A Statistical Framework for Data-dependent Retrieval-Augmented Models

Soumya Basu [*1] **Ankit Singh Rawat** [*2] **Manzil Zaheer** [*3]

Abstract

Modern ML systems increasingly augment input instances with additional relevant information to enhance final prediction. Despite growing interest in such retrieval-augmented models, their fundamental properties and training are not well understood. We propose a statistical framework to study such models with two components: 1) a *retriever* to identify the relevant information out of a large corpus via a data-dependent metric; and 2) a *predictor* that consumes the input instances along with the retrieved information to make the final predictions. We present a principled method for end-to-end training of both components and draw connections with various training approaches in the literature. Furthermore, we establish excess risk bounds for retrieval-augmented models while delineating the contributions of both retriever and predictor towards the model performance.We validate the utility of our proposed training methods along with the key takeaways from our statistical analysis on open domain question answering task where retrieval augmentation is important.

1. Introduction

Recent advancements in machine learning (ML) have not only led to breakthroughs on long-standing challenging tasks across various fields, but they have also inspired a great deal of interest to develop ML models that can solve even harder tasks (Meinhardt et al., 2022; Lewkowycz et al., 2022; Cramer, 2021) or focus on completely new fields (Austin et al., 2021; OpenAI, 2023; Singhal et al., 2023). While scaling the size of *parametric* ML models, such as neural networks, is becoming the predominant approach to meet such demands (Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023; Dosovitskiy et al., 2021; Dehghani et al., 2023), the excellent performance realized by this approach is marred by drawbacks such as high computational cost, inefficient storage of world knowledge in parameters, lack of transparency in model behavior, and reduced grounding/factuality of model predictions.

Recognizing these shortcomings, *retrieval-augmented models* (RAMs) have emerged as a promising alternative. Such models typically employ two components, namely *retriever* and *predictor*, during inference on a given input instance: The retriever first identifies instance-specific relevant information from a data-store, and then the predictor jointly processes the retrieved information and the input instance to make a final prediction. In practice, RAMs have enjoyed favorable performance vs. compute trade-off (Borgeaud et al., 2021; Das et al., 2021; Thai et al., 2023) as employing moderate-size parametric models as retriever and predictor in a RAM often matches or exceeds the performance of a much larger standalone ML model that directly maps input instances to predictions. Similarly, conditioning prediction on the retrieved information has shown to exhibit improved grounding (Shuster et al., 2021; Lin et al., 2023; Asai et al., 2023). Furthermore, having access to an external corpus can obviate the need to store task-specific world knowledge in model parameters and enable incorporating dynamically evolving knowledge (Izacard et al., 2022; Liska et al., 2022).

Despite these desirable characteristics, training RAMs presents multiple challenges. The natural approach of independently training retriever and predictor can be suboptimal (Izacard et al., 2022). Moreover, it requires collecting intermediate supervision on the instance-dependent relevant information to retrieve, which is missing in common datasets and expensive to obtain in general. A common strategy to circumvent the lack of intermediate supervision is to perform end-to-end training which presents its own unique challenges in the context of RAMs. Fundamentally, the retrieval corresponds to the non-differentiable discrete operation of selecting relevant information from a data-store, e.g., via top-k selection based on retriever scores, which prevents direct gradient propagation to the entire receiver. Several clever solutions to above-mentioned issues have been proposed in the literature that focus on different training objectives to propagate learning signal from the predictor into the retriever. However, a formal study that unifies these solutions is missing from the literature.

[*]Equal contribution; in alphabetical order [1]Google, New York, USA [2]Google Research, New York, USA [3]Google DeepMind, New York, USA. Correspondence to: Soumya Basu <basusoumya@google.com>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Another key challenge that prevents the resource-efficient development and deployment of RAMs is the limited understanding of their basic properties such as their generalization behavior and expressive power. For instance, how do the retriever and predictor components interact to ensure good task-specific performance? Are there any principles guiding the selection of the retriever and predictor components? How does (size of) the data-store feature in the final performance of a RAM?

In this paper, we address both aforementioned shortcoming in the literature pertaining RAMs. To unify the training of RAMs, we begin with writing down the natural objective function, which somehow has eluded the literature. This natural objective simply minimizes the expected prediction loss, where the expectation is taken over the distribution induced by the retriever. Empirically, we find this objective to be effective on standard benchmarks: NaturalQuestions (NQ; Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017).

As for the generalization and expressive power, we present an excess risk bound for RAMs that captures the effect of retrieval and prediction function classes. The proposed bound allows us to highlight how retriever and predictor components play complementary roles to reduce approximation error as we increase their respective function class complexity. We also capture the role of data store in improving the model performance by reducing the approximation error. On the generalization front, we carefully decouple the generalization term in the excess risk over the predictor and retriever function classes. This allows us to tightly control the generalization term with only logarithmic dependence on the data store size. As a concrete instantiation for our excess risk bounds, we consider feed-forward neural networks of varying depth for both the retriever and the predictor.

To summarize, our main contributions include:

- We present a principled objective for end-to-end training of RAMs focusing on a classification setting (Sec. 2 3) and draw connections between existing approaches for training RAMs (Sec. 3.6).
- We derive excess risk bound highlighting the role played by retriever and predictor functions classes as well as the data-store towards ensuing improved performance by RAMs (Sec. 3.4); capturing the trade off between model capacities at retriever and predictor (Sec. 3.5).
- We validated the utility of the proposed objective on two standard QA benchmarks: NaturalQuestions (NQ) and TriviaQA (Sec. 4).

2. Problem setup

In this paper, we focus on developing a systematic understanding of RAMs with learned retrievers in a classification setting where the model has access to a data-store. Towards this, we begin by formally defining the problem setup and providing the necessary background along with the notations used.

Let's first consider the standard classification setting which requires predicting a class in $\mathcal{Y}$ for a given instance $x \in \mathcal{X}$. Assume that D_{XY} captures the underlying data distribution and one has access to n training examples $\mathcal{S}_n \triangleq \{(x_i, y_i)\}_{i\in[n]}$ that are independent and identically distributed (i.i.d.) according to D_{XY}. Given $\mathcal{S}_n$, one hopes to learn a classifier $f : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|}$ that minimizes the miss-classification error:

$$R(f) = \mathbb{P}_{(X,Y)\sim\mathsf{D}_{XY}}\big[\arg\max_{y\in\mathcal{Y}} f^y(X) \neq Y\big], \quad (1)$$

where $f^y(x)$ denotes the score that f assigns to the y-th class, given the input instance x. Since directly optimizing the miss-classification error or $0/1$-loss poses computational challenges, one typically selects the classifier that minimizes the empirical risk associated with a well behaved surrogate loss function $\ell : \mathbb{R}^{|\mathcal{Y}|} \times \mathcal{Y} \to \mathbb{R}$ on the training sample $\mathcal{S}_n$:

$$R_{\ell,n}(f) = \frac{1}{n}\sum_{i\in[n]} \ell\big(f(x_i), y_i\big). \quad (2)$$

The (population) risk associated with the surrogate loss function takes the following form:

$$R_\ell(f) = \mathbb{E}_{(X,Y)\sim\mathsf{D}_{XY}}\big[\ell\big(f(X), Y\big)\big]. \quad (3)$$

Different from the standard classification setup described above, we now consider the classification task with access to a data-store: Given an instance x, the classifier can potentially leverage a data-store $\mathcal{I} \subseteq \mathcal{Z}$ – a collection of potentially relevant information or evidences, where $\mathcal{Z}$ denotes the space of all possible evidences. Accordingly, one can define the empirical and population risks of a classifier $f(\cdot, \mathcal{I}) : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|}$ as follows:

$$R_{\ell,\mathcal{I},n}(f) = \frac{1}{n}\sum_{i\in[n]} \ell\big(f(x_i, \mathcal{I}), y_i\big), \quad (4)$$

$$R_{\ell,\mathcal{I}}(f) = \mathbb{E}\big[\ell\big(f(X, \mathcal{I}), Y\big)\big], \quad (5)$$

where expectation is take over in $(X, Y) \sim \mathsf{D}_{XY}$ as well as the possible randomness in $f(\cdot, \mathcal{I})$. However, due its prohibitive computational cost, such a general classifier that directly processes the entire data-store for each prediction is far from how an additional data-store is utilized by ML models in practice.

This motivates us to study the following explicit retrieval-augmented classification setup to utilize the data-store: Given an input instance $x \in \mathcal{X}$, one first retrieves input-dependent supporting evidences $\mathcal{E}^x \subset \mathcal{I}$ with the help of a *retriever model* which has access to the entire data-store $\mathcal{I}$.

Now, given x and $\mathcal{E}^x$, one invokes a *predictor model* to predict the class associated with x. Thus, a retriever-augmented classification setup consists of two key components models: 1) retriever model and 2) predictor model, which we formally introduce next.

Retriever model. For the retrieval stage, we rely on a *retriever model* to capture the relevance of an evidence $z \in \mathcal{I}$ towards the input instance $x \in \mathcal{X}$. Let $r_\theta : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ be the retriever model parameterized by $\theta \in \Theta$ that assigns a relevance score $r_\theta(x, z)$ to the instance-evidence pair (x, z). Furthermore, for each instance x, the retriever model r_θ induces the following distribution over the set of potential evidences:

$$p_{\theta,\mathcal{I}}\big(z|x\big) = \frac{\exp\big(r_\theta(x,z)\big)}{\sum_{z' \in \mathcal{I}} \exp\big(r_\theta(x,z')\big)}, \quad \forall\, z \in \mathcal{I}. \tag{6}$$

There are multiple strategies to construct the set of input-dependent supporting evidences $\mathcal{E}^x$ based on r_θ. For example, for a fixed integer $k \geq 1$, one could select k evidences corresponding to the k highest scores in $\{r_\theta(x,z)\}_{z\in\mathcal{I}}$. Another strategy is to sample k evidences according to the distribution $p_{\theta,\mathcal{I}}(\cdot|x)$ in (6). Here, one could perform the sampling with or without replacement. In what follows, we denote the retrieved supporting evidence for the instance x as $\mathcal{E}^x_\theta$ to highlight the dependence on the underlying retriever model.

