfrak{K}(K, K) + \lambda I\big)^{-1}\mathfrak{K}(K, q). \tag{3.9}$$

From (3.7), (3.8), and (3.9), it holds that the response v_{t+1} for $(t+1)$-th query is distributed as $v_{t+1} \sim N(\mathtt{attn}_\dagger(q_{t+1}, K_t, V_t), \Sigma_t)$. We note that $\mathtt{attn}_\dagger$ **bakes the BMA algorithm** for the Gaussian linear model **in its architecture**, by first estimating $\bar{z}_t$ via (3.8) and deriving the final estimate from the inner product between $\bar{z}_t$ and q_{t+1}. Here $\mathtt{attn}_\dagger(\cdot)$ is an instance of the *intention mechanism* studied in Garnelo and Czarnecki (2023) and can be viewed as a generalization of linear attention. In the following proposition, we show that the attention in (3.9) coincides with the softmax attention $\mathtt{attn}(q, K, V) = V^\top\mathtt{softmax}(Kq)$ for q, K and V as the sequence length goes to infinity.

Proposition 3.6. We assume that Assumption 3.5 holds for the feature mapping ϕ of Gaussian RBF kernel $\mathfrak{K}_{\mathtt{RBF}}$. In addition, we assume that $\|k_t\| = \|v_t\| = 1$. Then, it holds for a constant $C > 0$ that depends on d_k and any $q \in \mathbb{R}^{d_k}$ with $\|q\| = 1$ that $\lim_{T\to\infty} \mathtt{attn}_\dagger(q, K_T, V_T) = C \cdot \lim_{T\to\infty} \mathtt{attn}(q, K_T, V_T)$.

The proof is in Appendix 5.4. Combined with the conditional probability of v_{t+1} in (3.7), this proposition shows that **softmax attention approximately encodes BMA** in long token sequences (Wasserman, 2000), and thus is able to perform ICL when prompted after pretraining. This proposition also implies that the output of the attention module contains the information of the hidden concept z_*, which will be verified in experiments.

3.4 Validations For Proposition 3.6

We conduct two experiments to verify Proposition 3.6. The first experiment is same as the hidden concept vectors construction experiments in Section 3.2. Proposition 3.6 and (3.8) imply that the heads of attention

contain the hidden concept information. Thus, we construct the hidden concept vector as the average of the values of twenty selected attention heads. The effectiveness of the constructed hidden concept is demonstrated in Figure 3, which strongly corroborates with (3.8).

In the second experiment, we directly calculate the ratio between $\texttt{attn}_\dagger$ and $\texttt{attn}$. We consider the case $d_v = 1$ and $d_k = d$ for some $d > 0$. The entries in K of (3.9) are i.i.d. samples of Gaussian distribution, and the $i-$th entry of V is calculated as the sum of noise and the inner product between a Gaussian vector and the $i-$th row of K. Figure 4 shows the results for $d = 2, 3$. It shows that the ratio between $\texttt{attn}_\dagger$ and $\texttt{attn}$ will converge to a constant. This constant depends on the dimension d, which originates from Proposition 5.1.

4 Theoretical Analysis of Pretraining

4.1 Pretraining Algorithm

In this section, we describe the pretraining setting. We largely follow the transformer structures in Brown et al. (2020). The whole network is a composition of D sub-modules, and each sub-module consists of a MHA and a Feed-Forward (FF) fully connected layer. Here, $D > 0$ is the depth of the network. The whole network takes $X^{(0)} = X \in \mathbb{R}^{L\times d}$ as its input. In the t-th layer for $t \in [D]$, it first takes the output $X^{(t-1)}$ of the $(t-1)$-th layer as the input and forwards it through MHA with a residual link and a layer normalization $\Pi_{\rm norm}(\cdot)$ to output $Y^{(t)}$, which projects each row of the input into the unit ℓ_2-ball. Here we take $d_h = d$ in MHA, and the generalization of our result to general cases is trivial. Then the intermediate output $Y^{(t)}$ is forwarded to the FF module. It maps each row of the input $Y^{(t)} \in \mathbb{R}^{L\times d}$ through the same single-hidden layer neural network with d_F neurons, that is $\texttt{ffn}(Y^{(t)}, A^{(t)}) = \texttt{ReLU}(Y^{(t)}A_1^{(t)})A_2^{(t)}$, where $A_1^{(t)} \in \mathbb{R}^{d\times d_F}$, and $A_2^{(t)} \in \mathbb{R}^{d_F\times d}$ are the weight matrices. Combined with a residual link and layer normalization, it outputs the output of layer t as $X^{(t)}$, i.e.,

$$\begin{aligned} Y^{(t)} &= \Pi_{\rm norm}\big[\texttt{mha}(X^{(t-1)}, W^{(t)}) + \gamma_1^{(t)} X^{(t-1)}\big], \\ X^{(t)} &= \Pi_{\rm norm}\big[\texttt{ffn}(Y^{(t)}, A^{(t)}) + \gamma_2^{(t)} Y^{(t)}\big]. \end{aligned} \tag{4.1}$$

Here we allocate weights $\gamma_1^{(t)}$ and $\gamma_2^{(t)}$ to residual links only for the convenience of theoretical analysis. In the last layer, the network outputs the probability of the next token via a softmax module, that is $Y^{(D+1)} = \texttt{softmax}(\mathbb{I}_L^\top X^{(D)} A^{(D+1)}/(L\tau)) \in \mathbb{R}^{d_y}$, where $\mathbb{I}_L \in \mathbb{R}^L$ is the vector with all ones, $A^{(D+1)} \in \mathbb{R}^{d\times d_y}$ is the weight matrix, $\tau \in (0, 1]$ is the fixed temperature parameter, and d_y is the output dimension. The parameters of each layer are denoted as $\theta^{(t)} = (\gamma_1^{(t)}, \gamma_2^{(t)}, W^{(t)}, A^{(t)})$ for $t \in [D]$ and $\theta^{(D+1)} = A^{(D+1)}$, and the parameter of the whole network is the concatenation of these parameters, i.e., $\theta = (\theta^{(1)}, \cdots, \theta^{(D+1)})$. We consider the transformers with bounded parameters. The set of parameters is

$$\begin{aligned} \Theta = \Big\{\theta \,|\, &\big\|A^{(D+1),\top}\big\|_{1,2} \le B_A, \max\big\{\big|\gamma_1^{(t)}\big|, \big|\gamma_2^{(t)}\big|\big\} \le 1, \\ &\big\|A_1^{(t)}\big\|_{\mathbf{F}} \le B_{A,1}, \big\|A_2^{(t)}\big\|_{\mathbf{F}} \le B_{A,2}, \big\|W_i^{Q,(t)}\big\|_{\mathbf{F}} \le B_Q, \\ &\big\|W_i^{K,(t)}\big\|_{\mathbf{F}} \le B_K, \big\|W_i^{V,(t)}\big\|_{\mathbf{F}} \le B_V, t \in [D], i \in [h]\Big\}, \end{aligned}$$

where B_A, $B_{A,1}$, $B_{A,2}$, B_Q, B_K, and B_V are the bounds of parameter. The probability induced by the transformer with parameter θ is denoted as $\mathbb{P}_\theta$.

The pretraining dataset consists of $N_{\rm p}$ independent trajectories. For the n-th trajectory with $n \in [N_{\rm p}]$, a hidden concept $z^n \sim \mathbb{P}_{\mathcal{Z}}(z) \in \Delta(\mathfrak{Z})$ is first sampled, which is the hidden concept of the token sequence to generate.Then the tokens are sequentially sampled from the model in (3.2). We view this model as a Markov chain in the *sequence space* $\mathfrak{X}^*$ induced by z^n, i.e., $x_{t+1}^n \sim \mathbb{P}(\cdot \,|\, S_t^n, z^n)$ and $S_{t+1}^n = (S_t^n, x_{t+1}^n)$, where $x_{t+1}^n \in \mathfrak{X}$ can be either r_t or $\widetilde{c}_t$ in (3.2), and $S_t^n, S_{t+1}^n \in \mathfrak{X}^*$. This Markov chain is defined with respect to the state S_t^n, which obviously satisfies the Markov property since S_i^n for $i \in [t-1]$ are contained in S_t^n. The pretraining dataset is $\mathcal{D}_{N_{\rm p},T_{\rm p}} = \{(S_t^n, x_{t+1}^n)\}_{n,t=1}^{N_{\rm p},T_{\rm p}}$ where the concepts z^n is hidden from the context and thus unobserved. Here each token sequence is divided into $T_{\rm p}$ pieces $\{(S_t^n, x_{t+1}^n)\}_{t=1}^{T_{\rm p}}$. We highlight that this pretraining dataset collecting process subsumes those for GPT. For GPT, each trajectory corresponds to a paragraph or an article in the pretraining dataset, and $z^n \sim \mathbb{P}_{\mathcal{Z}}(z)$ is realized by the selection process of these contexts from the Internet.

To pretrain the transformer, we adopt the cross-entropy as the loss function. The pretraining algorithm is

$$\widehat{\theta} = \operatorname*{argmin}_{\theta\in\Theta} -\frac{1}{N_{\rm p}T_{\rm p}} \sum_{n=1}^{N_{\rm p}} \sum_{t=1}^{T_{\rm p}} \log \mathbb{P}_\theta(x_{t+1}^n \,|\, S_t^n). \tag{4.2}$$

We first analyze the population version of (4.2). In the training dataset, the conditional distribution of x_{t+1}^n conditioned on S_t^n is $\mathbb{P}(x_{t+1}^n \,|\, S_t^n) = \int_{\mathfrak{Z}} \mathbb{P}(x_{t+1}^n \,|\, S_t^n, z)\mathbb{P}_{\mathcal{Z}}(z \,|\, S_t^n)\mathrm{d}z$, where the unobserved hidden concept is weighed via its posterior distribution. Thus, the population risk of (4.2) is $\mathbb{E}_t[\mathbb{E}_{S_t}[\texttt{KL}(\mathbb{P}(\cdot \,|\, S_t)\|\mathbb{P}_\theta(\cdot \,|\, S_t)) + H(\mathbb{P}(\cdot \,|\, S_t))]]$, where $t \sim \texttt{Unif}([T_{\rm p}])$, $H(p) = -\langle p, \log p\rangle$ is the entropy, and S_t

is distributed as the pertaining distribution. Thus, we expect that $\mathbb{P}_\theta$ will converge to $\mathbb{P}$.

4.2 Performance Guarantee for Pretraining

We first state the assumptions for the pretraining.

Assumption 4.1. There exists a constant $R > 0$ such that for any $z \in \mathfrak{Z}$ and $S_t \sim \mathbb{P}(\cdot \mid z)$, we have $\|S_t^\top\|_{2,\infty} \le R$ almost surely.

This assumption states that the ℓ_2-norm of each token in the token sequence is upper bounded by $R > 0$. It is satisfied in real applications, where the token is a finite-dimensional vector with bounded components. For example, the tokenizers used in GPT-NeoX (Black et al., 2022) and Llama2 (Touvron et al., 2023) both satisfy this assumption.

Assumption 4.2. There exists a constant $c_0 > 0$ such that for any $z \in \mathfrak{Z}$, $x \in \mathfrak{X}$ and $S \in \mathfrak{X}^*$, $\mathbb{P}(x \mid S, z) \ge c_0$.

This assumption states that the conditional probability of x conditioned on S and z is lower bounded. This comes from the ambiguity of language, that is, a sentence can take lots of words as its next word. It holds in a wide range of problems, which is verified by the success of LLMs. Concretely, the transformers with finite width, depth, and weights can only model the distribution that satisfies Assumption 4.2., since the last softmax layer of the transformer renders the probability of each token strictly larger than 0. Since transformers have successfully learned a large range of tasks, it is reasonable to assume that those distributions satisfy this assumption. Similar regularity assumptions are also widely adopted in ICL literature (Xie et al., 2021; Wies et al., 2023). The parameter c_0 depends on both the hidden concept set $\mathfrak{Z}$ and the prompt. To state our result, we respectively use $\mathbb{E}_{S\sim\mathcal{D}}$ and $\mathbb{P}_{\mathcal{D}}$ to denote the expectation and the distribution of the average distribution of S_t^n in $\mathcal{D}_{N_\mathrm{p},T_\mathrm{p}}$, i.e., $\mathbb{E}_{S\sim\mathcal{D}}[f(S)] = \sum_{t=1}^{T_\mathrm{p}} \mathbb{E}_{S_t}[f(S_t)]/T_\mathrm{p}$ for any function $f : \mathfrak{X}^* \to \mathbb{R}$.

Theorem 4.3. Let $\bar{B} = RhB_AB_{A,1}B_{A,2}B_QB_KB_V/\tau$ and $\bar{D} = D^2d(d_F + d_h + d) + d \cdot d_y$. Under Assumptions 4.1 and 4.2, the pretrained model $\mathbb{P}_{\widehat{\theta}}$ by the algorithm in (4.2) satisfies

$$\begin{aligned}
&\mathbb{E}_{S\sim\mathcal{D}}\Big[\mathtt{TV}\big(\mathbb{P}(\cdot \mid S), \mathbb{P}_{\widehat{\theta}}(\cdot \mid S)\big)\Big] \\
&= O\Bigg(\underbrace{\inf_{\theta^*\in\Theta}\sqrt{\mathbb{E}_{S\sim\mathcal{D}}\mathtt{KL}\big(\mathbb{P}(\cdot|S)\|\mathbb{P}_{\theta^*}(\cdot|S)\big)} + \frac{\sqrt{b^*}t_{\mathrm{mix}}^{1/4}\log 1/\delta}{(N_\mathrm{p}T_\mathrm{p})^{1/4}}}_{\text{approximation error}} \\
&\quad + \underbrace{\frac{\sqrt{t_{\mathrm{mix}}}}{\sqrt{N_\mathrm{p}T_\mathrm{p}}}\Big(\bar{D}\log(1+N_\mathrm{p}T_\mathrm{p}\bar{B}) + \log\frac{1}{\delta}\Big)}_{\text{generalization error}}\Bigg)
\end{aligned}$$

with probability at least $1 - \delta$, where $b^* = \log(\max\{c_0^{-1}, 1 + d_y\exp(B_A/\tau)\})$, and t_{mix} is the mixing time of the Markov chains induced by $\mathbb{P}$, formally defined in Appendix 6.1.

We define the right-hand side of the equation as $\Delta_{\mathrm{pre}}(N_\mathrm{p}, T_\mathrm{p}, \delta)$. The first and the second terms in the bound are the **approximation error**. It measures the distance between the nominal distribution $\mathbb{P}$ and the distributions induced by transformers with respect to KL divergence. If the nominal model $\mathbb{P}$ can be represented by transformers exactly, i.e., the realizable case, these two terms will vanish. The third term is the **generalization error**, and it does not increase with the growing sequence length T_p. This is proved via the PAC-Bayes framework.

The pretraining analysis is missing in most existing theoretical works about ICL. Xie et al. (2021), Wies et al. (2023), and Jiang (2023) all assume access to an arbitrarily precise pretraining model. Although the generalization bound in Li et al. (2023a) can be adapted to the pretraining analysis, the risk definition therein can not capture the approximation error in our result. Furthermore, their analysis cannot fit the maximum likelihood algorithm in (4.2). Concretely, their result can only show that the convergence rate of KL divergence is $O((N_\mathrm{p}T_\mathrm{p})^{-1/2})$ with a realizable function class. Combined with Pinsker's inequality, this gives the convergence rate for total variation as $O((N_\mathrm{p}T_\mathrm{p})^{-1/4})$ even in the realizable case.

5 ICL Error under Practical Settings

In Section 3, we study the ICL average error with a perfect pretrained model. In what follows, we characterize the ICL average error when the pretrained model has an error. Note that the distribution $\mathcal{D}_{\mathtt{ICL}}$ of the prompts of ICL tasks can be different from that of pretraining. We impose the following assumption on their relation.

Assumption 5.1. We assume that there exists an absolute constant $\kappa > 0$ such that for any ICL prompt $\mathtt{pt} \in \mathfrak{X}^*$ with length less and equal to T_p, it holds that $\mathbb{P}_{\mathcal{D}_{\mathtt{ICL}}}(\mathtt{pt}) \le \kappa \cdot \mathbb{P}_{\mathcal{D}}(\mathtt{pt})$.

This assumption states the coverage of the prompt distribution by the pretraining distribution. We note that there will be an information-theoretic barrier without this assumption. For example, if the pretraining data does not contain any material about a specific mathematical symbol in the ICL prompt. In this case, it will be extremely difficult for the LLM to derive the correct prediction, since the meaning of this math symbol is unclear to LLMs. The parameter κ scales with the size

of the hidden space $\mathfrak{Z}$ and the length of the prompt. In the worst case, $\kappa = O(|\mathfrak{Z}| * T_{\mathrm{p}})$. Usually, we note that N_{p} is substantially larger than κ. Concretely, in the pretraining data generation process, each hidden concept will generate a large number of sentences. The real-world data set contains a large number of sentences explaining the same concept, e.g., the data from Wikipedia. We then have the following theorem characterizing the ICL average error of the pretrained model.

Theorem 5.2 (ICL average error of Pretrained Model)**.** We assume that the underlying hidden concept z_* maximizes $\sum_{i=1}^{t} \log \mathbb{P}(r_i \,|\, \mathtt{pt}_{i-1}, z)$ for any $t \in [T]$ ($T \leq T_{\mathrm{p}}$) and there exists an absolute constant $\beta > 0$ such that $\log(1/p_0(z_*)) \leq \beta$. Under Assumptions 3.1, 4.1, 4.2, and 5.1, we have with probability at least $1 - \delta$ that

$$T^{-1} \sum_{t=1}^{T} \mathbb{E}_{\mathtt{pt} \sim \mathcal{D}_{\mathtt{ICL}}} \Big[\log \mathbb{P}(r_t \,|\, z_*, \mathtt{pt}_{t-1}) - \log \mathbb{P}_{\widehat{\theta}}(r_t \,|\, \mathtt{pt}_{t-1}) \Big] \leq \mathcal{O}\big(\beta/T + \kappa \cdot b^* \cdot \Delta_{\mathrm{pre}}(N_{\mathrm{p}}, T_{\mathrm{p}}, \delta)\big),$$

where we denote by $\Delta_{\mathrm{pre}}(N_{\mathrm{p}}, T_{\mathrm{p}}, \delta)$ the pretraining error in Theorem 4.3, and $b^* = \log \max\{c_0^{-1}, 1 + d_y \exp(B_A/\tau)\}$.

We note that d_y and B_A are the parameters of transformers defined in Section 4.1. The requirement $T \leq T_{\mathrm{p}}$ originates from that the pertaining process can only guarantee the performance for prompt not longer than T_{p}. Theorem 5.2 shows that the expected ICL average error for the pretrained model is upper bounded by the sum of two terms: **(a) the ICL average error for the underlying true model** and **(b) the pretraining error**. These two terms are separately bounded in Sections 3 and 4.

Till now, we focus on the situations where the examples in the prompt is generated by the true distribution. However, it has been shown that LLMs are still able to implement ICL when the examples contain some *errors* (Min et al., 2021).

6 Conclusion

In this paper, we investigated the theoretical foundations of ICL for the pretrained language models. We proved that the perfectly pretrained LLMs implicitly implements BMA with regret $\mathcal{O}(1/t)$ over a general response generation modeling, which subsumes the models in previous works. Based on this, we showed that the attention mechanism parameterizes the BMA algorithm. Analyzing the pretraining process, we demonstrated that the total variation between the pretrained model and the nominal distribution consists of the approximation error and the generalization error. The combination of the ICL regret and the pretraining performance gives the full description of ICL ability of pretrained LLMs. We mainly focus on the prompts that comprise several examples in this work and leave the analysis of instruction-based prompts for future works.

Acknowledgements

Zhaoran Wang acknoledges National Science Foundation (Awards 2235451, 2225087, 2211210, CAREER-2048075, 2015568, 2008827, 1934931/2216970), Simons Institute (Theory of Reinforcement Learning), Amazon, J.P. Morgan, Two Sigma, Tencent. Zhuoran Yang acknoledges support from NSF DMS 2413243.

References

Agarwal, A., Kakade, S., Krishnamurthy, A., and Sun, W. (2020). Flambe: Structural complexity and representation learning of low rank MDPs. *Advances in Neural Information Processing Systems*, 33:20095–20107.

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

Anthony, M., Bartlett, P. L., Bartlett, P. L., et al. (1999). *Neural network learning: Theoretical foundations*, volume 9. cambridge university press Cambridge.

Bai, Y., Chen, F., Wang, H., Xiong, C., and Mei, S. (2023). Transformers as statisticians: Provable in-context learning with in-context algorithm selection. *arXiv preprint arXiv:2306.04637*.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017). Spectrally-normalized margin bounds for neural networks. *Neural Information Processing Systems*.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. (2018). Mutual information neural estimation. In *International Conference on Machine Learning*, pages 531–540. PMLR.

Black, S., Biderman, S., Hallahan, ., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., et al. (2022). Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Neural Information Processing Systems*.

Caponnetto, A. and De Vito, E. (2007). Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7:331–368.

Chan, S. C., Santoro, A., Lampinen, A. K., Wang, J. X., Singh, A., Richemond, P. H., McClelland, J., and Hill, F. (2022). Data distributional properties drive emergent few-shot learning in transformers. *arXiv preprint arXiv:2205.05055*.

Dai, D., Sun, Y., Dong, L., Hao, Y., Sui, Z., and Wei, F. (2022). Why can GPT learn In-Context? Language models secretly perform gradient descent as meta optimizers. *arXiv preprint arXiv:2212.10559*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., and Hon, H.-W. (2019). Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. (2022). A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Duchi, J. C. (2019). Information theory and statistics. *Lecture Notes for Statistics*, 311:304.

Edelman, B. L., Goel, S., Kakade, S., and Zhang, C. (2021). Inductive biases and variable creation in self-attention mechanisms. *arXiv preprint arXiv:2110.10090*.

Elbrächter, D., Perekrestenko, D., Grohs, P., and Bölcskei, H. (2021). Deep neural network approximation theory. *IEEE Transactions on Information Theory*, 67(5):2581–2623.

Elman, J. L. (1995). Language as a dynamical system. *Mind as motion: Explorations in the dynamics of cognition*, pages 195–223.

Falck, F., Wang, Z., and Holmes, C. (2024). Is in-context learning in large language models bayesian? a martingale perspective. *arXiv preprint arXiv:2406.00793*.

Feng, G., Gu, Y., Zhang, B., Ye, H., He, D., and Wang, L. (2023). Towards revealing the mystery behind chain of thought: a theoretical perspective. *arXiv preprint arXiv:2305.15408*.

Fukumizu, K. (2015). Nonparametric bayesian inference with kernel mean embedding. In *Modern Methodology and Applications in Spatial-Temporal Modeling*, pages 1–24. Springer.

Garg, S., Tsipras, D., Liang, P., and Valiant, G. (2022). What can transformers learn in-context? A case study of simple function classes. *arXiv preprint arXiv:2208.01066*.

Garnelo, M. and Czarnecki, W. M. (2023). Exploring the space of key-value-query models with intention. *arXiv preprint arXiv:2305.10203*.

Hahn, M. and Goyal, N. (2023). A theory of emergent in-context learning as implicit structure induction. *arXiv preprint arXiv:2303.07971*.

Honovich, O., Shaham, U., Bowman, S. R., and Levy, O. (2022). Instruction induction: From few examples to natural language task descriptions. *arXiv preprint arXiv:2205.10782*.

Hron, J., Bahri, Y., Sohl-Dickstein, J., and Novak, R. (2020). Infinite attention: NNGP and NTK for deep attention networks. In *International Conference on Machine Learning*.

Iyer, S., Lin, X. V., Pasunuru, R., Mihaylov, T., Simig, D., Yu, P., Shuster, K., Wang, T., Liu, Q., Koura, P. S., et al. (2022). OPT-IML: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*.

Jeon, H. J., Lee, J. D., Lei, Q., and Van Roy, B. (2024). An information-theoretic analysis of in-context learning. *arXiv preprint arXiv:2401.15530*.

Jiang, H. (2023). A latent space theory for emergent abilities in large language models. *arXiv preprint arXiv:2304.09960*.

Ke, G., He, D., and Liu, T.-Y. (2020). Rethinking positional encoding in language pre-training. *arXiv preprint arXiv:2006.15595*.

Kim, H. J., Cho, H., Kim, J., Kim, T., Yoo, K. M., and Lee, S.-g. (2022). Self-generated in-context learning: Leveraging auto-regressive language models as a demonstration generator. *arXiv preprint arXiv:2206.08082*.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Kossen, J., Gal, Y., and Rainforth, T. (2024). In-context learning learns label relationships but is not conventional learning. In *The Twelfth International Conference on Learning Representations*.

Ledent, A., Mustafa, W., Lei, Y., and Kloft, M. (2021). Norm-based generalisation bounds for deep multi-class convolutional neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8279–8287.

Li, Y., Ildiz, M. E., Papailiopoulos, D., and Oymak, S. (2023a). Transformers as algorithms: Generalization and stability in in-context learning. *arXiv preprint arXiv:2301.07067*.

Li, Y., Li, Y., and Risteski, A. (2023b). How do transformers learn topic structure: Towards a mechanistic understanding. *arXiv preprint arXiv:2303.04245*.

Liao, R., Urtasun, R., and Zemel, R. (2020). A pac-bayesian approach to generalization bounds for graph neural networks. *arXiv preprint arXiv:2012.07690*.

Lin, S. and Zhang, J. (2019). Generalization bounds for convolutional neural networks. *arXiv preprint arXiv:1910.01487*.

Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., and Chen, W. (2021). What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Lu, Y., Bartolo, M., Moore, A., Riedel, S., and Stenetorp, P. (2021). Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Malladi, S., Wettig, A., Yu, D., Chen, D., and Arora, S. (2022). A kernel-based view of language model fine-tuning. *arXiv preprint arXiv:2210.05643*.

Michel, P., Levy, O., and Neubig, G. (2019). Are sixteen heads really better than one? *Advances in neural information processing systems*, 32.

Mikulik, V., Delétang, G., McGrath, T., Genewein, T., Martic, M., Legg, S., and Ortega, P. (2020). Meta-trained agents implement bayes-optimal agents. *Advances in neural information processing systems*, 33:18691–18703.

Min, S., Lewis, M., Zettlemoyer, L., and Hajishirzi, H. (2021). Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*.

Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., and Zettlemoyer, L. (2022). Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. (2017). A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*.

Niyogi, P., Berwick, R. C., et al. (1997). A dynamical systems model for language change. *Complex Systems*, 11(3):161–204.

Noci, L., Anagnostidis, S., Biggio, L., Orvieto, A., Singh, S. P., and Lucchi, A. (2022). Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. *arXiv preprint arXiv:2206.03126*.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Paulin, D. (2015). Concentration inequalities for markov chains by marton couplings and spectral methods. *Electronic Journal of Probability*.

Rubin, O., Herzig, J., and Berant, J. (2021). Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*.

Song, L., Huang, J., Smola, A., and Fukumizu, K. (2009). Hilbert space embeddings of conditional distributions with applications to dynamical systems. In *International Conference on Machine Learning*.

Todd, E., Li, M. L., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. (2023). Function vectors in large language models. *arXiv preprint arXiv:2310.15213*.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Neural Information Processing Systems*.

von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A.,

and Vladymyrov, M. (2022). Transformers learn in-context by gradient descent. *arXiv preprint arXiv:2212.07677*.

Vuckovic, J., Baratin, A., and Combes, R. T. d. (2020). A mathematical theory of attention. *arXiv preprint arXiv:2007.02876*.

Wang, X., Zhu, W., and Wang, W. Y. (2023). Large language models are implicitly topic models: Explaining and finding good demonstrations for in-context learning. *arXiv preprint arXiv:2301.11916*.

Wang, Y., Kordi, Y., Mishra, S., Liu, A., Smith, N. A., Khashabi, D., and Hajishirzi, H. (2022). Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Wasserman, L. (2000). Bayesian model selection and model averaging. *Journal of Mathematical Psychology*, 44(1):92–107.

Wei, C., Chen, Y., and Ma, T. (2022a). Statistically meaningful approximation: a case study on approximating turing machines with transformers. *Advances in Neural Information Processing Systems*, 35:12071–12083.

Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. (2021). Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., et al. (2022b). Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., and Zhou, D. (2022c). Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Wies, N., Levine, Y., and Shashua, A. (2023). The learnability of in-context learning. *arXiv preprint arXiv:2303.07895*.

Xie, S. M., Raghunathan, A., Liang, P., and Ma, T. (2021). An explanation of in-context learning as implicit Bayesian inference. *arXiv preprint arXiv:2111.02080*.

Yang, G. (2020). Tensor programs II: Neural tangent kernel for any architecture. *arXiv preprint arXiv:2006.14548*.

Yun, C., Bhojanapalli, S., Rawat, A. S., Reddi, S. J., and Kumar, S. (2019). Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*.

Zhang, F., Liu, B., Wang, K., Tan, V. Y., Yang, Z., and Wang, Z. (2022a). Relational reasoning via set transformers: Provable efficiency and applications to MARL. *arXiv preprint arXiv:2209.09845*.

Zhang, Z., Zhang, A., Li, M., and Smola, A. (2022b). Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Bousquet, O., Le, Q., and Chi, E. (2022a). Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Zhou, Y., Muresanu, A. I., Han, Z., Paster, K., Pitis, S., Chan, H., and Ba, J. (2022b). Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

Checklist

The checklist follows the references. For each question, choose your answer from the three possible options: Yes, No, Not Applicable. You are encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description (1-2 sentences). Please do not modify the questions. Note that the Checklist section does not count towards the page limit. Not including the checklist in the first submission won't result in desk rejection, although in such case we will ask you to upload it during the author response period and include it in camera ready (if accepted).