Predictor model. Let $h_\xi : \mathcal{X} \times \mathcal{I}^* \to \mathbb{R}^{|\mathcal{Y}|}$ be the predictor model parameterized by $\xi \in \Xi$, where $\mathcal{I}^*$ denotes the Kleene star on $\mathcal{I}$. Given $x \in \mathcal{X}$ and $\mathcal{E} \in \mathcal{I}^*$, the predictor model h_ξ assigns a score to each class in $\mathcal{Y}$, defining a distribution over $\mathcal{Y}$ as follows:

$$p_\xi\big(y|x,\mathcal{E}\big) = \frac{\exp\big(h^y_\xi(x,\mathcal{E})\big)}{\sum_{y' \in \mathcal{Y}} \exp\big(h^{y'}_\xi(x,\mathcal{E})\big)}, \quad \forall\, y \in \mathcal{Y}, \tag{7}$$

where $h^y_\xi(\cdot,\cdot)$ denotes the score assigned to the y-th class by the predictor model h_ξ.

For ease of exposition, we focus on the setting with $k = |\mathcal{E}^x_\theta| = 1, \forall x \in \mathcal{X}$, in our analysis throughout this paper. This corresponds to retrieving a single supporting evidence for each input instance. Our analysis can be generalized to $k > 1$ by working with a $\tilde{\mathcal{I}} \subseteq \mathcal{I}^k$ as the new data-store and $\tilde{p}_{\theta,\mathcal{I}}(\cdot|x)$ as a distribution over $\tilde{\mathcal{I}}$ obtained by suitably modifying $p_{\theta,\mathcal{I}}$ in (6). For example, when k supporting evidences are sampled with replacement, then the following holds $\forall (z_1, \ldots, z_k) \in \mathcal{I}^k$.

$$\tilde{p}_{\theta,\mathcal{I}}\big((z_1,\ldots,z_k)|x\big) = \prod_{j\in[k]} p_{\theta,\mathcal{I}}(z_j|x).$$

Empirical risk minimization and excess risk for RAMs. For a pair of retriever and predictor models parameterized by θ and ξ, respectively, we can define the empirical and population risks associated with a (surrogate) loss function ℓ as follows:

$$R_{\ell,\mathcal{I},n}(\xi,\theta) = \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x)\ell\big(h_\xi(x_i,z),y_i\big), \tag{8}$$

$$R_{\ell,\mathcal{I}}(\xi,\theta) = \mathbb{E}\big[\ell\big(h_\xi(X,\mathcal{E}^X_\theta),Y\big)\big]. \tag{9}$$

Note that the expectation in (9) is taken over $(X,Y) \sim \mathsf{D}_{XY}$ as well as the randomness involved in the retrieval stage, e.g., sampling the evidences according to $p_{\theta,\mathcal{I}}(\cdot|x)$ in (6). Given a pair of predictor class Ξ and retriever class Θ, let $(\hat{\xi},\hat{\theta})$ denote the predictor-retriever pair obtained via *empirical risk minimization* (ERM) as follows:

$$(\hat{\xi},\hat{\theta}) \in \operatorname*{arg\,min}_{(\xi,\theta)\in\Xi\times\Theta} R_{\ell,\mathcal{I},n}(\xi,\theta). \tag{10}$$

Let $\mathcal{F}_{\text{all}}$ denote the set of all measurable functions from $\mathcal{X}\times\mathcal{Z}$ to $\mathbb{R}^{|\mathcal{Y}|}$. The optimal risk for the classification with access to the data-store is achieved by the best possible predictor $f^\ell_{\text{opt},\mathcal{I}} \in \mathcal{F}$ when it has access to the best retrieved evidence in $\mathcal{I}$. In particular, we have

$$f^\ell_{\text{opt},\mathcal{I}} = \operatorname*{arg\,min}_{f\in\mathcal{F}_{\text{all}}} \mathbb{E}\big[\min_{z\in\mathcal{I}} \ell(f(X,z),Y)\big]. \tag{11}$$

Given $f^\ell_{\text{opt},\mathcal{I}}$, we defined the *excess* risk of a predictor-retriever pair (ξ,θ) as follows:

$$\begin{aligned}\Delta_{\ell,\mathcal{I}}(\xi,\theta) &= R_{\ell,\mathcal{I}}(\xi,\theta) - R_{\ell,\mathcal{I}}(f^\ell_{\text{opt},\mathcal{I}}) \\ &\triangleq R_{\ell,\mathcal{I}}(\xi,\theta) - \mathbb{E}\big[\min_{z\in\mathcal{I}} \ell(f^\ell_{\text{opt},\mathcal{I}}(X,z),Y)\big].\end{aligned} \tag{12}$$

With the formal definition of the classification setting with access to a data-store and the necessary background in place, we proceed to address the two key objectives of this work: 1) Proposing a natural and efficient joint end-to-end training procedure for the predictor-retriever pair in a RAM; and 2) Developing a rigorous statistical understanding of RAMs focusing on the interaction between predictor and retriever components towards reducing overall excess risk.

3. Joint training and excess risk

Recall that training a RAM involves training both the retriever $r_\theta : \mathcal{X}\times\mathcal{Z} \to \mathbb{R}$ and the predictor $h_\xi : \mathcal{X}\times\mathcal{I} \to \mathbb{R}^{|\mathcal{Y}|}$ components of the model without access to intermediate supervision on retrieval, which is infeasible to obtain in most practical settings. Thus, it becomes critical to devise methods to jointly train r_θ and h_ξ with access to only labeled instances $\mathcal{S}_n = \{(x_i,y_i)\}_{i\in[n]} \subseteq \mathcal{X}\times\mathcal{Y}$ with the predictor guiding the retriever training based on how valuable the retriever-provided evidences are towards the correct final prediction.

Towards this, we leverage the empirical risk from (8) along with the log-loss $\ell(h_\xi(x,z),y) = -\log p_\xi(y|x,z)$, where $p_\xi(y|x,z)$ is defined in (7). In particular, this leads to the following joint end-to-end training objective:

$$\begin{aligned}\mathcal{L}_n(\xi,\theta;\mathcal{I}) &\triangleq R_{\log,\mathcal{I},n}(\xi,\theta) \\ &= -\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_{\theta,\mathcal{I}}(z|x_i)\cdot\log p_\xi(y_i|x_i,z).\end{aligned} \tag{13}$$

Note that the objective in (13) aims to improve the end-to-end performance of a RAM in deployment in the sense that the objective aims to minimize the expected loss given the selected evidences as per the retriever-induced distribution. One can use gradient-based methods to jointly minimize the objective in (13) with respect to (ξ, θ); however, its efficient implementation is non-trivial due to the sum over entire data-store $\mathcal{I}$. In App. C.1, we discuss some approximate design choices. Lastly, please refer to Sec. 3.6 for connections between our proposed objective in (13) and some of the existing end-to-end training approaches for RAMs.

Next, to study the generalization and expressive power of RAMs, we want to bound the excess risk $\Delta_{\ell,\mathcal{I}}(\hat{\xi}, \hat{\theta})$ as defined in (12). We consider $\mathcal{X}$ to be a compact subspace of $\mathbb{R}^{d_x}$ and, for simplicity, take $\mathcal{X} \subseteq [-1, 1]^{d_x}$. Similarly, we consider that each retrieval example $z \in \mathcal{I}$ is embedded in the space $[-1, 1]^{d_z}$. We consider a data-store that polynomially scales with training data size, i.e., $|\mathcal{I}| = \text{poly}(n)$. For the purpose of analysis, we specialize our log-loss to be bounded by $\ell_{\max} > 0$, which is given as

$$\ell(h_\xi(x, z), y) = \min(\ell_{\max}, -\log p_\xi(y|x, z)) \tag{14}$$
$$= \min\Big(\ell_{\max}, \log\Big(\sum_{y' \in \mathcal{Y}} \exp(h_\xi^{y'}(x, z))\Big) - h_\xi^y(x, z)\Big),$$

where $p_\xi(y|x, z)$ and $h_\xi^y(x, z)$ are defined in (7).

3.1. Excess risk decomposition

Our excess risk relies on separating out the contribution coming from the retriever and the predictor during the joint training. Moreover, the retriever and predictor errors can be each split into generalization and approximation error.

The population risk optimizer of our joint training over the space $\Xi \times \Theta$ is defined as

$$\xi^*_{\text{joint}}, \theta^*_{\text{joint}} = \underset{(\xi,\theta)\in\Xi\times\Theta}{\arg\min}\ \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}\mathbb{E}_{Y|X}\ell\big(h_\xi(X, Z), Y\big)\big].$$

For a predictor ξ, sample $x \in \mathcal{X}$ and retrieved example $z \in \mathcal{I}$, let us denote the risk averaged over the labels $\mathcal{Y}$ as

$$g_\xi(x, z) = \mathbb{E}_{Y|X=x}[\ell\big(h_\xi(x, z), Y\big)]. \tag{15}$$

For any fixed predictor ξ (*not necessarily* in Ξ) and fixed data-store $\mathcal{I}$, the retriever that optimizes the joint population risk is given as $p^{*,\xi}(z|x) = \mathbb{1}_{\arg\min_{z'\in\mathcal{I}} g_\xi(x,z')}(z)$, where a tie is broken arbitrarily. Note that, for each sample x, the best retrieved evidence z may change. We define the optimal predictor *within the class* Ξ with best possible retriever as

$$\xi^* = \underset{\xi\in\Xi}{\arg\min}\, \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X, z)\big].$$

The optimal retriever *within the class* Θ for a given predictor ξ is defined as

$$\theta(\xi) = \underset{\theta\in\Theta}{\arg\min}\, \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_\xi(X, Z)\big].$$

The excess risk for the classes Θ and Ξ can be bounded as

$$\begin{aligned}
&\Delta_{\ell,\mathcal{I}}(\hat{\xi}, \hat{\theta}) \\
&\le \underbrace{\sum_{(\theta,\xi)\in\{(\hat{\theta},\hat{\xi}),(\theta^*_{\text{joint}},\xi^*_{\text{joint}})\}} |R_{\ell,\mathcal{I}}(\xi, \theta) - R_{\ell,\mathcal{I},n}(\xi, \theta)|}_{\text{Generalization Error}} \\
&\quad + \underbrace{R_{\ell,\mathcal{I}}(\xi^*, \theta(\xi^*)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X, z)\big]}_{\text{retriever error}} \\
&\quad + \underbrace{\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X, z)\big] - R_{\ell,\mathcal{I}}(f^\ell_{\text{opt},\mathcal{I}})}_{\text{predictor error}}
\end{aligned} \tag{16}$$

3.2. Generalization error

We first bound the generalization error and relate it to the covering number of the retriever and predictor class.