In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]

(b) Complete proofs of all theoretical results. [Yes]

(c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:

   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   (a) Citations of the creator If your work uses existing assets. [Yes]

   (b) The license information of the assets, if applicable. [Yes]

   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]

   (d) Information about consent from data providers/curators. [Not Applicable]

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Not Applicable]

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials for "What and How does In-Context Learning Learn? Bayesian Model Averaging, Parameterization, and Generalization"

1 Related Work

In-Context Learning. After Brown et al. (2020) showcased the in-context learning (ICL) capacity of GPT-3, there has been a notable surge in interest towards enhancing and comprehending this particular ability (Dong et al., 2022). The ICL ability has seen enhancements through the incorporation of extra training stages (Min et al., 2021; Wei et al., 2021; Iyer et al., 2022), carefully selecting and arranging informative demonstrations (Liu et al., 2021; Kim et al., 2022; Rubin et al., 2021; Lu et al., 2021; Kossen et al., 2024), giving explicit instructions (Honovich et al., 2022; Zhou et al., 2022b; Wang et al., 2022), and prompting a chain of thoughts (Wei et al., 2022c; Zhang et al., 2022b; Zhou et al., 2022a). In efforts to comprehend the mechanisms of ICL ability, researchers have also conducted extensive work. Empirically, Chan et al. (2022) demonstrated that the distributional properties, including the long-tailedness, are important for ICL. Garg et al. (2022) investigated the function class that ICL can approximate. Min et al. (2022) showed that providing wrong mappings between the input-output pairs in examples does not degrade the ICL. Theoretically, Akyürek et al. (2022), von Oswald et al. (2022), Bai et al. (2023), and Dai et al. (2022) indicated that ICL implicitly implements the gradient descent or least-square algorithms from the function approximation perspective. However, the first three works only showed that transformers are able to approximate these two algorithms, which may not align with the pretrained model. The last work ignored the softmax module, which turns out to be important in practical implementation. Feng et al. (2023) derived the impossibility results of ICL and the advantage of chain-of-thought for the function approximation. Li et al. (2023a) viewed ICL from the multi-task learning perspective and derived the generalization bound. Hahn and Goyal (2023) built the linguistic model for sentences and used the description length to bound the ICL error with this model. Jeon et al. (2024) analyzed ICL with the ratio-distortion tools of information theory. Li et al. (2023b) viewed this problem from the optimization perspective and discovered the characteristic of attention. Xie et al. (2021) analyzed ICL within the Bayesian framework, assuming the access to the nominal language distribution and that the tokens are generated from HMMs. However, the first assumption hides the relationship between pretraining and ICL, and the second assumption is restrictive. Following this thread, Wies et al. (2023) relaxed the HMM assumption and assumed access to a pretrained model that is close to the nominal distribution conditioned on any token sequence, which is also unrealistic. Two recent works Wang et al. (2023), and Jiang (2023) also provide the Bayesian analysis of ICL. Unfortunately, these Bayesian works cannot explain the importance of the attention mechanism for ICL and clarify how pretraining is related to ICL. In contrast, we prove that the attention mechanism enables BMA by encoding it in the network architecture and we relate the pretraining error of transformers to the ICL average error. We also note that some work shows that the LLMs do not perform exact Bayesian inference on the exchangeable data (Falck et al., 2024). However, this does not contradict our results, since our data model in Assumption 3.1 is not exchangeable, and the LLMs will not perform exact Bayesian inference due to the pretraining error in Theorem 5.2. In addition to ICL, the Bayesian optimal learning behavior is also studied in the meta-learning problem (Mikulik et al., 2020).

Transformers. Our work is also related to the works that theoretically analyze the performance of transformers. For the analytic properties of transformers, Vuckovic et al. (2020) proved that attention is Lipschitz-continuous via the view of interacting particles. Noci et al. (2022) provided the theoretical justification of the rank collapse phenomenon in transformers. Yun et al. (2019) demonstrated that transformers are universal approximators. For the statistical properties of transformers, Malladi et al. (2022), Hron et al. (2020), and Yang (2020) analyzed the training of transformers within the neural tangent kernel framework. Wei et al. (2022a) presented the approximation and generalization bounds for learning boolean circuits and Turing machines with transformers. Edelman et al. (2021) and Li et al. (2023a) derived the generalization error bound of transformers. In our work, we analyze transformers from both the analytic and statistical sides. We show that attention essentially

implements the BMA algorithm in the ICL setting. Furthermore, we derive the approximation and generalization bounds for transformers in the pretraining phase.

Generalization. Our analysis of the pretraining is also related to the generalization analysis of the neural networks. This topic has attracted a lot of interests for a long time. Anthony et al. (1999) derived the uniform generalization bound for fully-connected neural networks with the help pf VC dimension. Bartlett et al. (2017) sharpened this generalization bound for classification problem by adopting the Dudley's integral and calculating of the covering number of neural network class. At the same time, Neyshabur et al. (2017) derived a similar as Bartlett et al. (2017) from PAC-Bayes framework. Following this line, Liao et al. (2020) , Ledent et al. (2021) and Lin and Zhang (2019) built the generalization bound for graph neural networks and convolutional neural network. These results respected the underlying graph structure and the translation-invariance in the networks. Edelman et al. (2021) established the generalization bound for transformer, but this result did not reflect the permutation-invariance, still depending on the channel number. Our work focuses on the analysis of Maximum Likelihood Estimate (MLE) with transformer function class, which is not covered by previous works. Our bounds are sharper than that of Edelman et al. (2021) on the channel number dependency.

2 Additional Figures

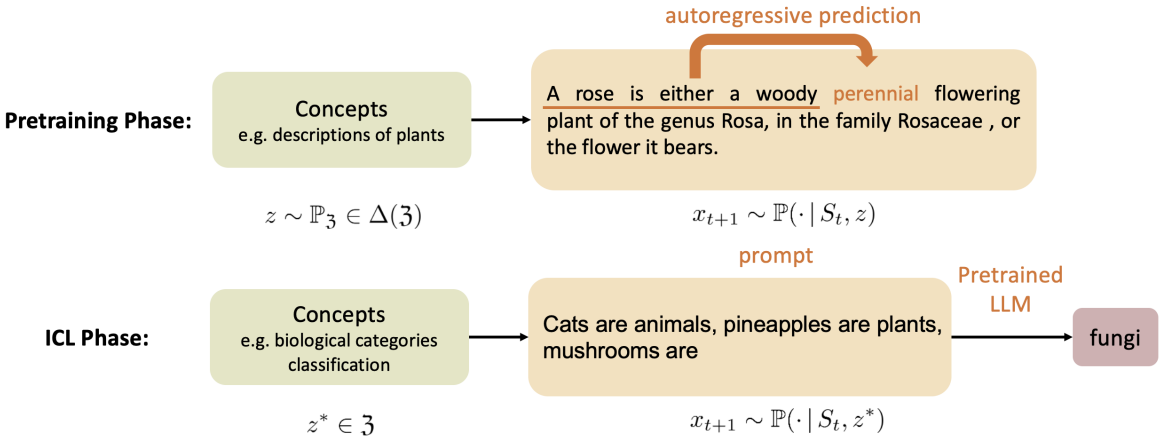

Figure 6: To form the pretraining dataset, a hidden concept z is first sampled according to $\mathbb{P}_3$, and a document is generated from the concept. Taking the token sequence S_t up to position $t \in [T]$ as the input, the LLM is pretrained to maximize the next token x_{t+1}. During the ICL phase, the pretrained LLM is prompted with several examples to predict the response of the query.

3 Notation

We denote $\{1, \cdots, N\}$ as $[N]$. For a Polish space $\mathcal{S}$, we denote the collection of all the probability measures on it as $\Delta(\mathcal{S})$. The total variation distance between two distributions $P, Q \in \Delta(\mathcal{S})$ is $\mathtt{TV}(P, Q) = \sup_{A \subseteq \mathcal{S}} |P(A) - Q(A)|$. The i^{th} entry of a vector x is denoted as x_i or $[x]_i$. For a matrix $X \in \mathbb{R}^{T \times d}$, we index its i^{th} row and column as $X_{i,:}$ and $X_{:,i}$ respectively. The $\ell_{p,q}$ norm of X is defined as $\|X\|_{p,q} = (\sum_{i=1}^{d} \|X_{:,i}\|_p^q)^{1/q}$, and the *Frobenius norm* of it is defined as $\|X\|_{\mathbf{F}} = \|X\|_{2,2}$. We would like to present a notation table here.

Notation	Interpretations	Notation	Interpretations
z, $\mathfrak{Z}$	hidden concept and its space	h_t, $\mathcal{H}$	hidden variable and its space
$\mathfrak{X}$	the space of all the tokens	$\mathfrak{X}^*$	the space of all the token sequences
$\widetilde{c}_t$	the covariate in the ICL examples	r_t	the response in the ICL examples
S_t	the first t ICL examples	$\mathtt{pt}_t$	the prompt with t examples and one query
$\mathbb{P}$	nominal distribution	θ, $\mathbb{P}_\theta$	the parameter of LLM and the distribution induced by it
N_{p}	the number of independent token trajectories in the pretraining dataset	T_{p}	the maximum length of trajectory in the pretraining dataset

4 Implementation Details of Experiments

In this section, we provide the implementation details of the experiments. In the hidden concepts construction experiment, we explicitly calculate the hidden concept vector for Llama2-7b with the method in Todd et al. (2023). Given the prompts generated from the same hidden concept, we calculate the average value of each attention head by prompting the LLM with different prompts. Then we select the attention head according to its average indirect effect, which is defined in Todd et al. (2023). The hidden concept vector is the sum of the average value of the selected attention heads. We test the performance of the learned hidden concept vectors on tasks: (1) Antonym: Given an input word, generate the word with the opposite meaning. (2) Country-Capital. Given a country name, generate the capital city. (3) Present-Past. Given a verb in the present tense, generate the verb's simple past inflection. To test the effectiveness of the learned hidden concept vector, we prompt the LLM only with the query, i.e., the zero-shot case, and set the attention head values at some layer as the learned hidden concept vector.

For the linear regression task, the model is trained with the loss

$$L(f) = \frac{1}{T}\sum_{t=1}^{T}\big(y_t - f(\mathtt{pt}_t)\big)^2,$$

where $\mathtt{pt}_t = (x_1, y_1, \cdots, x_{t-1}, y_{t-1}, x_t)$, $y_t = w^{\mathbf{T}}x_t$, $\{x_t\}_{t=1}^T$ and w are i.i.d. samples of Gaussian distribution (Garg et al., 2022). The model is designed based on GPT-2, and we add linear layers as the first and last layers to accommodate it for the value prediction task. In the testing phase, we sample w^* and $\{x_t\}_{t=1}^T$ from the Gaussian distribution and let the model predict the response value of a query x_{t+1} given the previous examples $\{x_i, y_i\}_{i=1}^t$. We reuse the code and model in Garg et al. (2022) for the experiments. The error bar in Figure 2 is derived from 90% confidence intervals over 1000 bootstrap trials.

We ran all the experiments on A100-40G.

5 Proofs for Section 3.3

5.1 Introduction of Conditional Mean Embedding

Let $\mathcal{H}_k$ and $\mathcal{H}_v$ be the two RKHSs over the spaces $\mathfrak{Q}$ and $\mathfrak{V}$ with the kernels $\mathfrak{K}$ and $\mathfrak{L}$, respectively. We denote by $\phi : \mathfrak{Q} \to \ell_2$ and $\varphi : \mathfrak{V} \to \ell_2$ the feature mappings associated with $\mathcal{H}_k$ and $\mathcal{H}_v$, respectively. Here ℓ_2 is the space of the square-integrable function class. Then it holds for any $k, k' \in \mathfrak{Q}$ and $v, v' \in \mathfrak{V}$ that

$$\phi(k)^\top \phi(k') = \mathfrak{K}(k, k'), \qquad \varphi(v)^\top \varphi(v) = \mathfrak{L}(v, v'). \tag{5.1}$$

Let $\mathbb{P}_{\mathcal{K},\mathcal{V}}$ be the joint distribution of the two random variables $\mathcal{K}$ and $\mathcal{V}$ taking values in $\mathfrak{Q}$ and $\mathfrak{V}$, respectively. Then the conditional mean embedding $\mathtt{CME}(q, \mathbb{P}_{\mathcal{K},\mathcal{V}}) \in \mathcal{H}_v$ of the conditional distribution $\mathbb{P}_{\mathcal{V}\,|\,\mathcal{K}}$ is defined as

$$\mathtt{CME}(q, \mathbb{P}_{\mathcal{K},\mathcal{V}}) = \mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \cdot)\,\big|\,\mathcal{K} = q\big].$$

The conditional mean embedding operator $C_{\mathcal{V}\,|\,\mathcal{K}} : \mathcal{H}_k \to \mathcal{H}_v$ is a linear operator such that

$$C_{\mathcal{V}\,|\,\mathcal{K}}\mathfrak{K}(q, \cdot) = \mathtt{CME}(q, \mathbb{P}_{\mathcal{K},\mathcal{V}}),$$

for any $q \in \mathfrak{Q}$. We define the (uncentered) covariance operator $C_{\mathcal{K}\mathcal{K}} : \mathcal{H}_k \to \mathcal{H}_k$ and the (uncentered) cross-covariance operator $C_{\mathcal{V}\mathcal{K}} : \mathcal{H}_k \to \mathcal{H}_v$ as follows,

$$C_{\mathcal{K}\mathcal{K}} = \mathbb{E}\big[\mathfrak{K}(\mathcal{K},\cdot) \otimes \mathfrak{K}(\mathcal{K},\cdot)\big], \qquad C_{\mathcal{V}\mathcal{K}} = \mathbb{E}\big[\mathfrak{L}(\mathcal{V},\cdot) \otimes \mathfrak{K}(\mathcal{K},\cdot)\big].$$

Here $\otimes$ is the tensor product. Song et al. (2009) shows that $C_{\mathcal{V}\,|\,\mathcal{K}} = C_{\mathcal{V}\mathcal{K}} C_{\mathcal{K}\mathcal{K}}^{-1}$. Thus, we have that

$$\texttt{CME}(c, \mathbb{P}_{\mathcal{K},\mathcal{V}}) = C_{\mathcal{V}\mathcal{K}} C_{\mathcal{K}\mathcal{K}}^{-1} \mathfrak{K}(c,\cdot). \tag{5.2}$$

For i.i.d. samples $\{(k^\ell, v^\ell)\}_{\ell\in[L]}$ of $\mathbb{P}_{\mathcal{K},\mathcal{V}}$, $\|\cdot\|_{\text{HS}}$ denotes the Hilbert-Schmidt norm, we write $\phi(K) = (\phi(k^1),\ldots,\phi(k^L))^\top \in \mathbb{R}^{L\times d_\phi}$ and $\varphi(V) = (\phi(v^1),\ldots,\phi(v^L))^\top \in \mathbb{R}^{L\times d_\varphi}$. Then the empirical covariance operator $\widehat{C}_{\mathcal{K}\mathcal{K}}$ and empirical cross-covariance operator $\widehat{C}_{\mathcal{V}\mathcal{K}}$ are defined as

$$\begin{aligned}
\widehat{C}_{\mathcal{K}\mathcal{K}} &= L^{-1}\sum_{\ell=1}^{L} \phi(k^\ell)\phi(k^\ell)^\top = L^{-1}\phi(K)^\top\phi(K) \in \mathbb{R}^{d_\phi\times d_\phi} \\
\widehat{C}_{\mathcal{V}\mathcal{K}} &= L^{-1}\sum_{\ell=1}^{L} \varphi(v^\ell)\varphi(k^\ell)^\top = L^{-1}\varphi(V)\phi(K)^\top \in \mathbb{R}^{d_\varphi\times d_\phi}.
\end{aligned} \tag{5.3}$$

The empirical version of the conditional operator is

$$\widehat{C}^{\lambda}_{\mathcal{V}\,|\,\mathcal{K}} = \varphi(Y)^\top\phi(X)\big(\phi(X)^\top\phi(X) + \lambda\mathcal{I}\big)^{-1} = \widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1} \in \mathbb{R}^{d_\varphi\times d_\phi}.$$

5.2 Proof of Proposition 3.3

Proof. By Assumption 3.2, we have that $\mathbb{P}_\theta(r_{t+1}\,|\,\texttt{pt}_t) = \mathbb{P}(r_{t+1}\,|\,\texttt{pt}_t)$. By (3.2), we have that

$$\begin{aligned}
\mathbb{P}(r_{t+1}\,|\,\texttt{pt}_t) &= \int \mathbb{P}(r_{t+1}\,|\,h_{t+1},\texttt{pt}_t)\mathbb{P}(h_{t+1}\,|\,\texttt{pt}_t)\mathrm{d}h_{t+1} &(5.4)\\
&= \int \mathbb{P}(r_{t+1}\,|\,\widetilde{c}_{t+1},h_{t+1})\mathbb{P}(h_{t+1}\,|\,S_t)\mathrm{d}h_{t+1} \\
&= \int \mathbb{P}(r_{t+1}\,|\,\widetilde{c}_{t+1},h_{t+1})\mathbb{P}(h_{t+1}\,|\,S_t,z)\mathbb{P}(z\,|\,S_t)\mathrm{d}h_{t+1}\mathrm{d}z \\
&= \int \mathbb{P}(r_{t+1}\,|\,\widetilde{c}_{t+1},h_{t+1},S_t,z)\mathbb{P}(h_{t+1}\,|\,S_t,z)\mathrm{d}h_{t+1}\mathbb{P}(z\,|\,S_t)\mathrm{d}z \\
&= \int \mathbb{P}(r_{t+1}\,|\,\widetilde{c}_{t+1},S_t,z)\mathbb{P}(z\,|\,S_t)\mathrm{d}z, &(5.5)
\end{aligned}$$

where the first inequality results from the Bayes rule, the second equality results from the fact that r_{t+1} is conditionally independent with the previous history given $h_{t+1},\widetilde{c}_{t+1}$ and the fact that h_{t+1} only parameterizes the transition kernel of r_{t+1} given c_{t+1} in (3.2), the fourth equality results from the fact that r_{t+1} is conditionally independent with the other variables given $h_{t+1},\widetilde{c}_{t+1}$, and the last equality results from the Bayes' rule.

□

5.3 Proof and Extension of Proposition 3.4

In this section, we first present the proof of Proposition 3.4 for discrete $\mathfrak{Z}$.

Proof. Note that

$$\mathbb{P}(z\,|\,S_t) = \frac{\mathbb{P}(S_t\,|\,z)\mathbb{P}_{\mathcal{Z}}(z)}{\int \mathbb{P}(S_t\,|\,z')\mathbb{P}_{\mathcal{Z}}(z')\mathrm{d}z'} = \frac{\prod_{i=1}^t \mathbb{P}(r_i\,|\,z,S_{i-1},c_i)\mathbb{P}_{\mathcal{Z}}(z)}{\int \prod_{i=1}^t \mathbb{P}(r_i\,|\,z',S_{i-1},c_i)\mathbb{P}_{\mathcal{Z}}(z')\mathrm{d}z'},$$

where the second equality results from the fact that the hidden variable z only parameterizes the *conditional probability* of r_t given c_t, and c_t and z are independent. Then, by Bayesian model averaging, we have the following

density estimation,

$$\begin{aligned}\mathbb{P}(r_{t+1} \,|\, S_t, c_{t+1}) &= \int \mathbb{P}(r_{t+1} \,|\, z, S_t, c_{t+1}) \mathbb{P}(z \,|\, S_t) \mathrm{d}z \\ &= \frac{\int \prod_{i=1}^{t+1} \mathbb{P}(r_i \,|\, z, S_{i-1}, c_i) \mathbb{P}_{\mathcal{Z}}(z) \mathrm{d}z}{\int \prod_{i=1}^{t} \mathbb{P}(r_i \,|\, z', S_{i-1}, c_i) \mathbb{P}_{\mathcal{Z}}(z') \mathrm{d}z'}.\end{aligned}$$

Thus, it holds that

$$\begin{aligned}-\sum_{t=0}^{T} \log \mathbb{P}(r_{t+1} \,|\, c_{t+1}, S_t) &= -\sum_{t=0}^{T} \Big(\log \int \prod_{i=1}^{t+1} \mathbb{P}(r_i \,|\, z, S_{i-1}, c_i) \mathbb{P}_{\mathcal{Z}}(z) \mathrm{d}z - \log \int \prod_{i=1}^{t} \mathbb{P}(r_i \,|\, z, S_{i-1}, c_i) \mathbb{P}_{\mathcal{Z}}(z) \mathrm{d}z \Big) \\ &= -\log \int \prod_{i=1}^{T+1} \mathbb{P}(r_i \,|\, z, S_{i-1}, c_i) \mathbb{P}_{\mathcal{Z}}(z) \mathrm{d}z \\ &= \inf_{q} \mathbb{E}_{z \sim q} \Big[-\sum_{i=1}^{T+1} \log \mathbb{P}(r_i \,|\, z, S_{i-1}, c_i) \Big] + \mathbb{E}_{z \sim q} \Big[\log \frac{q(z)}{\mathbb{P}_{\mathcal{Z}}(z)} \Big],\end{aligned}$$

where the second equality results from the fact that $\mathbb{P}(r_{t+1} \,|\, c_{t+1}, S_t) = \frac{\int \mathbb{P}(r_1 \,|\, c_1, z) \mathbb{P}(z) \mathrm{d}z}{1}$ when $t = 0$, and the last equality results from the standard Lagrangian arguments.

We consider q to be in the class of all Dirac measures. Then, we have that

$$-\frac{1}{T} \sum_{t=1}^{T} \log \mathbb{P}(r_t \,|\, c_t, S_{t-1}) \leq \frac{1}{T} \inf_{z} \Big(-\sum_{t=1}^{T} \log \mathbb{P}(r_t \,|\, z, S_{t-1}, c_t) - \log \mathbb{P}_{\mathcal{Z}}(z) \Big).$$

Thus, the statistical convergence rate of the Bayesian posterior averaging is $\mathcal{O}(1/T)$. □

5.4 Proof of Proposition 3.6

Proof. The proof of Proposition 3.6 mainly involves two steps

- Build the relationship between $\texttt{attn}_\dagger$ and conditional mean embedding.
- Build the relationship between the $\texttt{attn}$ and conditional mean embedding.

Step 1: Build the relationship between $\texttt{attn}_\dagger$ and conditional mean embedding.

In the following, we adopt $\mathcal{H}_k$ and $\mathcal{H}_v$ to denote the RKHSs for the key and the value with the kernel functions $\mathfrak{K}$ and $\mathfrak{L}$, respectively. Also, we use $\|\cdot\|$ to denote the norm of RKHS for an element in the corresponding RKHS and the operator norm of the operators that transform elements between RKHSs. For the value space, we adopt the Euclidean kernel $\mathfrak{L}(v, v') = v^\top v'$, and the feature mapping φ is the identity mapping. Recall the definition of the empirical covariance operator and the empirical cross-covariance operator in Appendix 5.1. For keys and values, we correspondingly define them as

$$\widehat{C}_{\mathcal{K}\mathcal{K}} = L^{-1} \phi(K)^\top \phi(K),\ \widehat{C}_{\mathcal{V}\mathcal{K}} = L^{-1} \varphi(V)^\top \phi(K),\ \widehat{C}_{\mathcal{V}\mathcal{V}} = L^{-1} \varphi(V)^\top \varphi(V),$$

where $\phi(K) = (\phi(k^1), \ldots, \phi(k^L))^\top \in \mathbb{R}^{L \times d_\phi}$ and $\varphi(V) = (\varphi(v^1), \ldots, \varphi(v^L))^\top \in \mathbb{R}^{L \times d_\varphi}$ By the definition of the newly defined attention in Section 3.3, we have that

$$\texttt{attn}_\dagger(q, K, V) = \widehat{C}_{\mathcal{V}\mathcal{K}} (\widehat{C}_{\mathcal{K}\mathcal{K}} + L^{-1} \lambda \mathcal{I})^{-1} \phi(q),$$

which implies that $\texttt{attn}_\dagger$ recovers the empirical conditional mean embedding. By (5.2), it holds that

$$\begin{aligned}&\big\| \texttt{attn}_\dagger(q, K, V) - \texttt{CME}(q, \mathbb{P}_{\mathcal{K},\mathcal{V}}) \big\| \\ &\quad \leq \underbrace{\big\| \widehat{C}_{\mathcal{V}\mathcal{K}} (\widehat{C}_{\mathcal{K}\mathcal{K}} + L^{-1} \lambda \mathcal{I})^{-1} \phi(q) - C_{\mathcal{V}\mathcal{K}} (C_{\mathcal{K}\mathcal{K}} + L^{-1} \lambda \mathcal{I})^{-1} \phi(q) \big\|}_{\text{(i)}} \\ &\quad + \underbrace{\big\| C_{\mathcal{V}\mathcal{K}} (C_{\mathcal{K}\mathcal{K}} + L^{-1} \lambda \mathcal{I})^{-1} \mathfrak{K}(q, \cdot) - C_{\mathcal{V}\mathcal{K}} C_{\mathcal{K}\mathcal{K}}^{-1} \mathfrak{K}(q, \cdot) \big\|}_{\text{(ii)}}. \end{aligned} \tag{5.6}$$

Upper bounding term (i) of (5.6). Following the proof from Song et al. (2009), we only need to upper bound $\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}-C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\|$. It holds that

$$\begin{aligned}&\big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}-C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\| \\ &\quad\le \big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}(\widehat{C}_{\mathcal{K}\mathcal{K}}-C_{\mathcal{K}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|+\big\|(\widehat{C}_{\mathcal{V}\mathcal{K}}-C_{\mathcal{V}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|.\end{aligned} \tag{5.7}$$

Considering the first term on the right-hand side of (5.7), we have the operator decomposition $\widehat{C}_{\mathcal{V}\mathcal{K}}=\widehat{C}_{\mathcal{V}\mathcal{V}}^{1/2}\mathcal{W}\widehat{C}_{\mathcal{K}\mathcal{K}}^{1/2}$ for $\mathcal{W}$ such that $\|\mathcal{W}\|\le 1$. This decomposition implies that

$$\begin{aligned}&\big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}(\widehat{C}_{\mathcal{K}\mathcal{K}}-C_{\mathcal{K}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\| \\ &\le \|\widehat{C}_{\mathcal{V}\mathcal{V}}\|^{1/2}\cdot\big\|\widehat{C}_{\mathcal{K}\mathcal{K}}^{1/2}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1/2}\big\|\cdot\big\|(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1/2}\big\|\cdot\big\|(\widehat{C}_{\mathcal{K}\mathcal{K}}-C_{\mathcal{K}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\| \\ &\le (L^{-1}\lambda)^{-1/2}\cdot\big\|(\widehat{C}_{\mathcal{K}\mathcal{K}}-C_{\mathcal{K}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|,\end{aligned} \tag{5.8}$$

where the last inequality follows from the fact that

$$\|\widehat{C}_{\mathcal{V}\mathcal{V}}\|^2=L^{-1}\sum_{\ell=1}^{L}\|v^\ell\|_2^2\le 1,\quad \widehat{C}_{\mathcal{K}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\le\mathcal{I},\quad (\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\le(L^{-1}\lambda)^{-1}\mathcal{I}.$$

Combining (5.8) and (5.7), we have

$$\begin{aligned}&\big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}-C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\| \\ &\quad\le (L^{-1}\lambda)^{-1/2}\cdot\big\|(\widehat{C}_{\mathcal{K}\mathcal{K}}-C_{\mathcal{K}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|+\big\|(\widehat{C}_{\mathcal{V}\mathcal{K}}-C_{\mathcal{V}\mathcal{K}})(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|.\end{aligned} \tag{5.9}$$

In the following, we will upper bound the second term on the right-hand side of (5.9) with Lemma 9.1. For this purpose, we define $\xi:\mathbb{R}^{d_{\mathrm{P}}}\times\mathbb{R}^d\to\mathcal{H}_k\otimes\mathcal{H}_v$ as follows,

$$\xi(k,v)=\varphi(v)\phi(k)^\top(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}.$$

Since the operator norm of $(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}$ is upper bounded by $(L^{-1}\lambda)^{-1}$, we have that

$$\big\|\xi(k,v)\big\|=\big\|(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|\cdot\big\|\varphi(v)\big\|\cdot\big\|\phi(k)\big\|\le C\cdot(L^{-1}\lambda)^{-1},$$

where $C>0$ is an absolute constant. Additionally, we can bound the expectation of the squared norm of $\xi(k,v)$ as

$$\begin{aligned}\mathbb{E}\Big[\big\|\xi(k,v)\big\|^2\Big]&=\mathbb{E}\Big[\big\|\phi(k)^\top(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|^2\cdot\big\|\varphi(v)\big\|^2\Big]\\&\le\mathbb{E}\Big[\big\|(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\phi(k)\big\|^2\Big]\\&\le(L^{-1}\lambda)^{-1}\cdot\mathbb{E}\Big[\big\langle(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\phi(k),\phi(k)\big\rangle\Big].\end{aligned}$$

Using the definition of the trace operator, we have

$$\begin{aligned}\mathbb{E}\Big[\big\|\xi(k,v)\big\|^2\Big]&\le\mathbb{E}\Big[\mathrm{tr}\big((C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-2}\phi(k)\phi(k)^\top\big)\Big]\\&\le(L^{-1}\lambda)^{-1}\cdot\mathrm{tr}\big((C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}}\big)\\&=(L^{-1}\lambda)^{-1}\cdot\Gamma(L^{-1}\lambda).\end{aligned}$$

Here $\Gamma(L^{-1}\lambda)$ is the effective dimension of $C_{\mathcal{K}\mathcal{K}}$ in Caponnetto and De Vito (2007), which is defined as follows,

$$\Gamma(L^{-1}\lambda)=\mathrm{tr}\big((C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}}\big).$$

We apply Lemma 9.1 with $B=C(L^{-1}\lambda)^{-1}$ and $\sigma^2=(L^{-1}\lambda)^{-1}\cdot\Gamma(L^{-1}\lambda)$, then we have that with probability at least $1-\delta$, the following holds

$$\big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}-C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}}+L^{-1}\lambda\mathcal{I})^{-1}\big\|\le C\cdot\left(\frac{2}{\lambda}+\sqrt{\frac{\Gamma(L^{-1}\lambda)}{\lambda}}\right)\log\frac{2}{\delta}, \tag{5.10}$$

where $C > 0$ is an absolute constant. Similarly, we can prove that with probability at least $1 - \delta$, the following holds

$$\big\|\widehat{C}_{\mathcal{K}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1} - C_{\mathcal{K}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}\big\| \le C' \cdot \bigg(\frac{2}{\lambda} + \sqrt{\frac{\Gamma(L^{-1}\lambda)}{\lambda}}\bigg)\log\frac{2}{\delta}. \tag{5.11}$$

Here $C' > 0$ is an absolute constant. Combining (5.9), (5.10), and (5.11), we have with probability at least $1 - \delta$ that

$$\begin{aligned}&\big\|\widehat{C}_{\mathcal{V}\mathcal{K}}(\widehat{C}_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1} - C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}\big\| \\ &\quad\le C'' \cdot \sqrt{\frac{L}{\lambda}} \cdot \bigg(\frac{2}{\lambda} + \sqrt{\frac{\Gamma(L^{-1}\lambda)}{\lambda}}\bigg)\log\frac{2}{\delta}.\end{aligned} \tag{5.12}$$

Upper bounding term (ii) of (5.6). We follow the procedures in the proof from Fukumizu (2015). For any $g \in \mathcal{H}_k$, we have that

$$\begin{aligned}\langle C_{\mathcal{V}\mathcal{K}}(g), C_{\mathcal{V}\mathcal{K}}(g)\rangle &= \mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}})g(\mathcal{K})g(\bar{\mathcal{K}})\big] \\ &= \Big\langle (C_{\mathcal{K}\mathcal{K}} \otimes C_{\mathcal{K}\mathcal{K}})\mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}}) \,\big|\, \mathcal{K} = \cdot, \bar{\mathcal{K}} = \ddagger\big], g \otimes g\Big\rangle.\end{aligned}$$