As our loss is bounded by $\ell_{\max}$, through standard concentration bounds (Shalev-Shwartz & Ben-David, 2014), we obtain that, for any $\delta > 0$, with probability at least $(1 - \delta)$:

$$|R_{\ell,\mathcal{I}}(\xi^*_{\text{joint}}, \theta^*_{\text{joint}}) - R_{\ell,\mathcal{I},n}(\xi^*_{\text{joint}}, \theta^*_{\text{joint}})| \le 3\ell_{\max}\sqrt{\tfrac{\log(1/\delta)}{n}}.$$

However, $(\hat{\xi}, \hat{\theta})$ is learned from the data. A high probability generalization error requires taking union over the space of $\Xi \times \Theta$. We employ Rademacher complexity based generalization error bounds. Next, the covering number of the space Ξ is used to bound the associated Rademacher complexity. See Shalev-Shwartz & Ben-David (2014) for details.

We define two norms which are used in defining the covering numbers for Θ and Ξ. In particular, $\forall \mathbf{u} \in \mathbb{R}^{n\times|\mathcal{I}|}$ and fixed $\xi \in \Xi, \theta \in \Theta$,

$$\|\mathbf{u}\|_{2,[n],\xi} = \Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} u_{i,z}\ell\big(h_\xi(x_i, z), y_i\big)\big)^2\Big)^{1/2},$$
$$\|\mathbf{u}\|_{2,[n],\theta} = \Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} p_\theta(z|x_i)u_{i,z}\big)^2\Big)^{1/2}. \tag{17}$$

We also define $\mathcal{N}(\Xi, \nu, \|\cdot\|_{2,[n],\theta})$ to be the ν-covering number for the class Ξ with respect to the norm $\|\cdot\|_{2,[n],\theta}$, and $\mathcal{N}(\Theta, \nu, \|\cdot\|_{2,[n],\xi})$ to be the ν-covering number for the class Θ with respect to the norm $\|\cdot\|_{2,[n],\xi}$. Then we

have the generalization bound given as

$$|R_{\ell,\mathcal{I}}(\hat{\xi},\hat{\theta}) - R_{\ell,\mathcal{I},n}(\hat{\xi},\hat{\theta})| \leq \inf_{\varepsilon\in[0,\ell_{\max}/2]} \Big(8\varepsilon + \tfrac{24}{\sqrt{n}} \int_{\varepsilon}^{\frac{\ell_{\max}}{2}} f_{\mathcal{N}}(\nu/2;\Theta,\Xi) + f_{\mathcal{N}}(\nu/2;\Xi,\Theta)d\nu\Big), \tag{18}$$

for $f_{\mathcal{N}}(\nu;\mathcal{A},\mathcal{B}) = \sup_{b\in\mathcal{B}} \sqrt{\log(\mathcal{N}(\mathcal{A},\nu,\|\cdot\|_{2,[n],b}))}$.

We use ideas in Zhang (2023) to upper bound the covering number with pseudo-dimension (defined in the Appendix A) of the function class. This allows us to have a $\log|\mathcal{I}|$ dependence in the generalization error, while working with norm unbounded function classes.

3.3. Approximation error

We next proceed to bound the retriever and predictor approximation errors. Towards this, we extensively use the Sobolev functions spaces. A Sobolev space for a domain Ω is characterized by two quantities, κ – the number of weak-derivatives a (real-valued) function within it possesses, and $L_p(\Omega)$ – the norm with respect to which these derivatives are integrable. Please see Appendix A for a complete definition.

3.3.1. Retriever error

The retriever error is given by how well the score function $r_\theta(x,z)$ approximates the optimal retriever given ξ^*. In order to do so we first need to impose some smoothness constraints on the function $g_{\xi^*}:\mathcal{X}\times\mathcal{Z}\to\mathbb{R}$. In particular, we assume the following.

Assumption 3.1 (Complexity of g_{ξ^*})**.** There exists a baseline function $b_{\xi^*}:[-1,1]^{d_x}\to\mathbb{R}$ such that the function $\mathrm{gap}_{\xi^*}:[-1,1]^{d_x+d_z}\to\mathbb{R}$ defined by $\mathrm{gap}_{\xi^*}(x,z)\triangleq(g_{\xi^*}(x,z)-b_{\xi^*}(x))$ lies in the Sobolev space with κ derivatives and $L_\infty([-1,1]^{d_x+d_z})$ norm.

The above assumption says that for the predictor ξ^* the loss profile (averaged over labels in $\mathcal{Y}$) $g_{\xi^*}(x,z)$, has two components – a (possibly) complex $b_{\xi^*}(x)$ component that is uniform over z, and a 'smooth' $\mathrm{gap}_{\xi^*}(x,z)$ component. In other words, given two similar retrieved evidences, the predictor incurs similar losses when each of the evidences is utilized with an input instance.

Then, for any $\tau>0$, we can bound the retriever loss as follows:

$$R_{\ell,\mathcal{I}}(\xi^*,\theta(\xi^*)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X,z)\big] \leq \inf_{\theta\in\Theta} \ell_{\max}\|r_\theta+\tau\cdot\mathrm{gap}_{\xi^*}\|_\infty + \frac{\log|\mathcal{I}|}{\tau^2} \tag{19}$$

3.3.2. Predictor error

The predictor error is measured with the optimal retrieval (as the retriever error is considered separately above). For this, we need to first quantify how the retrieval augmentation using the data-store $\mathcal{I}$ helps.

Usefulness of retrieval set: We start with characterization of the prediction task in the presence of the data-store $\mathcal{I}\subset\mathcal{Z}$. We assume that there exists a score function $h_*:\mathcal{X}\times\mathcal{Z}\to\mathbb{R}^{|\mathcal{Y}|}$, and the corresponding probability distribution

$$p^y_*(x,z) = \frac{\exp(h^y_*(x,z))}{\sum_{y'}\exp(h^{y'}_*(x,z))}, \tag{20}$$

that approximates $p^y_{\mathsf{D}_{XY}}(x) := \mathbb{P}_{Y\sim\mathsf{D}_{Y|X}}(y|X=x)$ well for all $x\in\mathcal{X}$ and $y\in\mathcal{Y}$. Furthermore, we want this score function h_* to lie coordinate wise in a Sobolev space. The following assumption formalizes this.

Assumption 3.2 (Retrieval quality)**.** There exists a score function $h_*:\mathcal{X}\times\mathcal{Z}\to\mathbb{R}^{|\mathcal{Y}|}$ such that

1. for each $y\in\mathcal{Y}$, the function h^y_* lies in the Sobolev space with $\kappa_{\mathcal{I}}$ derivatives and finite $L_\infty([-1,1]^{d_x+d_z})$ norm,
2. for any $x\in\mathcal{X}$, there exists a retrieved evidence $z^*(x)\in\mathcal{I}$ such that $p^y_*(x,z)$, as defined in (20), satisfies
$$\max_{y\in\mathcal{Y}}\sup_{x\in\mathcal{X}} |p^y_*(x,z^*(x)) - p^y_{\mathsf{D}_{XY}}(x)| \leq c_{\mathcal{I}}|\mathcal{I}|^{-\gamma_{\mathcal{I}}}.$$

Note that this is independent of the retriever class Θ and Ξ, and captures intrinsic property of the data-store $\mathcal{I}$. The tuple $(\gamma_{\mathcal{I}}, d_z, \kappa_{\mathcal{I}})$ defines the usefulness of $\mathcal{I}$. In particular, the higher $\gamma_{\mathcal{I}}$ the closer the approximation; and the higher the $\kappa_{\mathcal{I}}$ and smaller the embedding dimension d_z the 'simpler' the score function used for this approximation.

Under the Assumption 3.2, we bound the predictor error as

$$\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X,z)\big] - R_{\ell,\mathcal{I}}(f^\ell_{\mathrm{opt},\mathcal{I}}) \leq \inf_{\xi\in\Xi} 2\mathbb{E}_X\big[\max_{y\in\mathcal{Y}} |h^y_\xi(X,z^*(X)) - h^y_*(X,z^*(X))|\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{I}}|\mathcal{I}|^{-\gamma_{\mathcal{I}}}. \tag{21}$$

One key step in arriving to the above inequality is expressing the loss of $f^\ell_{\mathrm{opt},\mathcal{I}}$ using the probability function h_* defined in Assumption 3.2. In particular, under Assumption 3.2, we show that

$$\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{f^\ell_{\mathrm{opt},\mathcal{I}}}(X,z)\big] \geq \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] - (|\mathcal{Y}|-1)\exp(-\ell_{\max}) - c_{\mathcal{I}}|\mathcal{I}|^{-\gamma_{\mathcal{I}}}.$$

3.4. Final excess risk bound

We now combine the three components of the excess risk bounds under Assumptions 3.1 and 3.2 and discuss the design tradeoffs. The following theorem captures our main theoretical result.

Theorem 3.3 (Excess risk of joint training). *Under Assumption 3.1 and 3.2, the excess risk for the retriever class Θ and predictor class Ξ is bounded as*

$$\Delta_{\ell,\mathcal{J}}(\hat{\xi},\hat{\theta}) \le 3\ell_{\max}(\tfrac{1}{n}+\sqrt{\tfrac{\log(n)}{n}})+$$
$$\inf_{\varepsilon\in[0,\frac{\ell_{\max}}{2}]} 8\varepsilon + \tfrac{24}{\sqrt{n}}\int_{\varepsilon}^{\frac{\ell_{\max}}{2}} f_{\mathcal{N}}(\tfrac{\nu}{2};\Theta,\Xi)+f_{\mathcal{N}}(\tfrac{\nu}{2};\Xi,\Theta)d\nu$$
$$+\inf_{\theta\in\Theta}\inf_{\tau>0}\ell_{\max}\|r_\theta+\tau\cdot\mathrm{gap}_{\xi^*}\|_\infty+\frac{\log|\mathcal{J}|}{\tau^2}$$
$$+\inf_{\xi\in\Xi} 2\mathbb{E}_X\big[\max_{y\in\mathcal{Y}}|h^y_\xi(X,z^*(X))-h^y_*(X,z^*(X))|\big]+$$
$$(|\mathcal{Y}|-1)\exp(-\ell_{\max})+c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}},$$

where $f_{\mathcal{N}}(\nu;\mathcal{A},\mathcal{B}) \triangleq \sup_{b\in\mathcal{B}}\sqrt{\log(\mathcal{N}(\mathcal{A},\nu,\|\cdot\|_{2,[n],b}))}$ and $\|\cdot\|_{2,[n],\theta}$ and $\|\cdot\|_{2,[n],\xi}$ are defined in (17).

3.5. Illustrative example: MLPs

We instantiate both our retriever and predictor classes to be multi-layer perceptron (MLP) with depth L_{ret} & width $W_{\text{ret}} = O(d_x+d_z)$ and depth L_{pred} & width $W_{\text{pred}} = O(|\mathcal{Y}|(d_x+d_z))$, respectively. The class MLP $\left(\mathbb{R}^d,\mathbb{R}^k;L,W\right)$ is defined in Appendix A. The specialized excess risk bound for this setting is given as

Theorem 3.4 (Excess risk for MLP). *Under Assumption 3.1 and 3.2, the excess risk for the retriever class $\Theta = MLP\left(\mathbb{R}^{d_x+d_z},\mathbb{R};L_{\text{pred}},O(d_x+d_z)\right)$ and predictor class $\Xi = MLP\left(\mathbb{R}^{d_x+d_z},\mathbb{R}^{|\mathcal{Y}|};L_{\text{pred}},O(|\mathcal{Y}|(d_x+d_z))\right)$ is bounded as*

$$\Delta_{\ell,\mathcal{J}}(\hat{\xi},\hat{\theta}) \le \tilde{O}\left(\frac{\ell_{\max}}{\sqrt{n}}\left(L_{\text{ret}}+L_{\text{pred}}|\mathcal{Y}|\right)\right)+$$
$$O\Big(\ell_{\max}L_{\text{ret}}^{-\frac{4\kappa}{3(d_x+d_z)}}\log^{1/3}(|\mathcal{J}|)\Big)+$$
$$O\left(L_{\text{pred}}^{-\frac{2\kappa_{\mathcal{J}}}{(d_x+d_z)}}+(|\mathcal{Y}|-1)\exp(-\ell_{\max})+c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}\right).$$

Finally, to capture the optimal trade-off under finite data size n, we consider classes of retriever and predictors that change with the data size, denoted by Θ_n and Ξ_n, with growing depths $L_{\text{ret},n}$ and $L_{\text{pred},n}$ respectively. Similarly, we also consider growing upper bound on the loss function by $\ell_{\max,n}$. Let $d_{\text{tot}} = d_x+d_z$. For $L_{\text{ret},n} = n^{\frac{3d_{\text{tot}}}{6d_{\text{tot}}+8\kappa}}$, $L_{\text{pred},n} = (\sqrt{n}/|\mathcal{Y}|)^{\frac{d_{\text{tot}}}{2d_{\text{tot}}+4\kappa_{\mathcal{J}}}}$, and $\ell_{\max,n} = \log|\mathcal{Y}| + \frac{\kappa_{\mathcal{J}}}{(d_{\text{tot}}+2\kappa_{\mathcal{J}})}\log n$, the excess risk is bounded by

$$O\left(n^{-\frac{2\kappa}{3d_{\text{tot}}+4\kappa}}+\max\left(|\mathcal{J}|^{-\gamma_{\mathcal{J}}},\left(\frac{n}{|\mathcal{Y}|^2}\right)^{-\frac{\kappa_{\mathcal{J}}}{d_{\text{tot}}+2\kappa_{\mathcal{J}}}}\right)\right).$$

We should contrast the above result with the prediction when there is no retrieval. Let us assume that the functions

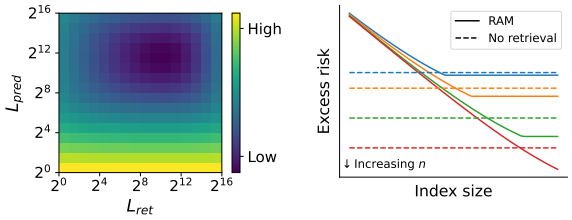

Figure 1. **Left**: Excess risk bound as we vary retriever and predictor size for a fixed n and $\mathcal{J}$ based on Theorem 3.4. Note that different size combination of predictor and retriever achieves same risk bound. **Right**: Excess risk bound of RAM as we increase data-store size in contrast to direct MLP predictor with no retrieval. We plot for various values of n, with each cure corresponding to a fixed n.

$p^y_{\mathsf{D}_{XY}}(x)$ for all $y\in\mathcal{Y}$ lies in the Sobolev space with derivative κ_{true} and L_∞ norm. The predictor excess risk rate with $L_{\text{pred},n} = (\sqrt{n}/|\mathcal{Y}|)^{\frac{d_x}{d_x+2\kappa_{\text{true}}}}$ is $O((n/|\mathcal{Y}|^2)^{-\frac{\kappa_{\text{true}}}{d_x+2\kappa_{\text{true}}}})$.

Note that our analysis indicates that we may potentially *gain through retrieval*: For large enough retrieval set $|\mathcal{J}| \ge (n/|\mathcal{Y}|^2)^{\frac{\kappa_{\mathcal{J}}\gamma_{\mathcal{J}}^{-1}}{d_{\text{tot}}+2\kappa_{\mathcal{J}}}}$, as the data size n increases and we have $\kappa > \frac{3d_{\text{tot}}}{2d_x}\kappa_{\text{true}}$ & $\kappa_{\mathcal{J}} > \frac{d_{\text{tot}}}{d_x}\kappa_{\text{true}}$ (see Fig. 1).

3.6. Connections with prior end-to-end training

We conclude our treatment of end-to-end training of RAMs by drawing parallels between our proposed method with some representative approaches from the literature.

EMDR^2 Sachan et al. (2021) minimize the following objective based on the negative log-likelihood:

$$\mathcal{L}_n^{\text{EMDR}^2}(\xi,\theta;\mathcal{J}) = -\frac{1}{n}\sum_{i\in[n]}\log p_{\xi,\theta,\mathcal{J}}(y|x)$$
$$= -\frac{1}{n}\sum_{i\in[n]}\log\Big(\sum_{z\in\mathcal{J}}p_{\theta,\mathcal{J}}(z|x_i)\cdot p_\xi(y_i|x_i,z)\Big). \quad (22)$$

It follows from the convexity of $-\log(\cdot)$ and Jensen's inequality that our objective in (13) upper bounds the EMDR^2 objective in (22); as a result, minimizing the former also minimizes the latter but not vice versa.

Perplexity distillation (PDist) Another approach for joint training of RAMs in the literature involves optimizing two distinct objectives for training the predictor and retriever components. For example, Izacard et al. (2022) propose multiple objectives for retriever training, including PDist (Sachan et al., 2023) which is defined as follows:

$$\mathcal{L}_{\mathcal{J},n}^{\text{PDIST}}(\theta;\xi,\mathcal{J}) =$$
$$\frac{1}{n}\sum_{i\in[n]}\text{CE}\big(p_{\xi,\mathcal{J}}^{\text{PDIST}}(Z|x_i,y_i),p_{\theta,\mathcal{J}}(Z|x_i)\big), \quad (23)$$

where $\text{CE}(\cdot,\cdot)$ denotes the cross entropy between two dis-

Method	small			base			large		
	small	base	large	small	base	large	small	base	large
No retriever, train predictor ξ									
Cross-Entropy		19.6			25.5			29.1	
Fixed retriever θ_0, train predictor ξ									
Cross-Entropy	23.2	26.6	28.3	27.5	32.4	34.7	32.2	36.4	37.8
Fixed predictor $\xi^\star(\theta_0)$, train retriever θ									
EMDR2	23.9	28.5	31.0	29.2	34.2	36.6	33.4	38.0	40.8
PDist	30.1	34.5	38.4	34.0	39.7	42.8	37.6	42.8	44.7
Cross-Entropy + PG	25.9	30.6	31.7	31.5	36.4	37.9	36.0	40.2	41.4
Cross-Entropy + TopK	29.4	35.5	37.9	33.8	39.7	43.0	37.2	42.3	45.0
Jointly train predictor ξ and retriever θ									
EMDR2	24.1	30.4	32.7	30.4	35.6	39.3	34.5	39.7	42.1
PDist	28.7	33.2	36.6	33.3	37.1	38.8	36.2	40.2	41.6
Cross-Entropy + PG	27.1	31.0	32.7	33.3	37.2	38.2	36.5	39.8	41.4
Cross-Entropy + TopK	32.8	37.8	40.1	36.6	41.8	44.8	38.8	43.8	46.4

Table 1. **Exact match accuracy on NQ**. We measure the performance of RAMs across various training paradigms and model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

tributions and

$$p^{\text{PDIST}}_{\xi,\mathcal{J}}(z|x,y) = p_\xi(y|x,z)/\sum_{z'\in\mathcal{J}} p_\xi(y|x,z') \quad \forall\, z\in\mathcal{J},$$

represents a predictor-assigned distribution over evidences based on their utility towards making correct prediction. As for the predictor training, they optimize an objective akin to (13) with respect to ξ. Besides this similarity in the predictor training, our approach for retrieval training has a subtle connection with PDist. Note that PDist optimizes *forward cross-entropy* between the predictor and the retriever induced distributions to train the retriever. On the other hand, our objective in (13) is closer to $\frac{1}{n}\sum_i \text{CE}(p_{\theta,\mathcal{J}}(Z|x_i), p^{\text{PDIST}}_{\xi,\mathcal{J}})$, the reversed cross-entropy between the two distributions. The former has the "mean-seeking" behavior whereas the latter has the "mode-seeking" behavior (Huszár, 2015; Gu et al., 2023; Agarwal et al., 2023).