Similarly, for any $q \in \mathbb{R}^{d_{\mathrm{p}}}$ and any $g \in \mathcal{H}_k$, we have that

$$\begin{aligned}\Big\langle C_{\mathcal{V}\mathcal{K}}, \mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \cdot) \,\big|\, \mathcal{K} = q\big]\Big\rangle &= \Big\langle \mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}}) \,\big|\, \mathcal{K} = q, \mathcal{K} = \ddagger\big], C_{\mathcal{K}\mathcal{K}}g\Big\rangle \\ &= \Big\langle (\mathcal{I} \otimes C_{\mathcal{K}\mathcal{K}})\mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}}) \,\big|\, \mathcal{K} = \cdot, \bar{\mathcal{K}} = \ddagger\big], \mathfrak{L}(\cdot, q) \otimes g\Big\rangle.\end{aligned}$$

Taking $g = (C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}\mathfrak{K}(q, \cdot)$, we have that

$$\begin{aligned}&\big\|C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}\mathfrak{K}(q, \cdot) - C_{\mathcal{V}\mathcal{K}}C_{\mathcal{K}\mathcal{K}}^{-1}\mathfrak{K}(q, \cdot)\big\|^2 \\ &\quad= \Big\langle \Big((C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \otimes (C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} - \mathcal{I} \otimes (C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \\ &\qquad(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \otimes \mathcal{I} + \mathcal{I} \otimes \mathcal{I}\Big)\mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}}) \,\big|\, \mathcal{K} = \cdot, \bar{\mathcal{K}} = \ddagger\big], \mathfrak{K}(q, \cdot) \otimes \mathfrak{K}(q, \dagger)\Big\rangle.\end{aligned}$$

We note that $\mathbb{E}[\mathfrak{L}(v, \bar{v}) \,|\, k = \cdot, \bar{k} = \ddagger] \in \mathcal{H}_k \otimes \mathcal{H}_k$ is in the range spanned by $C_{\mathcal{K}\mathcal{K}} \otimes C_{\mathcal{K}\mathcal{K}}$. Thus, we can define $\widetilde{\mathcal{C}} \in \mathcal{H}_k \times \mathcal{H}_k$ such that $(C_{\mathcal{K}\mathcal{K}} \otimes C_{\mathcal{K}\mathcal{K}})\widetilde{\mathcal{C}} = \mathbb{E}[\mathfrak{L}(v, \bar{v}) \,|\, k = \cdot, \bar{k} = \ddagger]$. Let $\{\lambda_i\}_{i=1}^{\infty}$ and $\{\varphi_i\}_{i=1}^{\infty}$ be the eigenvalues and eigenvectors of $C_{\mathcal{K}\mathcal{K}}$, respectively. We then have that

$$\begin{aligned}&\big\|C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}\mathfrak{K}(q, \cdot) - C_{\mathcal{V}\mathcal{K}}C_{\mathcal{K}\mathcal{K}}^{-1}\mathfrak{K}(q, \cdot)\big\|^4 \\ &\quad\le \bigg\|\Big((C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \otimes (C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} - \mathcal{I} \otimes (C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \\ &\qquad(C_{\mathcal{K}\mathcal{K}} + L^{-1}\lambda\mathcal{I})^{-1}C_{\mathcal{K}\mathcal{K}} \otimes \mathcal{I} + \mathcal{I} \otimes \mathcal{I}\Big)\mathbb{E}\big[\mathfrak{L}(\mathcal{V}, \bar{\mathcal{V}}) \,\big|\, \mathcal{K} = \cdot, \bar{\mathcal{K}} = \ddagger\big]\bigg\|^2 \\ &\quad= \sum_{i,j}\bigg(\frac{\lambda_i\lambda_j(L^{-1}\lambda)^2}{(\lambda_i + L^{-1}\lambda)(\lambda_j + L^{-1}\lambda)}\bigg)^2 \cdot \langle \varphi_i \otimes \varphi_j, \widetilde{\mathcal{C}}\rangle^2 \\ &\quad\le (L^{-1}\lambda)^4 \cdot \|\widetilde{\mathcal{C}}\|^2.\end{aligned}$$

Thus, we have

$$\big\|C_{\mathcal{V}\mathcal{K}}(C_{\mathcal{K}\mathcal{K}} + \lambda\mathcal{I})^{-1}\mathfrak{K}(q, \cdot) - C_{\mathcal{V}\mathcal{K}}C_{\mathcal{K}\mathcal{K}}^{-1}\mathfrak{K}(q, \cdot)\big\|_2 \le C \cdot \lambda L^{-1}, \tag{5.13}$$

where $C > 0$ is an absolute constant.

Combining (5.6), (5.12), and (5.13), we have with probability at least $1-\delta$, the following holds

$$\left\|\texttt{attn}_\dagger(q,K,V)-\texttt{CME}(q,\mathbb{P}_{\mathcal{K},\mathcal{V}})\right\|\leq\mathcal{O}\bigg(\sqrt{\frac{L}{\lambda}}\cdot\bigg(\frac{2}{\lambda}+\sqrt{\frac{\Gamma(L^{-1}\lambda)}{\lambda}}\bigg)\log\frac{1}{\delta}+\lambda L^{-1}\bigg). \tag{5.14}$$

Since $\mathfrak{K}$ is Gaussian RBF kernel, we have that $\Gamma(L^{-1}\lambda)=\mathcal{O}(L/\lambda)$.

Step 2: Build the relationship between the `attn` and conditional mean embedding.

We achieve our goal in two sub-steps. In the first step, we prove that there exists a constant $C>0$ such that

$$\texttt{attn}_{\texttt{SM}}(q,K,V)=C\int_{\mathbb{S}^{d-1}}v\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}(v\,|\,q)\mathrm{d}v, \tag{5.15}$$

where $\mathbb{S}^{d-1}$ is the $(d-1)$-dimensional unit sphere. Here $\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}$ is the kernel conditional density estimation of $\mathbb{P}_{\mathcal{V}\,|\,\mathcal{K}}$ defined as follows,

$$\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}(v\,|\,q)=\iota\cdot\frac{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)\cdot\mathfrak{K}(v^\ell,v)}{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)},$$

where $\iota=1/\int_{\mathbb{S}^{d-1}}\mathfrak{K}(v,q)\mathrm{d}q$ is a normalization constant. Note that ι does not depend on the value of k by symmetry. We transform the right-hand side of this equality as

$$\begin{aligned}\int v\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}(v\,|\,q)\mathrm{d}v&=\iota\cdot\int_{\mathbb{S}^{d-1}}v\cdot\frac{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)\cdot\mathfrak{K}(v^\ell,v)}{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)}\mathrm{d}v\\&=\frac{\iota\cdot\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)\cdot\int_{\mathbb{S}^{d-1}}v\cdot\mathfrak{K}(v^\ell,v)\mathrm{d}v}{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)}.\end{aligned} \tag{5.16}$$

Thus, it suffices to calculate the integration term $\int_{\mathbb{S}^{d-1}}v\cdot\mathfrak{K}(v^\ell,v)\mathrm{d}v$. To this end, we have the following lemma.

Proposition 5.1. Let $\mathfrak{K}(a,b)=\exp(a^\top b/\gamma)$ be the exponential kernel with a fixed $\gamma>0$. It holds for any $b\in\mathbb{S}^{d-1}$ that

$$\int_{\mathbb{S}^{d-1}}a\cdot\mathfrak{K}(a,b)\mathrm{d}a=C_1\cdot b,$$

where $C_1>0$ is an absolute constant.

Proof. See Section 8.1 for a detailed proof. □

Thus, it holds for the right-hand side of (5.16) that

$$\iota\cdot C_1\cdot\frac{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)\cdot v^\ell}{\sum_{\ell=1}^{L}\mathfrak{K}(k^\ell,q)}=\iota\cdot C_1\cdot V^\top\texttt{softmax}(Kq/\gamma)=\iota\cdot C_1\cdot\texttt{attn}_{\texttt{SM}}(q,K,V),$$

where the first equality follows from the definition of the softmax function and the second equality follows from the definition of the softmax attention.

The second step is to relate the right-hand side of (5.15) to conditional mean embedding. In fact, under the condition that $\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}(v\,|\,q)\to\mathbb{P}(v\,|\,q)$ uniformly for any $q\in\mathbb{S}^{d_{\mathrm{p}}-1}$ as $L\to\infty$, we have

$$\int v\widehat{\mathbb{P}}^{\mathfrak{K}}_{\mathcal{V}\,|\,\mathcal{K}}(v\,|\,q)\mathrm{d}v\to\mathbb{E}[\mathcal{V}\,|\,\mathcal{K}=q]\qquad\text{as}\quad L\to\infty.$$

Thus, we have that

$$\texttt{attn}_{\texttt{SM}}(q,K,V)\to C\cdot\mathbb{E}[\mathcal{V}\,|\,\mathcal{K}=q]\qquad\text{as}\quad L\to\infty \tag{5.17}$$

for some constant $C>0$. Combining (5.17) and (5.14) and choosing $\lambda=L^{3/4}$, we complete the proof of Proposition 3.6.

□

6 Appendix for Section 4

6.1 Supplemental Definitions for Markov Chains

We follow the notations in Paulin (2015). Let Ω be a Polish space. The transition kernel for a time-homogeneous Markov chain $\{X_i\}_{i=1}^{\infty}$ supported on Ω is a probability distribution $\mathbb{P}(x, \mathrm{d}y)$ for every $x \in \Omega$. Given $X_1 = x_1, \cdots, X_{t-1} = x_{t-1}$, the conditional distribution of X_t equals $\mathbb{P}(x_{t-1}, \mathrm{d}y)$. A distribution π is said to be a stationary distribution of this Markov chain if $\int_{x\in\Omega} \mathbb{P}(x, \mathrm{d}y)\pi(\mathrm{d}x) = \pi(\mathrm{d}y)$. We adopt $\mathbb{P}^t(x, \cdot)$ to denote the distribution of X_t conditioned on $X_1 = x$. The *mixing time* of the chain is defined by

$$d(t) = \sup_{x\in\Omega} \mathtt{TV}\big(P^t(x, \cdot), \pi\big), \quad t_{\mathrm{mix}}(\varepsilon) = \min\{t \,|\, d(t) \le \varepsilon\}, \quad t_{\mathrm{mix}} = t_{\mathrm{mix}}(1/4).$$

6.2 Proof of Theorem 4.3

Proof of Theorem 4.3. Our proof mainly involves three steps.

- Error decomposition with the PAC-Bayes framework.
- Control each term in the error decomposition.
- Conclude the proof.

Step 1: Error decomposition with the PAC-Bayes framework.

For ease of notation, we temporarily write T_{p} and N_{p} as T and N, respectively. Recall that the pretraining dataset is $\mathcal{D} = \{(S_t^n, x_{t+1}^n)\}_{n,t=1}^{N,T}$, which consists of N trajectories (essays), and each essay have $T+1$ words. Given S_t^n, the next word is generated as $x_{t+1}^n \sim \mathbb{P}(\cdot \,|\, S_t^n)$, and $S_{t+1}^n = (S_t^n, x_{t+1}^n)$. Here, we construct a ghost sample $\widetilde{\mathcal{D}} = \{(\widetilde{S}_t^n, \widetilde{x}_{t+1}^n)\}_{n,t=1}^{N,T}$ as $\widetilde{S}_t^n = S_t^n$ and $\widetilde{x}_{t+1}^n \sim \mathbb{P}(\cdot \,|\, \widetilde{S}_t^n)$ independently from $\mathcal{D}$. We define function $g(\theta) = L(\theta, D) - \log \mathbb{E}_{\widetilde{\mathcal{D}}}[\exp(L(\theta, \widetilde{\mathcal{D}})) \,|\, \mathcal{D}]$, where

$$L(\theta, \widetilde{D}) = -\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T} \log \frac{\mathbb{P}(\widetilde{x}_{t+1}^n \,|\, S_t^n)}{\mathbb{P}_\theta(\widetilde{x}_{t+1}^n \,|\, S_t^n)}.$$

For distributions $Q, P \in \Delta(\Theta)$, where P can potentially depends on $\mathcal{D}$, Lemma 9.3 shows that

$$\mathbb{E}_P\big[g(\theta)\big] \le \mathtt{KL}(P\|Q) + \log \mathbb{E}_Q\big[\exp\big(g(\theta)\big)\big].$$

Substituting the definition of $g(\theta)$ and taking expectation with respect to the distribution of $\mathcal{D}$ on the both sides of the inequality, we can derive that

$$\mathbb{E}_{\mathcal{D}}\bigg[\exp\Big\{\mathbb{E}_P\Big[L(\theta, \mathcal{D}) - \log \mathbb{E}_{\widetilde{\mathcal{D}}}\big[\exp\big(L(\theta, \widetilde{\mathcal{D}})\big) \,|\, \mathcal{D}\big]\Big] - \mathtt{KL}(P \,\|\, Q)\Big\}\bigg] \le 1.$$

With Chernoff inequality, we can show that with probability at least $1-\delta$, the following holds

$$-\mathbb{E}_{\theta\sim P}\Big[\log \mathbb{E}_{\widetilde{\mathcal{D}}}\big[\exp\big(L(\theta, \widetilde{\mathcal{D}})\big) \,|\, \mathcal{D}\big]\Big] \le -\mathbb{E}_P\big[L(\theta, \mathcal{D})\big] + \mathtt{KL}(P \,\|\, Q) + \log\frac{1}{\delta}. \tag{6.1}$$

We first cope with the left-hand side of (6.1).

$$
\begin{aligned}
&-\mathbb{E}_P\Big[\log\mathbb{E}_{\widetilde{\mathcal{D}}}\big[\exp\big(L(\theta,\widetilde{\mathcal{D}})\big)\,|\,\mathcal{D}\big]\Big]\\
&\quad\geq -\frac{1}{2}\log\mathbb{E}_{\widetilde{\mathcal{D}}}\bigg[\exp\bigg(-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\widehat{\theta}}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\\
&\qquad-\frac{1}{2}\mathbb{E}_{\theta\sim P}\bigg[\log\mathbb{E}_{\widetilde{\mathcal{D}}}\bigg[\exp\bigg(-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}_{\widehat{\theta}}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\bigg]\\
&\quad=-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\mathbb{E}_{\widetilde{x}_{t+1}^n\sim\mathbb{P}(\cdot\,|\,S_t^n)}\bigg[\exp\bigg(-\frac{1}{2}\log\frac{\mathbb{P}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\widehat{\theta}}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\\
&\qquad-\frac{1}{2}\mathbb{E}_{\theta\sim P}\bigg[\log\mathbb{E}_{\widetilde{\mathcal{D}}}\bigg[\exp\bigg(-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}_{\widehat{\theta}}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\bigg]\\
&\quad\geq\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S_t^n),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S_t^n)\big)^2-\frac{1}{2}\mathbb{E}_{\theta\sim P}\bigg[\log\mathbb{E}_{\widetilde{\mathcal{D}}}\bigg[\exp\bigg(-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}_{\widehat{\theta}}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(\widetilde{x}_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\bigg],
\end{aligned}
\tag{6.2}
$$

where the first inequality results from the definition of $L(\theta,\mathcal{D})$ and Cauchy-Schwarz inequality, the equality results from that the transitions of $\widetilde{x}_{t+1}^n$ are independent given $\mathcal{D}$, and the last inequality results from Lemma 9.5. The second term in the right-hand side of (6.2) can be controlled if the distribution P is chosen to concentrate around $\widehat{\theta}$. This will be done in Step 2. Now we consider the right-hand side of (6.1). For any $\theta^*\in\Theta$, we can decompose it as

$$
\begin{aligned}
&-\mathbb{E}_P\big[L(\theta,\mathcal{D})\big]\\
&\quad=\mathbb{E}_P\bigg[\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n\,|\,S_t^n)}+\log\frac{\mathbb{P}_{\theta^*}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)}+\log\frac{\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(x_{t+1}^n\,|\,S_t^n)}\bigg]\\
&\quad\leq\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n\,|\,S_t^n)}+\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathbb{E}_P\bigg[\log\frac{\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(x_{t+1}^n\,|\,S_t^n)}\bigg],
\end{aligned}
\tag{6.3}
$$

where the inequality results from the fact that $\widehat{\theta}$ maximizes the likelihood. We will choose θ^* as the projection of $\mathbb{P}$ onto $\{\mathbb{P}_\theta\,|\,\theta\in\Theta\}$, i.e., $\mathbb{P}_\theta^*$ is the best approximation of $\mathbb{P}$ with respect to the KL divergence. Thus, the first term in the right-hand side of (6.3) is the approximation error. The second term in the right-hand side of (6.3) can be controlled in the same way as the second term in the right-hand side of (6.2). Combining inequalities (6.1), (6.2), and (6.3), we have that

$$
\begin{aligned}
&\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S_t^n),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S_t^n)\big)^2\\
&\quad\leq\underbrace{\frac{1}{2}\mathbb{E}_{\theta\sim P}\bigg[\log\mathbb{E}_{\widetilde{\mathcal{D}}}\bigg[\exp\bigg(-\frac{1}{2}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(x_{t+1}^n\,|\,S_t^n)}\bigg)\,\bigg|\,\mathcal{D}\bigg]\bigg]+\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathbb{E}_P\bigg[\log\frac{\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta}(x_{t+1}^n\,|\,S_t^n)}\bigg]}_{\text{(I)}}\\
&\qquad+\underbrace{\frac{1}{4}\sum_{n=1}^{N}\sum_{t=1}^{T}\log\frac{\mathbb{P}(x_{t+1}^n\,|\,S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n\,|\,S_t^n)}}_{\text{(II)}}+\underbrace{\mathtt{KL}(P\,\|\,Q)}_{\text{(III)}}+\log\frac{1}{\delta},
\end{aligned}
\tag{6.4}
$$

where term (I) is the fluctuation error induced by $\theta\sim P$, term (II) is the approximation error, and term (III) is the KL divergence between P and Q.