Similarity with RLHF/RLAIF Note that the per-example objective of our retrieval training approach takes the form:

$$\mathbb{E}_{Z\sim p_{\theta,\mathcal{J}}(\cdot|x_i)}\big[\ell\big(h_\xi(x_i,Z),y_i\big)\big], \tag{24}$$

i.e., the predictor model provides feedback on the (value) of the evidences sampled by the retriever model. Alternatively, one can view $-\ell\big(h_\xi(x_i,Z),y_i\big)$ as the reward assigned to the evidence z by the predictor model h_ξ and retriever model aims to select those evidences that maximize this reward value. This is similar to RLHF (Ziegler et al., 2019) or RLAIF (Bai et al., 2022) paradigm, where the underlying LLM aims to sample those generations which maximize the reward assigned by a reward model. However, note that in RLHF/RLAIF paradigm the policy network and reward model are not jointly trained together unlike in RAM.

4. Experiments

There have been numerous successful practical applications of RAMs in the literature (e.g., Sachan et al. (2021); Izacard et al. (2022)). Here, we present a brief empirical study for such models in order to corroborate the benefits predicted by our theoretical results. In particular, we consider the task of open-domain question answering and show that proposed objective is competitive to the objectives proposed in the literature and observe the trade-offs in model capacity between retriever and predictor model.

Data Our evaluation is based on two benchmark datasets: NQOpen (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017), which serve as sources for supervised examples (x, y), while chunked Wikipedia 2018 is used as the data-store $\mathcal{J}$ following literature (Karpukhin et al., 2020a). Consistent with established practices, we employ the exact match metric to assess the correspondence between the predicted answers and the ground truth. Additionally, we introduce a recall metric to measure the frequency at which the answer string appears within the retrieved documents.

Models We implement the retriever component using GTR (Ni et al., 2022) and the predictor component using T5 (Raffel et al., 2020). We sweep across small, base, and large configurations for both retriever and predictor. The details regarding the model sizes, expressed in terms of the number of parameters, are provided in Table 6 (App. C).

Methods We compare following approaches: 1) utilizing no retriever, directly training predictor, 2) employing a fixed retriever, but training the predictor, 3) using a fixed predictor,

Method	small			base			large		
	small	base	large	small	base	large	small	base	large
No retriever, train predictor ξ									
Cross-Entropy		17.9			23.1			28.0	
Fixed retriever θ_0, train predictor ξ									
Cross-Entropy	31.5	34.9	38.8	37.0	40.6	44.4	43.4	45.9	49.7
Fixed predictor $\xi^\star(\theta_0)$, train retriever θ									
EMDR2	34.6	41.3	48.3	40.1	48.2	53.4	46.0	50.7	54.9
PDist	45.7	53.3	57.2	50.8	53.2	61.6	53.5	55.4	62.3
Cross-Entropy + PG	43.2	46.7	54.3	48.6	56.1	55.1	51.7	56.4	56.7
Cross-Entropy + TopK	43.6	50.4	54.4	48.6	54.9	58.5	52.1	56.6	60.3
Jointly train predictor ξ and retriever θ									
EMDR2	37.0	43.1	49.7	42.4	50.5	55.6	47.1	53.4	59.2
PDist	46.7	54.3	57.3	48.8	56.7	60.7	51.0	58.5	63.3
Cross-Entropy + PG	47.0	52.9	55.7	49.9	57.6	61.1	52.1	59.8	59.2
Cross-Entropy + TopK	46.8	52.9	56.0	49.2	56.6	60.1	52.3	58.8	62.4

Table 2. **Exact match accuracy on TriviaQA**. We measure the performance of RAMs across various training paradigms and model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

but training the retriever, and 4) conducting joint training of both components. For the joint training and the retriever training phases, we experiment with multiple objectives: EMDR2 (cf. (22)), PDist (cf. (23)), Cross-Entropy + PG (cf. (40) in App. C.1), and Cross-Entropy + TopK (cf. (39) in App. C.1). Efficiently implementing any of these objectives is challenging due to the need to compute the gradient with respect to expectation over the entire data-store. We consider two approaches for computing the gradients approximately by: 1) restricting the expectation to top-K elements similar to EMDR2 and PDist; and 2) using REINFORCE (Williams, 1992) to obtain an unbaised estimate. More details can be found in App. C.1.

Observation 1 The addition of a retrieval component markedly enhances performance, as demonstrated in Tables 1 and 2, which present the exact match accuracy. Further improvements are observed when the retriever is specifically trained while keeping the predictor fixed. Joint training emerges as the most effective strategy.

Observation 2 Tables 4 and 5 (App. C) list the recall for the presence of the answer string within the retrieved content. PDist consistently achieves the highest recall, aligning with expectations given its design for distilling the retriever based on the predictor's scores. However, despite its superior recall, other objectives may lead to better overall performance than PDist, suggesting that different objectives optimize the retriever and predictor with varying efficiencies.

Observation 3 Finally, in Table 3, we report the query per second (QPS), as a proxy for computational cost, achieved by different configuration of retriever and predictor model sizes. For achieving a specific accuracy threshold (e.g., $\geq$38.8 on NQ), multiple configurations are viable, such as pairing a large predictor with a small retriever, a base model for both, or a small predictor with a large retriever. The associated query per second (QPS) rates for these configurations are 135, 333, and 800, respectively, illustrating that equivalent accuracy levels can be attained with significantly differing QPS rate. This corroborates with our trade-offs in excess risk bounds for MLPs with different capacity in retriever and predictor components as illustrated in Figure 1. Thus, adding capacity to different parts of the model has different repercussion on quality and computational cost.

5. Discussion and related work

Several works have proposed some form of retrieval augmented models. Here, we provide a brief account of the evolution of RAMs and discuss how our proposed joint-learning objective and the framework for excess risk analysis compare with existing end-to-end training methods.

Augment with local neighborhood The first approaches dating back to 1970s employed just augmenting training instance in the local neighborhood of the input space (Stone, 1977; 1980). Such approaches gained a lot of attention as parametric regression was not adequate in various practical applications of the time. This line of work aims to fit a low-degree polynomial at each point in the data set based on a subset of data points, which resulted in a rich literature on local polynomial regression in low dimensions. (Katkovnik & Kheisin, 1979; Cleveland, 1979; Pinsker, 1980; Donoho & Liu, 1988; Ruppert & Wand, 1994; Ibragimov & Has Min-

small			base			large		
small	base	large	small	base	large	small	base	large
822.60	819.83	800.89	334.30	333.22	331.06	135.06	135.34	134.87

Table 3. **Query per second**. We measure the query per second processed by RAMs as a proxy for computational cost across various model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

skii, 2013). These classical ideas have found their application in many ML algorithms such as face recognition (Jain & Learned-Miller, 2011), dimensionality reduction via local linear embeddings (Roweis & Saul, 2000), domain adaptation (Yang et al., 2021), test time training on neighboring points (Sun et al., 2020; Gandelsman et al., 2022), etc. Recently, Basu et al. (2023) generalized this setup of augmenting with a local neighborhood of the input instance in the context of modern ML models like neural networks and proposed a statistical framework to study such retrieval-based models. However, they *do not* consider a learned or a specialized distance metric to find the augmenting set, which is critical for realizing good performance in practice (Schonberger et al., 2017; Karpukhin et al., 2020b) and studied in the present work.

Fixed retriever augmentation Next generation retrieval augmented models started to deploy either a hand crafted or a learned retriever. Zhang et al. (2006) employed SIFT (Lowe, 1999) based retrieval followed by a SVM (Cortes & Vapnik, 1995) classifier to improve performance on multiple vision tasks. Chen et al. (2009) studied generalization bounds for SVM-kNN methods – one of the limited works in this domain with formal analysis. For natural language understanding, methods like TF-IDF (Sparck Jones, 1972) were employed in the tasks like case based reasoning (Leake et al., 1996) and open-domain question answering (ODQA; Voorhees et al. 1999). Unlike many previous methods, one retrieves relevant text passages in ODQA settings as opposed to retrieving labelled training pairs. With introduction of transformers (Vaswani et al., 2017), both retriever and predictor models based on encoder and decoder, respectively, have become popular across various domains, including image classification (Long et al., 2022; Iscen et al., 2023), text classification (Wang et al., 2022; Zemlyanskiy et al., 2022), ODQA (Lee et al., 2019; Izacard & Grave, 2021), language modelling (Borgeaud et al., 2021), and even protein folding prediction (Cramer, 2021). Even using the same transformer model as both retriever and predictor boosts performance in language modeling (Khandelwal et al., 2020). Unlike SVM-kNN (Chen et al., 2009), to best of our knowledge, a formal analysis of retrieval-augmented approaches with modern neural networks is missing from the literature. Interestingly, retrieving examples also helps in-context learning (Rubin et al., 2022; Li et al., 2023). Our framework covers this scenario with z representing the in-context examples retrieved from a datastore of examples. Our risk bounds can provide insights into why in-context learning with *retrieved* few-shot examples performs better than a zero-shot model.

End-to-end trained retriever augmentation For ODQA, Guu et al. (2020) proposed maximizing the marginalized likelihood by considering the retrieved set as a latent variable. EMDR2 (Sachan et al., 2021) optimized the same objective by approximating it based on the retriever induced distribution on the elements that receive top-K scores by the retriever. Hindsight (Paranjape et al., 2022) instead optimizes the ELBO by introducing a variational distribution with access to the outputs. VOD (Liévin et al., 2023) further generalized the standard ELBO based on KL divergence by employing Rényi divergence thereby tightening the lower bound. On the other hand, Atlas (Izacard et al., 2022) proposed an auxiliary loss for training the retriever directly rather than following the latent variable approach. Interestingly, RAG (Lewis et al., 2020) proposed to only train the query encoder for retriever, leaving the retrieval index fixed, thereby alleviating much of the end-to-end training difficulties of RAMs, but at cost of limiting model adaptation flexibility. None of these prior works studied statistical properties vis-à-vis expressivity and generalization of RAMs.

6. Conclusion

In this work, we initiate the development of a theoretical framework to study the statistical properties of RAMs with data-dependent retrieval. Our excess-risks analysis allows us to highlight how retriever and predictor components play complementary roles in reducing approximation error as we increase their respective function class complexity. We surface both theoretically and empirically a Pareto surface achieving the same performance with different size predictors and retrievers. As future work, it would be interesting to study the effect of dynamically updatable data-store and multi-step retrievals for making predictions.

Impact statement

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

References

Agarwal, R., Vieillard, N., Stanczyk, P., Ramos, S., Geist, M., and Bachem, O. Gkd: Generalized knowledge distillation for auto-regressive sequence models. *arXiv preprint arXiv:2306.13649*, 2023.

Asai, A., Wu, Z., Wang, Y., Sil, A., and Hajishirzi, H. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*, 2023.

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Bartlett, P. L., Harvey, N., Liaw, C., and Mehrabian, A. Nearly-tight vc-dimension and pseudodimension bounds for piecewise linear neural networks. *The Journal of Machine Learning Research*, 20(1):2285–2301, 2019.

Basu, S., Rawat, A. S., and Zaheer, M. A statistical perspective on retrieval-based models. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 1852–1886. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/basu23a.html.

Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., van den Driessche, G., Lespiau, J., Damoc, B., Clark, A., de Las Casas, D., Guy, A., Menick, J., Ring, R., Hennigan, T., Huang, S., Maggiore, L., Jones, C., Cassirer, A., Brock, A., Paganini, M., Irving, G., Vinyals, O., Osindero, S., Simonyan, K., Rae, J. W., Elsen, E., and Sifre, L. Improving language models by retrieving from trillions of tokens. *CoRR*, abs/2112.04426, 2021.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners, 2020.

Burda, Y., Grosse, R., and Salakhutdinov, R. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

Chen, Y., Garcia, E. K., Gupta, M. R., Rahimi, A., and Cazzanti, L. Similarity-based classification: Concepts and algorithms. *Journal of Machine Learning Research*, 10(3), 2009.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Cleveland, W. S. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836, 1979.

Cortes, C. and Vapnik, V. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

Cramer, P. Alphafold2 and the future of structural biology. *Nature Structural & Molecular Biology*, 28(9):704–705, 2021.

Das, R., Zaheer, M., Thai, D., Godbole, A., Perez, E., Lee, J. Y., Tan, L., Polymenakos, L., and McCallum, A. Case-based reasoning for natural language queries over knowledge bases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 9594–9611, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.755.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A. P., Caron, M., Geirhos, R., Alabdulmohsin, I., et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pp. 7480–7512. PMLR, 2023.

Donoho, D. L. and Liu, R. C. The" automatic" robustness of minimum distance functionals. *The Annals of Statistics*, 16(2):552–586, 1988.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Epasto, A., Mahdian, M., Mirrokni, V., and Zampetakis, E. Optimal approximation-smoothness tradeoffs for soft-max functions. *Advances in Neural Information Processing Systems*, 33:2651–2660, 2020.

Gandelsman, Y., Sun, Y., Chen, X., and Efros, A. Test-time training with masked autoencoders. *Advances in Neural Information Processing Systems*, 35:29374–29385, 2022.

Grathwohl, W., Swersky, K., Hashemi, M., Duvenaud, D., and Maddison, C. Oops i took a gradient: Scalable sampling for discrete distributions. In *International Conference on Machine Learning*, pp. 3831–3841. PMLR, 2021.

Gu, Y., Dong, L., Wei, F., and Huang, M. Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M.-W. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

henrikl (https://math.stackexchange.com/users/351007/henrikl). 1-smoothness of the symmetric softmax function. Mathematics Stack Exchange, 2021. URL https://math.stackexchange.com/q/4170855. URL:https://math.stackexchange.com/q/4170855 (version: 2021-06-12).

Huszár, F. How (not) to train your generative model: Scheduled sampling, likelihood, adversary? *arXiv preprint arXiv:1511.05101*, 2015.

Ibragimov, I. A. and Has Minskii, R. Z. *Statistical estimation: asymptotic theory*, volume 16. Springer Science & Business Media, 2013.

Iscen, A., Fathi, A., and Schmid, C. Improving image recognition by retrieving from web-scale image-text data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19295–19304, 2023.

Izacard, G. and Grave, E. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 874–880, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.74. URL https://aclanthology.org/2021.eacl-main.74.

Izacard, G., Lewis, P., Lomeli, M., Hosseini, L., Petroni, F., Schick, T., Dwivedi-Yu, J., Joulin, A., Riedel, S., and Grave, E. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*, 2022.

Jain, V. and Learned-Miller, E. Online domain adaptation of a pre-trained cascade of classifiers. In *CVPR 2011*, pp. 577–584. IEEE, 2011.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, Online, November 2020a. Association for Computational Linguistics.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020b.

Katkovnik, V. Y. and Kheisin, V. Dynamic stochastic approximation of polynomials drifts. *Avtomatika i Telemekhanika*, pp. 89–98, 1979.

Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., and Lewis, M. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*, 2020.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Leake, D. B., Kinley, A., and Wilson, D. C. Acquiring case adaptation knowledge: A hybrid approach. In *AAAI/IAAI, Vol. 1*, 1996. URL https://api.semanticscholar.org/CorpusID:11169287.

Lee, K., Chang, M.-W., and Toutanova, K. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6086–6096, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1612. URL https://aclanthology.org/P19-1612.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

Lewkowycz, A., Andreassen, A., Dohan, D., Dyer, E., Michalewski, H., Ramasesh, V., Slone, A., Anil, C., Schlag, I., Gutman-Solo, T., et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022.

Li, Y., Ildiz, M. E., Papailiopoulos, D., and Oymak, S. Transformers as algorithms: Generalization and stability in in-context learning. In Krause, A., Brunskill, E., Cho, K.,

Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19565–19594. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/li23l.html.

Liévin, V., Motzfeldt, A. G., Jensen, I. R., and Winther, O. Variational open-domain question answering. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 20950–20977. PMLR, 23–29 Jul 2023.

Lin, X. V., Chen, X., Chen, M., Shi, W., Lomeli, M., James, R., Rodriguez, P., Kahn, J., Szilvasy, G., Lewis, M., et al. Ra-dit: Retrieval-augmented dual instruction tuning. *arXiv preprint arXiv:2310.01352*, 2023.

Liska, A., Kocisky, T., Gribovskaya, E., Terzi, T., Sezener, E., Agrawal, D., De Masson D'Autume, C., Scholtes, T., Zaheer, M., Young, S., Gilsenan-Mcmahon, E., Austin, S., Blunsom, P., and Lazaridou, A. StreamingQA: A benchmark for adaptation to new knowledge over time in question answering models. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13604–13622. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/liska22a.html.

Long, A., Yin, W., Ajanthan, T., Nguyen, V., Purkait, P., Garg, R., Blair, A., Shen, C., and van den Hengel, A. Retrieval augmented classification for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6959–6969, 2022.

Lowe, D. G. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pp. 1150–1157. Ieee, 1999.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pp. 94–103. IEEE, 2007.

Meinhardt, T., Kirillov, A., Leal-Taixe, L., and Feichtenhofer, C. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8844–8854, 2022.

Ni, J., Qu, C., Lu, J., Dai, Z., Abrego, G. H., Ma, J., Zhao, V., Luan, Y., Hall, K., Chang, M.-W., et al. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9844–9855, 2022.

OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. URL https://api.semanticscholar.org/CorpusID:257532815.

Paranjape, A., Khattab, O., Potts, C., Zaharia, M., and Manning, C. D. Hindsight: Posterior-guided training of retrievers for improved open-ended generation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=Vr_BTpw3wz.

Pinsker, M. S. Optimal filtering of square-integrable signals in gaussian noise. *Problemy Peredachi Informatsii*, 16(2): 52–68, 1980.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Roweis, S. T. and Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500): 2323–2326, 2000.

Rubin, O., Herzig, J., and Berant, J. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2655–2671, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.191. URL https://aclanthology.org/2022.naacl-main.191.

Ruppert, D. and Wand, M. P. Multivariate locally weighted least squares regression. *The annals of statistics*, pp. 1346–1370, 1994.

Sachan, D. S., Reddy, S., Hamilton, W. L., Dyer, C., and Yogatama, D. End-to-end training of multi-document reader and retriever for open-domain question answering. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=5KWmB6JePx.

Sachan, D. S., Lewis, M., Yogatama, D., Zettlemoyer, L., Pineau, J., and Zaheer, M. Questions are all you need to train a dense passage retriever. *Transactions of the Association for Computational Linguistics*, 11:600–616, 2023.

Schonberger, J. L., Hardmeier, H., Sattler, T., and Pollefeys, M. Comparative evaluation of hand-crafted and learned local features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1482–1491, 2017.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shuster, K., Poff, S., Chen, M., Kiela, D., and Weston, J. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*, 2021.