Step 2: Control each term in the error decomposition.

We first consider term (I). Since $\widehat{\theta}$ is a deterministic function of $\mathcal{D}$ and that $\log(\mathbb{P}_{\widehat{\theta}}(x_{t+1}^n\,|\,S_t^n)/\mathbb{P}_\theta(x_{t+1}^n\,|\,S_t^n))$ is close to 0 if θ is close to $\widehat{\theta}$, we need to design P for any $\widehat{\theta}\in\Theta$ such that $\theta\sim P$ is close to $\widehat{\theta}$ almost surely.

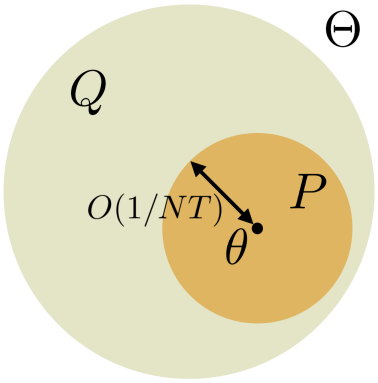

Figure 7: The distribution P in (6.5) is the uniform distribution on the neighborhood of θ with radius proportional to $1/NT$, and Q in (6.8) is the uniform distribution on Θ.

We need to quantify the fluctuation of $\mathbb{P}_\theta$ when θ is changing, i.e., how $\mathbb{P}_\theta$ is close to $\mathbb{P}_{\widehat{\theta}}$ when θ is close to $\widehat{\theta}$.

Proposition 6.1. For any input $X \in \mathbb{R}^{L\times d}$ and $\theta, \widetilde{\theta} \in \Theta$, we have that

$$\begin{aligned}&\mathtt{TV}\big(\mathbb{P}_\theta(\cdot\,|\,X), \mathbb{P}_{\widetilde{\theta}}(\cdot\,|\,X)\big)\\ &\quad\le \frac{2}{\tau}\big\|A^{(D+1),\top} - \widetilde{A}^{(D+1),\top}\big\|_{1,2} + \sum_{t=1}^{D}\alpha_t(\beta_t+\iota_t+\kappa_t+\rho_t),\end{aligned}$$

where

$$\begin{aligned}\alpha_t &= \frac{2}{\tau}B_A(1+B_{A,1}\cdot B_{A,2})\big(1+hB_V(1+4B_QB_K)\big)^{D-t}\\ \beta_t &= |\gamma_2^{(t)} - \widetilde{\gamma}_2^{(t)}| + (1+B_{A,1}\cdot B_{A,2})\cdot\big(1+(\|X^\top\|_{2,\infty}-1)\mathbb{I}_{t=1}\big)\cdot|\gamma_1^{(t)}-\widetilde{\gamma}_1^{(t)}|\\ \iota_t &= B_{A,2}\cdot\|A_1^{(t)} - \widetilde{A}_1^{(t)}\|_{\mathbf{F}} + B_{A,1}\cdot\|A_2^{(t)} - \widetilde{A}_2^{(t)}\|_{\mathbf{F}}\\ \kappa_t &= (1+B_{A,1}\cdot B_{A,2})\cdot\big(1+(\|X^\top\|_{2,\infty}-1)\mathbb{I}_{t=1}\big)\cdot\sum_{i=1}^{h}\big\|W_i^{V,(t)} - \widetilde{W}_i^{V,(t)}\|_{\mathbf{F}}\\ \rho_t &= 2(1+B_{A,1}\cdot B_{A,2})\cdot\big(1+(\|X^\top\|_{2,\infty}-1)\mathbb{I}_{t=1}\big)\cdot B_V\\ &\qquad\cdot\sum_{i=1}^{h}B_K\cdot\|W_i^{Q,(t+1)} - \widetilde{W}_i^{Q,(t+1)}\|_{\mathbf{F}} + B_Q\cdot\|W_i^{K,(t+1)} - \widetilde{W}_i^{K,(t+1)}\|_{\mathbf{F}}\end{aligned}$$

for all $t \in [D]$.

Proof of Proposition 6.1 . See Appendix 8.3. □

Proposition 6.1 implies that the difference between $\mathbb{P}_\theta$ and $\mathbb{P}_{\widetilde{\theta}}$ can be upper-bounded by the difference between the parameters of each layer. Thus, for any $\theta \in \mathcal{D}$, we set the distribution P as uniform distribution on the neighborhood of parameters, and the radius of the neighborhood is set proportional to $1/NT$ shown in Figure 7.

$$P = \prod_{t=1}^{D+1} \mathcal{L}_P\big(\theta^{(t)}\big) \tag{6.5}$$

$$\begin{aligned}
\mathcal{L}_P\big(\theta^{(D+1)}\big) &= \texttt{Unif}\Big(\mathbb{B}\big(\widehat{A}^{(D+1)}, r^{(D+1)}, \|\cdot\|_{1,2}\big)\Big) \\
\mathcal{L}_P\big(\theta^{(t)}\big) &= \texttt{Unif}\Big(\mathbb{B}\big(\widehat{\gamma}_1^{(t)}, r_{\gamma,1}^{(t)}, |\cdot|\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(\widehat{\gamma}_2^{(t)}, r_{\gamma,2}^{(t)}, |\cdot|\big)\Big) \cdot \mathcal{L}_P(A^{(t)}) \cdot \mathcal{L}_P(W^{(t)}) \\
\mathcal{L}_P(A^{(t)}) &= \texttt{Unif}\Big(\mathbb{B}\big(\widehat{A}_1^{(t)}, r_{A,1}^{(t)}, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(\widehat{A}_2^{(t)}, r_{A,2}^{(t)}, \|\cdot\|_{\mathbf{F}}\big)\Big) \\
\mathcal{L}_P(W^{(t)}) &= \prod_{i=1}^{h} \texttt{Unif}\Big(\mathbb{B}\big(\widehat{W}_i^{Q,(t)}, r_Q^{(t)}, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(\widehat{W}_i^{K,(t)}, r_K^{(t)}, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(\widehat{W}_i^{V,(t)}, r_V^{(t)}, \|\cdot\|_{\mathbf{F}}\big)\Big)
\end{aligned}$$

for $t \in [D]$, where $\texttt{Unif}$ denotes the uniform distribution on the set, $\mathbb{B}(a, r, \|\cdot\|) = \{x \,|\, \|x - a\| \le r\}$ denotes the ball centered in a with radius r, the radius is set as

$$\begin{aligned}
r_{\gamma,1}^{(t)} &= R^{-1}(1 + B_{A,1} \cdot B_{A,2})^{-1}\alpha_t^{-1}/NT, & r_{\gamma,2}^{(t)} &= R^{-1}\alpha_t^{-1}/NT \\
r_{A,1}^{(t)} &= R^{-1}B_{A,2}^{-1}\alpha_t^{-1}/NT, & r_{A,2}^{(t)} &= R^{-1}B_{A,1}^{-1}\alpha_t^{-1}/NT, \\
r_V^{(t)} &= R^{-1}h^{-1}(1 + B_{A,1} \cdot B_{A,2})^{-1}\alpha_t^{-1}/NT, & r_Q^{(t)} &= R^{-1}h^{-1}(1 + B_{A,1} \cdot B_{A,2})^{-1}B_V^{-1}B_K^{-1}\alpha_t^{-1}/NT \\
r_K^{(t)} &= R^{-1}h^{-1}(1 + B_{A,1} \cdot B_{A,2})^{-1}B_V^{-1}B_Q^{-1}\alpha_t^{-1}/NT, & r^{(D+1)} &= \tau B_A^{-1}/NT.
\end{aligned}$$

Under this assignment, we now bound $|\log \mathbb{P}_{\widehat{\theta}}(x \,|\, S)/\mathbb{P}_\theta(x \,|\, S)|$ for any $S \in \mathbb{R}^{L \times d}$ and $x \in \mathbb{R}^{d_y}$. We first note that

$$\mathbb{P}_{\widehat{\theta}}(x \,|\, S) \ge b_y = (1 + d_y \exp(B_A/\tau))^{-1} \tag{6.6}$$

for any S and x, which results from the softmax layer defined below (4.1). This results from the fact that the last layer of the transformer is softmax with inverse temperature parameter τ and that

$$\left\| \frac{1}{L\tau} \mathbb{I}_L^\top X^{(D)} A^{(D+1)} \right\|_1 \le \left\| A^{(D+1),\top} \right\|_{1,2} \le B_A.$$

If $\texttt{TV}(\mathbb{P}_\theta(\cdot \,|\, S), \mathbb{P}_{\widehat{\theta}}(\cdot \,|\, S)) = \varepsilon \le b_y/2$, some basic calculations show that

$$\frac{b_y}{b_y + \varepsilon} \le \frac{\mathbb{P}_{\widehat{\theta}}(x \,|\, S)}{\mathbb{P}_\theta(x \,|\, S)} \le 1 + \frac{2\varepsilon}{b_y}.$$

Thus, if we set the distribution P as the uniform distribution on the neighborhood around $\widehat{\theta}$ with radius proportional to $1/NT$, i.e., (6.5), then for $\theta \sim P$ we have that

$$\left| \log \frac{\mathbb{P}_{\widehat{\theta}}(x \,|\, S)}{\mathbb{P}_\theta(x \,|\, S)} \right| \le \frac{2\varepsilon}{b_y} = \mathcal{O}\left(\frac{1}{NT}\right) \quad \text{for } P \text{ a.s.}$$

Based on this, we conclude that

$$(\mathbf{I}) = \mathcal{O}(1). \tag{6.7}$$

Next, we control term (III) in (6.4). In order to upper bound $\texttt{KL}(P \,\|\, Q)$, we need to make sure that the support of P is a subset of that of Q. Thus, we take Q as the uniform distribution on the parameter space.

$$Q = \prod_{t=1}^{D+1} \mathcal{L}_Q\big(\theta^{(t)}\big) \tag{6.8}$$

$$\begin{aligned}
\mathcal{L}_Q\big(\theta^{(D+1)}\big) &= \texttt{Unif}\Big(\mathbb{B}\big(0, B_A, \|\cdot\|_{1,2}\big)\Big) \\
\mathcal{L}_Q\big(\theta^{(t)}\big) &= \texttt{Unif}\Big(\mathbb{B}\big(1/2, 1/2, |\cdot|\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(1/2, 1/2, |\cdot|\big)\Big) \cdot \mathcal{L}_Q(A^{(t)}) \cdot \mathcal{L}_Q(W^{(t)}) \\
\mathcal{L}_Q(A^{(t)}) &= \texttt{Unif}\Big(\mathbb{B}\big(0, B_{A,1}, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(0, B_{A,2}, \|\cdot\|_{\mathbf{F}}\big)\Big) \\
\mathcal{L}_Q(W^{(t)}) &= \prod_{i=1}^{h} \texttt{Unif}\Big(\mathbb{B}\big(0, B_Q, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(0, B_K, \|\cdot\|_{\mathbf{F}}\big)\Big) \cdot \texttt{Unif}\Big(\mathbb{B}\big(0, B_V, \|\cdot\|_{\mathbf{F}}\big)\Big).
\end{aligned}$$

Then the KL divergence between P and Q is

$$\mathtt{KL}(P \,\|\, Q) = \mathcal{O}\Big((D^2 \cdot d \cdot (d_F + d_h + d) + d \cdot d_y) \cdot \log\big(1 + NT\tau^{-1} RhB_A B_{A,1} B_{A,2} B_Q B_K B_V\big)\Big). \tag{6.9}$$

Finally, we control term (II) in (6.4). This term can be controlled as

$$\begin{aligned}
&\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \log \frac{\mathbb{P}(x_{t+1}^n \,|\, S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n \,|\, S_t^n)} \\
&\quad = \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \log \frac{\mathbb{P}(x_{t+1}^n \,|\, S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n \,|\, S_t^n)} - \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbb{E}_{S_t^n} \mathtt{KL}\big(\mathbb{P}(\cdot \,|\, S_t^n) \,\|\, \mathbb{P}_{\theta^*}(\cdot \,|\, S_t^n)\big) \\
&\qquad + \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbb{E}_{S_t^n} \mathtt{KL}\big(\mathbb{P}(\cdot \,|\, S_t^n) \,\|\, \mathbb{P}_{\theta^*}(\cdot \,|\, S_t^n)\big).
\end{aligned}$$

The first two terms in the right-hand side of the equality is the generalization error, which can be bounded with Lemma 9.4. With Assumption 4.2, we note that

$$\left| \log \frac{\mathbb{P}(x \,|\, S)}{\mathbb{P}_{\theta^*}(x \,|\, S)} \right| \le b^* = \log \max\{c_0^{-1}, b_y^{-1}\}, \tag{6.10}$$

so the function satisfies the condition in Lemma 9.4 with $c_i = 2b^*$. Using the moment generating function bound in Lemma 9.4 and Chernoff bound, we have that

$$\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \log \frac{\mathbb{P}(x_{t+1}^n \,|\, S_t^n)}{\mathbb{P}_{\theta^*}(x_{t+1}^n \,|\, S_t^n)} - \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbb{E}_{S_t^n} \mathtt{KL}\big(\mathbb{P}(\cdot \,|\, S_t^n) \,\|\, \mathbb{P}_{\theta^*}(\cdot \,|\, S_t^n)\big) \le \sqrt{\frac{t_{\min} b^{*,2}}{2NT}} \log \frac{1}{\delta} \tag{6.11}$$

with probability at least $1 - \delta$.

Step 3: Conclude the proof.

Combining inequalities (6.4), (6.7), (6.9), and (6.11), we have that

$$\begin{aligned}
&\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathtt{TV}\big(\mathbb{P}(\cdot \,|\, S_t^n), \mathbb{P}_{\widehat{\theta}}(\cdot \,|\, S_t^n)\big) \\
&\quad \le \sqrt{\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathtt{TV}\big(\mathbb{P}(\cdot \,|\, S_t^n), \mathbb{P}_{\widehat{\theta}}(\cdot \,|\, S_t^n)\big)^2} \\
&\quad = \mathcal{O}\bigg(\frac{t_{\min}^{1/4}}{(NT)^{1/4}} \log \frac{1}{\delta} + \frac{\sqrt{D^2 d(d_F + d_h + d) + d \cdot d_y}}{\sqrt{NT}} \cdot \log\big(1 + NT\bar{B}\big) \\
&\qquad + \inf_{\theta^* \in \Theta} \sqrt{\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbb{E}_{S_t^n} \mathtt{KL}\big(\mathbb{P}(\cdot \,|\, S_t^n) \,\|\, \mathbb{P}_{\theta^*}(\cdot \,|\, S_t^n)\big)} \bigg),
\end{aligned}$$

where we take θ^* as the best approximation parameters. Finally, we will change the left-hand side of this inequality to the expectation of it. In fact, we have that

Proposition 6.2. Let $\mathcal{F}$ be the collection of functions of $f : \mathbb{R}^n \to \mathbb{R}$, and we assume that $|f| \le b$ for any function $f \in \mathcal{F}$. For a Markov chain $X = (X_1, \cdot, X_N)$, we define $f(X) = \sum_{i=1}^N f(X_i)/N$. The mixing time of this Markov chain is denoted as $t_{\text{mix}}(\varepsilon)$. Given a distribution Q on $\mathcal{F}$, with probability at least $1 - \delta$, we have

$$\Big| \mathbb{E}_P\Big[\mathbb{E}_X\big[f(X)\big] - f(X) \Big] \Big| \le \sqrt{\frac{b^2 \cdot t_{\min}}{2 \log 2 \cdot N}} \Big[\mathtt{KL}(P \,\|\, Q) + \log \frac{4}{\delta} \Big],$$

for any distribution P on $\mathcal{F}$ simultaneously with probability at least $1 - \delta$, where

$$t_{\min} = \inf_{0 \le \varepsilon < 1} t_{\text{mix}}(\varepsilon) \cdot \left(\frac{2 - \varepsilon}{1 - \varepsilon} \right)^2.$$

Proof of Proposition 6.2. See Appendix 8.2. □

We note that Proposition 6.2 is indeed an uniform convergence bound, since it holds simultaneously for all P. Thus, we can set P and Q as those in equalities (6.5) and (6.8), then we have that

$$\begin{aligned}&\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathbb{E}_{S_t^n}\Big[\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S_t^n),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S_t^n)\big)\Big]-\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S_t^n),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S_t^n)\big)\\&\quad=\mathcal{O}\bigg(\frac{\sqrt{t_{\min}}}{\sqrt{NT}}\Big(\bar{D}\log(1+NT\bar{B})+\log\frac{1}{\delta}\Big)\bigg).\end{aligned}$$

Thus, we have that

$$\begin{aligned}&\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathbb{E}_{S_t^n}\Big[\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S_t^n),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S_t^n)\big)\Big]\\&\quad=\mathcal{O}\bigg(\frac{t_{\min}^{1/4}}{(NT)^{1/4}}\log\frac{1}{\delta}+\frac{\sqrt{t_{\min}}}{\sqrt{NT}}\Big(\bar{D}\log(1+NT\bar{B})+\log\frac{1}{\delta}\Big)\\&\qquad+\inf_{\theta^*\in\Theta}\sqrt{\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\mathbb{E}_{S_t^n}\mathtt{KL}\big(\mathbb{P}(\cdot\,|\,S_t^n)\,\|\,\mathbb{P}_{\theta^*}(\cdot\,|\,S_t^n)\big)}\bigg).\end{aligned}$$

We conclude the proof of Theorem 4.3.

□

7 Proofs and Formal Statements for §5

7.1 Proof of Theorem 5.2

Proof. By Proposition 3.4 and the fact that $\log(1/p_0(z_*))\le\beta$, we have that

$$T^{-1}\cdot\mathbb{E}_{\mathcal{D}_{\texttt{ICL}}}\Big[\sum_{t=1}^{T}\log\mathbb{P}(r_t\,|\,z_*,\mathtt{pt}_{t-1})-\sum_{t=1}^{T}\log\mathbb{P}(r_t\,|\,\mathtt{pt}_{t-1})\Big]\le\beta/T. \tag{7.1}$$

In addition, we have that

$$T^{-1}\cdot\mathbb{E}_{\mathcal{D}_{\texttt{ICL}}}\Big[\sum_{t=1}^{T}\log\mathbb{P}(r_t\,|\,\mathtt{pt}_{t-1})-\sum_{t=1}^{T}\log\mathbb{P}_{\widehat{\theta}}(r_t\,|\,\mathtt{pt}_{t-1})\Big]=\mathbb{E}_{\mathcal{D}_{\texttt{ICL}}}\Big[\mathtt{KL}\big(\mathbb{P}(\cdot\,|\,\mathtt{pt})\,\Big\|\,\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,\mathtt{pt})\big)\Big]. \tag{7.2}$$

Similar to (6.10), we have that

$$\Big|\log\big(\mathbb{P}(r\,|\,\mathtt{pt})/\mathbb{P}_{\widehat{\theta}}(r\,|\,\mathtt{pt})\big)\Big|\le b^*=\log\max\{c_0^{-1},b_y^{-1}\}.$$

By Lemma 9.10, we have that

$$\mathtt{KL}\big(\mathbb{P}(\cdot\,|\,\mathtt{pt})\,\|\,\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,\mathtt{pt})\big)\le(3+b^*)/2\cdot\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,\mathtt{pt}),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,\mathtt{pt})\big). \tag{7.3}$$

By Assumption 5.1, we have that $\mathbb{P}_{\mathcal{D}_{\texttt{ICL}}}(\mathtt{pt})\le\kappa\mathbb{P}_{\mathcal{D}}(\mathtt{pt})$. Thus, by Theorem 4.3, we have with probability at least $1-\delta$ that

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}_{\texttt{ICL}}}\Big[\mathtt{KL}\big(\mathbb{P}(\cdot\,|\,\mathtt{pt})\,\|\,\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,\mathtt{pt})\big)\Big]\\&\quad\le C\cdot b^*\cdot\kappa\cdot\mathbb{E}_{S\sim\mathcal{D}}\Big[\mathtt{TV}\big(\mathbb{P}(\cdot\,|\,S),\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S)\big)\Big]\le C\cdot b^*\cdot\kappa\cdot\Delta_{\mathrm{pre}}(N,T,\delta).\end{aligned} \tag{7.4}$$

Combining (7.4), (7.1), and (7.2), we have with probability at least $1-\delta$ that

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}_{\text{ICL}}}\Big[T^{-1}\cdot\sum_{t=1}^{T}\log\mathbb{P}(r_t\,|\,z_*,\mathtt{pt}_{t-1})-T^{-1}\cdot\sum_{t=1}^{T}\log\mathbb{P}_{\widehat{\theta}}(r_t\,|\,\mathtt{pt}_{t-1})\Big]\\
&\quad\le\beta/T+\mathbb{E}_{S\sim\mathcal{D}}\Big[\mathtt{KL}\big(\mathbb{P}(\cdot\,|\,S)\,\|\,\mathbb{P}_{\widehat{\theta}}(\cdot\,|\,S)\big)\Big]\\
&\quad\le\mathcal{O}\big(\beta/T+b^*\cdot\kappa\cdot\Delta_{\text{pre}}(N,T,\delta)\big),
\end{aligned}\tag{7.5}$$

which completes the proof of Theorem 5.2. □

8 Proof of Supporting Propositions

8.1 Proof of Proposition 5.1

Proof. Let a, b be two vectors in the $(d-1)$-dimensional unit sphere $\mathbb{S}^{d-1}$. We first define the following vector,

$$c=(a^\top b)\cdot b-\big(a-(a^\top b)\cdot b\big)\in\mathbb{S}^{d-1}.\tag{8.1}$$

By direct calculation, we have the following property of c defined in (8.1),

$$c^\top b=(a^\top b)\cdot\|b\|_2^2-a^\top b+(a^\top b)\cdot\|b\|_2^2=a^\top b.\tag{8.2}$$

By (8.1) and (8.2), we have that

$$a+c=2(a^\top b)\cdot b=2(c^\top b)\cdot b=(a^\top b)\cdot b+(c^\top b)\cdot b.\tag{8.3}$$

We now calculate the desired integration. Note that

$$\int_{\mathbb{S}^{d-1}}a\cdot\exp(a^\top b)\mathrm{d}a=b\cdot\int_{\mathbb{S}^{d-1}}(a^\top b)\exp(a^\top b)\mathrm{d}a+\int_{\mathbb{S}^{d-1}}\big(a-(a^\top b)\cdot b\big)\cdot\exp(a^\top b)\mathrm{d}a.\tag{8.4}$$

For the second term on the right-hand side of (8.4), it follows from (8.1) and (8.2) and (8.3) that

$$\int_{\mathbb{S}^{d-1}}\big(a-(a^\top b)\cdot b\big)\cdot\exp(a^\top b)\mathrm{d}a=-\int_{\mathbb{S}^{d-1}}\big(c-(c^\top b)\cdot b\big)\cdot\exp(c^\top b)\mathrm{d}c,\tag{8.5}$$

where the equality follows from the fact that $\mathrm{d}c=2\|b\|_2^2\mathrm{d}a-\mathrm{d}a=\mathrm{d}a$. By replacing c by a on the right-hand side of (8.5), we have

$$\int_{\mathbb{S}^{d-1}}\big(a-(a^\top b)\cdot b\big)\cdot\exp(a^\top b)\mathrm{d}a=-\int_{\mathbb{S}^{d-1}}\big(a-(a^\top b)\cdot b\big)\cdot\exp(a^\top b)\mathrm{d}a=0\tag{8.6}$$

Finally, by plugging (8.6) into (8.4), we obtain that

$$\int_{\mathbb{S}^{d-1}}a\cdot\exp(a^\top b)\mathrm{d}a=b\cdot\int_{\mathbb{S}^{d-1}}(a^\top b)\exp(a^\top b)\mathrm{d}a.$$

Thus, by setting

$$C_1=\int_{\mathbb{S}^{d-1}}(a^\top b)\exp(a^\top b)\mathrm{d}a,\quad\forall b\in\mathbb{S}^{d-1},$$

we complete the proof of Proposition 5.1. Note that here C_1 is an absolute constant that does not depend on b due to the symmetry on the unit sphere. □

8.2 Proof of Proposition 6.2

Proof of Proposition 6.2. We note that $f(X)$ satisfies the condition in Lemma 9.4 with $c_i = 2b/N$ for $i \in [N]$. Then Lemma 9.4 shows that

$$\mathbb{E}_{f\sim P_0}\Big[\mathbb{E}_X\Big(\exp\big[\lambda(f(X)-\mathbb{E}f(X))\big]\Big)\Big] \le \exp\left(\frac{\lambda^2\cdot b^2\cdot t_{\min}}{2N}\right).$$

Take $\lambda = \sqrt{2N\log 2/(b^2 t_{\min})}$. The Markov inequality shows that

$$P\left(\mathbb{E}_{f\sim P_0}\Big(\exp\big[\lambda(f(X)-\mathbb{E}f(X))\big]\Big) \ge \frac{2}{\delta}\right) \le \delta$$

for any $0<\delta<1$. We note that this probability inequality does not involve P. Take the function g in Lemma 9.3 as $g(f) = \lambda(f(X)-\mathbb{E}f(X))$, then it shows that

$$\log \mathbb{E}_{P_0}\Big[\exp\big(g(X)\big)\Big] + \mathtt{KL}(P\,\|\,P_0) \ge \mathbb{E}_P\big[g(X)\big]$$

for any P simultaneously. Combining these inequalities, we have

$$\Big|\mathbb{E}_P\Big[\mathbb{E}_X\big[f(X)\big]-f(X)\Big]\Big| \le \sqrt{\frac{b^2\cdot t_{\min}}{2\log 2N}}\Big[\mathtt{KL}(P\,\|\,P_0)+\log\frac{4}{\delta}\Big],$$

for any distribution P on $\mathcal{F}$ simultaneously with probability at least $1-\delta$. Thus, we conclude the proof of Proposition 6.2. □

8.3 Proof of Proposition 6.1

Proof of Proposition 6.1 . We analyze the error layer by layer in the neural network. Denote the outputs of each layer in the networks parameterized by θ and $\widetilde{\theta}$ as $X^{(t)}$ and $\widetilde{X}^{(t)}$, respectively. In the final layer, we have that

$$\begin{aligned}
&\mathtt{TV}\big(P_\theta(\cdot\,|\,X), P_{\widetilde{\theta}}(\cdot\,|\,X)\big)\\
&\quad\le 2\left\|\frac{1}{L\tau}\mathbb{I}_L^\top X^{(D)}A^{(D+1)} - \frac{1}{L\tau}\mathbb{I}_L^\top \widetilde{X}^{(D)}\widetilde{A}^{(D+1)}\right\|_\infty\\
&\quad\le \frac{2}{\tau}\Big[\big\|A^{(D+1),\top}\big\|_{1,2}\cdot\big\|X^{(D),\top}-\widetilde{X}^{(D),\top}\big\|_{2,\infty} + \big\|A^{(D+1),\top}-\widetilde{A}^{(D+1),\top}\big\|_{1,2}\Big],
\end{aligned}$$

where the first inequality results from Lemma 9.6, and the second inequality results from Lemma 9.7 and that $\|X^{(D),\top}\|_{2,\infty}\le 1$ due to the layer normalization. In the following, we build the recursion relationship between $\|X^{(t),\top}-\widetilde{X}^{(t),\top}\|_{2,\infty}$ for $t\in[D]$.

$$\begin{aligned}
&\|X^{(t+1),\top}-\widetilde{X}^{(t+1),\top}\|_{2,\infty}\\
&\quad\le \big\|\mathtt{ffn}(Y^{(t+1)},A^{(t+1)})^\top - \mathtt{ffn}(\widetilde{Y}^{(t+1)},\widetilde{A}^{(t+1)})^\top\big\|_{2,\infty} + |\gamma_2^{(t+1)}-\widetilde{\gamma}_2^{(t+1)}| + \big\|Y^{(t+1),\top}-\widetilde{Y}^{(t+1),\top}\big\|_{2,\infty}\\
&\quad\le |\gamma_2^{(t+1)}-\widetilde{\gamma}_2^{(t+1)}| + \big\|Y^{(t+1),\top}-\widetilde{Y}^{(t+1),\top}\big\|_{2,\infty} + B_{A,1}\cdot B_{A,2}\cdot\|Y^{(t+1),\top}-\widetilde{Y}^{(t+1),\top}\|_{2,\infty}\\
&\qquad + B_{A,2}\cdot\|A_1^{(t+1)}-\widetilde{A}_1^{(t+1)}\|_{\mathbf{F}} + B_{A,1}\cdot\|A_2^{(t+1)}-\widetilde{A}_2^{(t+1)}\|_{\mathbf{F}},
\end{aligned}\tag{8.7}$$

where the first inequality results from the triangle inequality and that Π_{norm} is not expansive, the second inequality results from the following proposition