Siegel, J. W. Optimal approximation rates for deep relu neural networks on sobolev and besov spaces. *Journal of Machine Learning Research*, 24(357):1–52, 2023.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., et al. Large language models encode clinical knowledge. *Nature*, pp. 1–9, 2023.

Sparck Jones, K. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

Stone, C. J. Consistent nonparametric regression. *The annals of statistics*, pp. 595–620, 1977.

Stone, C. J. Optimal rates of convergence for nonparametric estimators. *The annals of Statistics*, pp. 1348–1360, 1980.

Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., and Hardt, M. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Thai, D., Agarwal, D., Chaudhary, M., Das, R., Zaheer, M., Lee, J.-Y., Hajishirzi, H., and McCallum, A. Machine reading comprehension using case-based reasoning. *arXiv preprint arXiv:2305.14815*, 2023.

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

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Voorhees, E. M. et al. The trec-8 question answering track report. In *Trec*, volume 99, pp. 77–82, 1999.

Wang, S., Xu, Y., Fang, Y., Liu, Y., Sun, S., Xu, R., Zhu, C., and Zeng, M. Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3170–3179, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.226. URL https://aclanthology.org/2022.acl-long.226.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Yang, S., van de Weijer, J., Herranz, L., Jui, S., et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. *Advances in neural information processing systems*, 34:29393–29405, 2021.

Yarotsky, D. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017.

Zemlyanskiy, Y., de Jong, M., Ainslie, J., Pasupat, P., Shaw, P., Qiu, L., Sanghai, S., and Sha, F. Generate-and-retrieve: Use your predictions to improve retrieval for semantic parsing. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 4946–4951, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.438.

Zhang, H., Berg, A. C., Maire, M., and Malik, J. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 2126–2136. IEEE, 2006.

Zhang, T. *Mathematical analysis of machine learning algorithms*. Cambridge University Press, 2023.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., and Irving, G. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A. Preliminaries

Definition A.1 (Rademacher complexity)**.** Given a sample $\mathcal{S} = \{z_i = (x_i, y_i)\}_{i\in[n]} \subset \mathcal{Z}$ and a real-valued function class $\mathcal{F} : \mathcal{Z} \to \mathbb{R}$, the *empirical* Rademacher complexity of $\mathcal{F}$ with respect to $\mathcal{S}$ is defined as

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{F}) = \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^{n}\sigma_i f(z_i)\right], \tag{25}$$

where $\boldsymbol{\sigma} = \{\sigma_i\}_{i\in[n]}$ is a collection of n i.i.d. Bernoulli random variables. For $n \in \mathbb{N}$, the Rademacher complexity $\bar{\mathfrak{R}}_n(\mathcal{F})$ and *worst case* Rademacher complexity $\mathfrak{R}_n(\mathcal{F})$ are defined as follows.

$$\bar{\mathfrak{R}}_n(\mathcal{F}) = \mathbb{E}_{\mathcal{S}\sim\mathsf{D}^n}\left[\mathfrak{R}_{\mathcal{S}}(\mathcal{F})\right], \quad \text{and} \quad \mathfrak{R}_n(\mathcal{F}) = \sup_{\mathcal{S}\sim\mathcal{Z}^n}\mathfrak{R}_{\mathcal{S}}(\mathcal{F}). \tag{26}$$

Definition A.2 (Covering nsumber)**.** Let $\epsilon > 0$ and $\|\cdot\|$ be a norm defined over $\mathbb{R}^n$. Given a function class $\mathcal{F} : \mathcal{Z} \to \mathbb{R}$ and a collection of points $\mathcal{S} = \{z_i\}_{i\in[n]} \subset \mathcal{Z}$, we call a set of points $\{u_j\}_{j\in[m]} \subset \mathbb{R}^n$ an $(\epsilon, \|\cdot\|)$-cover of $\mathcal{F}$ with respect to $\mathcal{S}$, if we have

$$\sup_{f\in\mathcal{F}}\min_{j\in[m]}\|f(\mathcal{S}) - u_j\| \le \epsilon, \tag{27}$$

where $f(\mathcal{S}) = \big(f(z_1), \ldots, f(z_n)\big) \in \mathbb{R}^n$. The $\|\cdot\|$-covering number $\mathcal{N}(\mathcal{F}, \epsilon, \|\cdot\|)$ denotes the cardinally of the minimal $(\epsilon, \|\cdot\|)$-cover of $\mathcal{F}$ with respect to $\mathcal{S}$. In particular, if $\|\cdot\|$ is an normalized-ℓ_p norm ($\|v\| = (\frac{1}{dim(v)}\sum_{i=1}^{dim(v)}|v_i|^p)^{1/p}$), then we simply use $\mathcal{N}(\mathcal{F}, \epsilon, \mathcal{F}, \|\|_{L_p}, \mathcal{S})$ to denote the corresponding ℓ_p-covering number.

When $\mathcal{S}$ is unambiguous we may drop it, i.e. write $\mathcal{N}(\mathcal{F}, \epsilon, \mathcal{F}, \|\|_{L_p})$

Definition A.3 (Multi-layer perceptron (MLP))**.** We consider for both retrieval and predictor, the class of multi-layer-perceptron, aka fully connected Deep Neural Network, with Relu nonlinearity $\sigma(x) = \max(x, 0)$. An MLP is specified by the number of layers L, width W. We define an affine transform $A_{\mathbf{W},b}(\mathbb{R}^{d_1}, \mathbb{R}^{d_2}) = \mathbf{W}x + b$, with weight $\mathbf{W} \in \mathbb{R}^{d_2} \times \mathbb{R}^{d_1}$ and bias $b \in \mathbb{R}^{d_2}$. Let $\sigma \circ A_{\mathbf{W},b}(\mathbb{R}^{d_1}, \mathbb{R}^{d_2})$ define the elementwise application of Relu non-linearity on the affine transform. The class of L layers and W width MLP is defined as

$$\mathrm{MLP}(\mathbb{R}^d, \mathbb{R}^k; W, L) = \{A_{\mathbf{W}_L,b_L} \circ \sigma \circ A_{\mathbf{W}_{L-1},b_{L-1}} \circ \ldots \sigma \circ A_{\mathbf{W}_0,b_0}\}, \tag{28}$$

where $\mathbf{W}_L \in \mathbb{R}^{k\times W}$ and $b_L \in \mathbb{R}^k$, $\mathbf{W}_i \in \mathbb{R}^{W\times W}$ and $b_i \in \mathbb{R}^W$ for $1 \le i \le (L-1)$, and $\mathbf{W}_0 \in \mathbb{R}^{W\times d}$ and $b_0 \in \mathbb{R}^W$.

Definition A.4 (Sobolev space)**.** We denote the set of functions with finite L_p norm over Ω as $L_p(\Omega)$ i.e. for any $f \in L_p(\Omega)$, $\|f\|_{L_p(\Omega)} = \big(\int_{s\in\Omega} f(s)^p ds\big)^{1/p} < \infty$ for $p \ge 1$. Note for $p = \infty$ we have $\|f\|_{L_\infty(\Omega)} = \operatorname{ess\,sup}_{s\in\Omega}|f(s)|$. Let $\alpha \in \mathbb{N}^{d_{in}}$ denote a multi-index, and $|\alpha| = \sum_{i\in d_{in}}\alpha_i$ be it's degree. We denote by D^α the weak-derivative with respect to multi-index α for any function.

For an integer $\kappa > 0$, the Sobolev semi-norm $W^\kappa(L_p(\Omega))$ for a function f that has weak-derivatives of order κ is defined as

$$\forall 1 \ge p < \infty, |f|_{W^\kappa(L_p(\Omega))} = \Big(\sum_{\alpha:|\alpha|=\kappa}\|D^\alpha f\|^p_{L_p(\Omega)}\Big)^{1/p}, \quad |f|_{W^\kappa(L_\infty(\Omega))} = \max_{\alpha:|\alpha|=\kappa}\|D^\alpha f\|_{L_\infty(\Omega)}.$$

The Sobolev norm $W^\kappa(L_p(\Omega))$ for the same function f is defined as $\|f\|_{W^\kappa(L_p(\Omega))} = \|f\|_{L_p(\Omega)} + |f|_{W^\kappa(L_p(\Omega))}$. A function f with all weak-derivatives of order κ, and a finite $W^\kappa(L_p(\Omega))$ norm lies in the Sobolev space with κ derivatives and $L_p(\Omega)$ norm.

In our approximation guarantees for MLP retreiver and predictor classes later, we use Theorem 1 in (Siegel, 2023). We restate the theorem here for completeness.

Theorem A.5 (Restated Theorem 1 in (Siegel, 2023))**.** *For a function $f_0 : \Omega \to \mathbb{R}$ in the Sobolev space with κ derivatives and $L_q(\Omega)$, $p, q \in [1, \infty)$ and $\kappa \in (0, \infty)$ satisfying $(1/q - 1/p) \le s/d$, we have for some $C = c(\kappa, d) < \infty$, $\Omega = [-1, 1]^d$ and $W = 25d + 31$*

$$\inf_{f\in\mathrm{MLP}(\mathbb{R}^d,\mathbb{R};W,L)}\|f - f_0\|_{L_p(\Omega)} \le C\|f_0\|_{W^\kappa(L_q(\Omega))}L^{-\frac{2\kappa}{d}}.$$

Our VC Dimension bounds of MLP is based on the (Bartlett et al., 2019) which is in turn used for generalization bounds. We collect some necessary definitions and results from (Bartlett et al., 2019) and restate here for completeness.

Definition A.6 (VC dimension and growth of a binary function class)**.** For H, a class of functions from $\mathcal{A}$ to $0, 1$ the growth function of H evaluate on a input set of size m, is defined as

$$\Pi_H(m) = \max_{a_1,\ldots,a_m \in \mathcal{A}} |\{h(a_1), \ldots, h(a_m) : h \in H\}|.$$

The $VCdim(H)$ is defined as the largest m such that $\Pi_H(m) = m$, where if no such m is there we have $VCdim(H) = \infty$.