Proposition 8.1. For any $X,\widetilde{X}\in\mathbb{R}^{L\times d}$, $A_1,\widetilde{A}_1\in\mathbb{R}^{d\times d_F}$, and $A_2,\widetilde{A}_2\in\mathbb{R}^{d_F\times d}$, we have that

$$\begin{aligned}
&\big\|\mathtt{ffn}(X,A)^\top - \mathtt{ffn}(\widetilde{X},\widetilde{A})^\top\big\|_{2,\infty}\\
&\quad\le \|A_1\|_{\mathbf{F}}\cdot\|A_2\|_{\mathbf{F}}\cdot\|X^\top-\widetilde{X}^\top\|_{2,\infty} + \|A_1-\widetilde{A}_1\|_{\mathbf{F}}\cdot\|A_2\|_{\mathbf{F}}\cdot\|\widetilde{X}^\top\|_{2,\infty}\\
&\qquad + \|\widetilde{A}_1\|_{\mathbf{F}}\cdot\|A_2-\widetilde{A}_2\|_{\mathbf{F}}\cdot\|\widetilde{X}^\top\|_{2,\infty}.
\end{aligned}$$

Proof of Proposition 8.1. See Appendix 8.4. □

Next, we build the relationship between $\|Y^{(t+1),\top} - \widetilde{Y}^{(t+1),\top}\|_{2,\infty}$ in the right-hand side of inequality (8.7) and $\|X^{(t),\top} - \widetilde{X}^{(t),\top}\|_{2,\infty}$.

$$
\begin{aligned}
&\|Y^{(t+1),\top} - \widetilde{Y}^{(t+1),\top}\|_{2,\infty} \\
&\quad \le \big\|\mathtt{mha}(X^{(t)}, W^{(t+1)})^\top - \mathtt{mha}(\widetilde{X}^{(t)}, \widetilde{W}^{(t+1)})^\top\big\|_{2,\infty} + |\gamma_1^{(t+1)} - \widetilde{\gamma}_1^{(t+1)}| + \big\|X^{(t),\top} - \widetilde{X}^{(t),\top}\big\|_{2,\infty} \\
&\quad \le |\gamma_1^{(t+1)} - \widetilde{\gamma}_1^{(t+1)}| + \big\|X^{(t),\top} - \widetilde{X}^{(t),\top}\big\|_{2,\infty} \\
&\qquad + h \cdot B_V\big(1 + 4B_Q B_K\big)\|X^{(t),\top} - \widetilde{X}^{(t),\top}\|_{2,\infty} + \sum_{i=1}^{h} \big\|W_i^{V,(t+1)} - \widetilde{W}_i^{V,(t+1)}\|_{\mathbf{F}} \\
&\qquad + 2B_V \cdot B_K \sum_{i=1}^{h} \|W_i^{Q,(t+1)} - \widetilde{W}_i^{Q,(t+1)}\|_{\mathbf{F}} + 2B_V \cdot B_Q \sum_{i=1}^{h} \|W_i^{K,(t+1)} - \widetilde{W}_i^{K,(t+1)}\|_{\mathbf{F}},
\end{aligned} \tag{8.8}
$$

where the first inequality results from the triangle inequality, and the second inequality results from Lemma 9.8. Combining inequalities (8.7) and (8.8), we derive that

$$
\begin{aligned}
&\|X^{(t+1),\top} - \widetilde{X}^{(t+1),\top}\|_{2,\infty} \\
&\quad \le (1 + B_{A,1} \cdot B_{A,2})\big(1 + hB_V(1 + 4B_Q B_K)\big)\|X^{(t),\top} - \widetilde{X}^{(t),\top}\|_{2,\infty} + \beta_{t+1} + \iota_{t+1} + \kappa_{t+1} + \rho_{t+1}.
\end{aligned}
$$

This concludes the proof of Proposition 6.1. □

8.4 Proof of Proposition 8.1

Proof of Proposition 8.1. We have that

$$
\begin{aligned}
&\big\|\mathtt{ffn}(X, A)^\top - \mathtt{ffn}(\widetilde{X}, \widetilde{A})^\top\big\|_{2,\infty} \\
&\quad \le \max_{i\in[L]} \Big[\big\|\mathtt{ReLU}(X_{i,:}A_1)A_2 - \mathtt{ReLU}(\widetilde{X}_{i,:}A_1)A_2\big\|_2 + \big\|\mathtt{ReLU}(\widetilde{X}_{i,:}A_1)A_2 - \mathtt{ReLU}(\widetilde{X}_{i,:}\widetilde{A}_1)\widetilde{A}_2\big\|_2\Big] \\
&\quad \le \max_{i\in[L]} \Big[\|A_1\|_{\mathbf{F}} \cdot \|A_2\|_{\mathbf{F}} \cdot \|X_{i,:} - \widetilde{X}_{i,:}\|_2 + \big\|\mathtt{ReLU}(\widetilde{X}_{i,:}A_1)A_2 - \mathtt{ReLU}(\widetilde{X}_{i,:}\widetilde{A}_1)A_2\big\|_2 \\
&\qquad + \big\|\mathtt{ReLU}(\widetilde{X}_{i,:}\widetilde{A}_1)A_2 - \mathtt{ReLU}(\widetilde{X}_{i,:}\widetilde{A}_1)\widetilde{A}_2\big\|_2\Big] \\
&\quad \le \max_{i\in[L]} \Big[\|A_1\|_{\mathbf{F}} \cdot \|A_2\|_{\mathbf{F}} \cdot \|X_{i,:} - \widetilde{X}_{i,:}\|_2 + \|A_1 - \widetilde{A}_1\|_{\mathbf{F}} \cdot \|A_2\|_{\mathbf{F}} \cdot \|\widetilde{X}_{i,:}\|_2 \\
&\qquad + \|\widetilde{A}_1\|_{\mathbf{F}} \cdot \|A_2 - \widetilde{A}_2\|_{\mathbf{F}} \cdot \|\widetilde{X}_{i,:}\|_2\Big],
\end{aligned}
$$

where the first inequality results from the triangle inequality, the second and the last inequalities result from Lemma 9.7 and that `ReLU` is not expansive. Thus, we conclude the proof of Proposition 8.1. □

9 Technical Lemmas

Lemma 9.1 (Caponnetto and De Vito (2007))**.** Let (Ω, ν) be a probability space and ξ be a random variable on Ω taking value in a real separable Hilbert space $\mathcal{H}$. We assume that there exists constants $B, \sigma > 0$ such that

$$
\big\|\xi(w)\big\|_{\mathcal{H}} \le B/2, \text{ a.s.}, \quad \mathbb{E}\big[\|\xi\|_{\mathcal{H}}^2\big] \le \sigma^2.
$$

Then, it holds with probability at least $1 - \delta$ that

$$
\left\|L^{-1}\sum_{i=1}^{L} \xi(\omega_i) - \mathbb{E}[\xi]\right\| \le 2\left(\frac{B}{L} + \frac{\sigma}{\sqrt{L}}\right)\log\frac{2}{\delta}.
$$

Lemma 9.2 (Proposition 4.5 in Duchi (2019)). Let $\mathcal{F}$ be the collection of functions of $f : \mathbb{R}^n \to \mathbb{R}$. For any $f \in \mathcal{F}$, we define

$$\mu(f) = \mathbb{E}_X\big[f(X)\big], \quad \sigma^2(f) = \mathbb{E}_X\big[(f(X) - \mathbb{E}_X[f(X)])^2\big],$$

where the expectation is taken with respect to a random variable $X \sim \nu$ on $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$. Assume that $|f(X) - \mu(f)| \le b$ a.s. for some constant $b \in \mathbb{R}$ for all $f \in \mathcal{F}$. Then for any $0 < \lambda \le 1/(2b)$, given a distribution P_0 on $\mathcal{F}$, with probability at least $1 - \delta$, we have

$$\left|\mathbb{E}_Q\left[\mathbb{E}_X[f(X)] - \frac{1}{n}\sum_{i=1}^n f(X_i)\right]\right| \le \lambda \mathbb{E}_Q\big[\sigma^2(f)\big] + \frac{1}{n\lambda}\left[\mathtt{KL}(Q \,\|\, P_0) + \log\frac{2}{\delta}\right],$$

for any distribution Q on $\mathcal{F}$, where X_i are i.i.d. samples of ν. If the function class $\mathcal{F}$ further satisfies $\sigma^2(f) \le c\mu(f)$ for some constant $c \in \mathbb{R}$ for all $f \in \mathcal{F}$, we have

$$\left|\mathbb{E}_Q\left[\mathbb{E}_X\big[f(X)\big] - \frac{1}{n}\sum_{i=1}^n f(X_i)\right]\right| \le \lambda c \mathbb{E}_Q\big[\mu(f)\big] + \frac{1}{n\lambda}\left[\mathtt{KL}(Q \,\|\, P_0) + \log\frac{2}{\delta}\right],$$

with probability at least $1 - \delta$.

Lemma 9.3 (Donsker–Varadhan representation in Belghazi et al. (2018)). Let P and Q be distributions on a common space $\mathcal{X}$. Then

$$\mathtt{KL}(P \,\|\, Q) = \sup_{g \in \mathcal{G}}\left\{\mathbb{E}_P\big[g(X)\big] - \log \mathbb{E}_Q\Big[\exp\big(g(X)\big)\Big]\right\},$$

where $\mathcal{G} = \{g : \mathcal{X} \to \mathbb{R} \mid \mathbb{E}_Q[\exp(g(X))] < \infty\}$.

Lemma 9.4 (Corollary 2.11 in Paulin (2015)). Let $X = (X_1, \cdots, X_N)$ be a Markov chain, taking values in $\Lambda = \prod_{i=1}^N \Lambda_i$ with mixing time $t_{\mathrm{mix}}(\varepsilon)$ for $\varepsilon \in [0, 1]$. Let

$$t_{\min} = \inf_{0 \le \varepsilon < 1} t_{\mathrm{mix}}(\varepsilon) \cdot \left(\frac{2-\varepsilon}{1-\varepsilon}\right)^2.$$

If function $f : \Lambda \to \mathbb{R}$ is such that $f(x) - f(y) \le \sum_{i=1}^N c_i \mathbb{I}_{x_i \ne y_i}$ for every $x, y \in \Lambda$, then for any $\lambda \in \mathbb{R}$,

$$\log \mathbb{E}\Big(\exp\big[\lambda(f(X) - \mathbb{E}f(X))\big]\Big) \le \frac{\lambda^2 \cdot \|c\|_2^2 \cdot t_{\min}}{8}.$$

For any $t \ge 0$, we have

$$P\Big(\big|f(X) - \mathbb{E}f(X)\big| \ge t\Big) \le 2\exp\left(\frac{-2t^2}{\|c\|_2^2 \cdot t_{\min}}\right).$$

Lemma 9.5 (Lemma 25 in Agarwal et al. (2020)). For any two conditional probability densities $P(\cdot \,|\, X), P'(\cdot \,|\, X)$ and any distribution $\nu \in \Delta(\mathcal{X})$,we have

$$\mathbb{E}_\nu\Big[\mathtt{TV}\big(P(\cdot \,|\, X), P'(\cdot \,|\, X)\big)^2\Big] \le -2\log\left(\mathbb{E}_{X\sim\nu, Y\sim P(\cdot\,|\,X)}\left[\exp\left(-\frac{1}{2}\log\frac{P(Y \,|\, X)}{P'(Y \,|\, X)}\right)\right]\right).$$

Lemma 9.6 (Corollary A.7 in Edelman et al. (2021)). For any $x, y \in \mathbb{R}^d$, we have

$$\|\mathtt{softmax}(x) - \mathtt{softmax}(y)\|_1 \le 2\|x - y\|_\infty.$$

Lemma 9.7 (Lemma 17 in Zhang et al. (2022a)). Given any two conjugate numbers $u, v \in [1, \infty]$, i.e., $\frac{1}{u} + \frac{1}{v} = 1$, and $1 \le p \le \infty$, for any $A \in \mathbb{R}^{r\times c}$ and $x \in \mathbb{R}^c$, we have

$$\|Ax\|_p \le \|A\|_{p,u}\|x\|_v \quad \text{and} \quad \|Ax\|_p \le \|A^\top\|_{u,p}\|x\|_v.$$

Lemma 9.8 (Propositions 20 and 21 in Zhang et al. (2022a))**.** For any $X, \widetilde{X} \in \mathbb{R}^{L\times d}$, and any $W_i^Q, \widetilde{W}_i^Q, W_i^K, \widetilde{W}_i^K \in \mathbb{R}^{d\times d_h}, W_i^V, \widetilde{W}_i^V \in \mathbb{R}^{d\times d}$ for $i \in [h]$, if $\|X^\top\|_{p,\infty}, \|\widetilde{X}^\top\|_{2,\infty} \le B_X$, $\|W_i^Q\|_{\mathbf{F}}, \|\widetilde{W}_i^Q\|_{\mathbf{F}} \le B_Q$, $\|W_i^K\|_{\mathbf{F}}, \|\widetilde{W}_i^K\|_{\mathbf{F}} \le B_K$, $\|W_i^V\|_{\mathbf{F}}, \|\widetilde{W}_i^V\|_{\mathbf{F}} \le B_V$ for $i \in [h]$, then we have

$$\begin{aligned}
&\Big\|\big(\mathtt{mha}(X, W) - \mathtt{mha}(\widetilde{X}, \widetilde{W})\big)^\top\Big\|_{2,\infty} \\
&\quad\le h \cdot B_V\big(1 + 4B_X^2 \cdot B_Q B_K\big)\|X^\top - \widetilde{X}^\top\|_{2,\infty} + B_X \sum_{i=1}^h \big\|W_i^V - \widetilde{W}_i^V\|_{\mathbf{F}} \\
&\qquad + 2B_X^3 \cdot B_V \cdot B_K \sum_{i=1}^h \|W_i^Q - \widetilde{W}_i^Q\|_{\mathbf{F}} + 2B_X^3 \cdot B_V \cdot B_Q \sum_{i=1}^h \|W_i^K - \widetilde{W}_i^K\|_{\mathbf{F}}.
\end{aligned}$$

Lemma 9.9 (Lemma A.6 in Elbrächter et al. (2021))**.** For $a, b \in \mathbb{R}$ with $a < b$, let

$$\mathcal{S}_{[a,b]} = \Big\{f \in \mathcal{S}^\infty([a,b], \mathbb{R}) \,|\, \big\|f^{(n)}(x)\big\| \le n! \text{ for all } n \in \mathbb{N}\Big\}.$$

There exists a constant $C > 0$ such that for all $a, b \in \mathbb{R}$ with $a < b$, $f \in \mathcal{S}_{[a,b]}$, and $\varepsilon \in (0, 1/2)$, there is a fully connect network Ψ_f such that

$$\|f - \Psi_f\|_\infty \le \varepsilon,$$

with the depth of the network as $D(\Psi_f) \le C\max\{2, b-a\}(\log \varepsilon^{-1})^2 + \log(\lceil\max\{|a|, |b|\}\rceil) + \log(\lceil 1/(b-a)\rceil)$, the width of the network as $W(\Psi_f) \le 16$, and the maximal weight in the network as $B(\Psi_f) \le 1$.

Lemma 9.10. Let $b = \sup_x \log(p(x)/q(x))$. We have that

$$\mathtt{KL}(p \,\|\, q) \le 2(3 + b) \cdot \mathtt{TV}(p, q). \tag{9.1}$$

Proof. We let $f(t) = \log t$ and $g(t) = |1/t - 1|$. Then, for $0 \le t \le \exp(b)$, we have that

$$\sup_{0\le t\le \exp(b)} \frac{f(t)}{g(t)} = \sup_{0\le t\le \exp(b)} \frac{\log t}{|1/t - 1|} = \sup_{1\le t\le \exp(b)} \frac{t\log t}{t-1} \le 2(b+3).$$

Note that $\mathtt{KL}(p \,\|\, q) = \mathbb{E}_p[f(p(x)/q(x))]$ and $\mathtt{TV}(p, q) = \mathbb{E}_p[g(p(x)/q(x))]$, which concludes the proof. □