Definition A.7 (Pseudo dimension of real valued function class)**.** Let $\mathcal{F}$ be a class of functions from some space $\mathcal{A}$ to the real $\mathbb{R}$. The pseudo-dimension of class $\mathcal{F}$, denoted by $Pdim(\mathcal{F})$, is the largest m such that there exists $\{a_1, \ldots, a_m, r_1, \ldots, r_m\} \in \mathcal{A}^m \times \mathbb{R}^m$ such that for any binary sequence $\{b_1, \ldots, b_m\} \in \{0, 1\}^m$ there exists a function $f \in \mathcal{F}$ satisfying $\forall i : f(a_i) > r_i \iff b_i = 1$.

Note Pseudo-dimension is same as the VC dimension of the subgraph of class $\mathcal{F}$ which is used in (Zhang, 2023). We denote by $sgn(f)$ the sign of the function $f : \mathcal{A} \to \mathbb{R}$ from and $sgn(x) = \mathbb{1}(x \geq 0)$, and let $sgn(\mathcal{F}) = \{sgn(f) : f \in \mathcal{F}\}$. We define the VC dimension of the real valued function class $\mathcal{F}$ as $VCdim(\mathcal{F}) := VCdim(sgn(\mathcal{F}))$. It is mentioned in (Bartlett et al., 2019) that for neural network with a fixed architecture and fixed activation functions, namely class MLP, we have the $VCdim(sgn(\text{MLP})) = Pdim(\text{MLP})$.

We now adapt Theorem 6 in (Bartlett et al., 2019) to use it for the class $\text{MLP}(\mathbb{R}^d, \mathbb{R}; L, W)$ specialized for Relu non-linearity, i.e. in terminology of (Bartlett et al., 2019), number of breakpoint $pnt = 1$, and degree of polynomial is $deg = 1$.[1]

Theorem A.8 (Adapted Theorem 6 in (Bartlett et al., 2019))**.** *Consider the neural network class* $\text{MLP}(\mathbb{R}^d, \mathbb{R}; L, W)$ *that has Relu non-linearity. Let* $W_{tot,l}$ *denote the total number of parameters upto layer* $l \leq (L-1)$*, and* u_l *denote the number of units in layer* l*. Also define the parameters* $\bar{L} = \frac{1}{W_{tot,L}} \sum_{l=1}^{L} W_{tot,l} \leq L$*, and* $R = \sum_{l=1}^{L} l k_l \leq L^2 W$*. Then for the function class* $\mathcal{F}$ *of all real-valued functions computed by the MLP class and* m

$$\Pi_{sgn(\mathcal{F})}(m) \leq \prod_{l=1}^{L} 2 \left(\frac{2emk_l l}{W_{tot,l}} \right)^{W_{tot,l}} \leq (4emL)^{W_{tot,L}}.$$

Moreover, we have

$$VCdim(F) = L + \bar{L} W_{tot,L} \log_2(4e \sum_l l k_l \log_2(\sum_l 2elk_l)) = O(\bar{L} W_{tot,L} \log(L^2 W)).$$

We generalize the above result to capture the MLP with multi dimensional output as used by our predictor.

Theorem A.9 (Multi-ouput version of Theorem 6 in (Bartlett et al., 2019))**.** *Consider the neural network class* $\text{MLP}(\mathbb{R}^d, \mathbb{R}^k; L, W)$ *that has Relu non-linearity. Let* $W_{tot,l}$*,* u_l*,* $\bar{L}$*, and* R *be as defined in Theorem A.8. Then for the function class* $\mathcal{F}$ *of all real-valued functions computed by the MLP class with* k *dimensional output we have*

$$VCdim(F) = L + \bar{L} W_{tot,L} \log_2(4e \sum_l l k_l \log_2(\sum_l 2elk_l)) = O(\bar{L} W_{tot,L} \log(L^2 W)).$$

Proof. We proceed with our proof now for the class $\text{MLP}(\mathbb{R}^d, \mathbb{R}^K; L, W)$. $\mathcal{F}$ is the class of k dimensional output function computed by the MLP class. Let $a \in \mathbb{R}^{W_{tot,L}}$ parameterize one function $f \in \mathcal{F}$. We need to find the $VCdim$ of the set $\{sgn(f(x_i, j, a)) : a \in \mathbb{R}^{W_{tot,L}}, i \in [m], j \in [k]\}$. We partition the above set with respect to y and obtain the following inequality.

$$\begin{aligned} &|\{sgn(f(x_i, j, a)) : a \in \mathbb{R}^{W_{tot,L}}, i \in [m], j \in [k]\}| \\ &\leq \sum_{j \in [k]} |\{sgn(f(x_i, j, a)) : a \in \mathbb{R}^{W_{tot,L}}, i \in [m]\}| \quad \leq \sum_{j \in [k]} \Pi_{sgn(\text{MLP}(\mathbb{R}^d, \mathbb{R}; L, W))}(m) \leq k 2^L (2eRm/W_{tot,L})^{W_{tot,L}}. \end{aligned}$$

[1] Originally in (Bartlett et al., 2019) degree is denoted by d and break point by p, but we use deg and pnt, respectively, to avoid confusion.

For the second inequality we notice that for a fixed j the function $f(x_i, j, a)$ is computed by $\mathrm{MLP}(\mathbb{R}^d, \mathbb{R}; L, W)$ and bound it with the growth function $\Pi_{sgn(\mathrm{MLP}(\mathbb{R}^d,\mathbb{R};L,W))}$ over m points. Therefore, for the third inequality we can apply the specified bound for $\Pi_{sgn(\mathrm{MLP}(\mathbb{R}^d,\mathbb{R};L,W))}(m)$ inside the proof of Theorem 6 in (Bartlett et al., 2019). Here, we have specialized for Relu nonlinearlity, i.e. breaking point $pnt = 1$, and degree $deg = 1$. Applying Lemma 6 in (Bartlett et al., 2019) we obtain

$$VCdim(\mathrm{MLP}(\mathbb{R}^d, \mathbb{R}^k; L, W)) \leq L\log(k) + W_{tot,L}\log_2(4eR\log_2(4eR)) = O(L\log(k) + L^2W^2\log(LW)).$$

□

Finally, we state the following proposition that is a bounded version of the Gibb's inequality, that maximizes the cross entropy of a discrete probability function.

Proposition A.10 (Truncated Gibb's inequality)**.** *Let us consider two discrete distributions α, β over alphabet size K. Then for any constant $C > 0$, we have*

$$\sum_{i=1}^{K} \alpha_i \min(C, -\log(\beta_i)) \geq \sum_{i=1}^{K} \alpha_i \min(C, -\log(\alpha_i)) - (K-1)\exp(-C).$$

Proof. For two discrete distributions α, β over alphabet size K.

$$\begin{aligned}
&\sum_{i=1}^{K} \alpha_i \min(C, -\log(\beta_i)) \\
&= -\sum_{i=1}^{K} \alpha_i \log(\max(\exp(-C), \beta_i)) \\
&= -\sum_{i=1}^{K} \alpha_i \log(\alpha_i) + \sum_{i=1}^{K} \alpha_i \log\left(\alpha_i / \max(\exp(-C), \beta_i)\right) \\
&\geq -\sum_{i=1}^{K} \alpha_i \log(\alpha_i) + (\sum_{i=1}^{K} \alpha_i) \log\left(\sum_{i=1}^{K} \alpha_i / \sum_{i=1}^{K} \max(\exp(-C), \beta_i)\right)\right) \\
&\geq -\sum_{i=1}^{K} \alpha_i \log(\alpha_i) - \log(1 + (K-1)\exp(-C)) \\
&\geq -\sum_{i=1}^{K} \alpha_i \log(\alpha_i) - (K-1)\exp(-C) \\
&\geq \sum_{i=1}^{K} \alpha_i \min(C, -\log(\alpha_i)) - (K-1)\exp(-C)
\end{aligned}$$

The first inequality follows from the log-sum-inequality. The second inequality inequality uses the fact that $\sum_{i=1}^{K} \max(\exp(-C), \beta_i)$ is maximized by setting one $\beta_i = 1$ for some $1 \leq i \leq K$, while the rest are set to 0. The second last inequality follows by $\log(1 + x) \leq x$. The final inequality states taking a minimum with C can only decrease the value. □

B. Derivations of main result

As discussed in Section 2, the objective here is to study how the excess risk in Eq. (12). Our excess risk has three main components, generalization error, retriever approximation error, and predictor approximation error. In this section, we structure our results somewhat differently than the main body to capture the general setting of learning retriever with a fixed predictor, and vice versa. We first prove excess risk bounds for Learning the Retriever, then excess risk bounds for Learning the Predictor. Finally, we combine the results to obtain the final joint learning guarantees, which are presented in the paper. For the rest of the analysis. We need to specify the space of retrieved examples to define the complexity of the gap function. We recall that our retrieved samples are embedded in a compact subspace of $\mathbb{R}^{d_z}$, for simplicity say $\mathcal{Z} \subseteq [-1, 1]^{d_z}$. Similarly, we assume $\mathcal{X}$ to be a compact subspace of $\mathbb{R}^d$, for simplicity $\mathcal{X} \subseteq [-1, 1]^{d_x}$.

B.1. Learning the retriever

We first study learning the retriever over class Θ when the predictor ξ is fixed. The task of the retriever is to minimize

$$\mathbb{E}_{(X,Y)\sim\mathcal{D}}[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}\ell(h_\xi(X,Z),Y)] \quad = \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}\mathbb{E}_{Y|X}\ell(h_\xi(X,Z),Y)|X]\big] = \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}g_\xi(X,Z)\big],$$

where $g_\xi(X,Z) = \mathbb{E}_{Y|X}\ell(h_\xi(X,Z),Y)$. We have a closed form for the optimal retriever when not restricted within a function class. The optimal retriever is $p^{*,\xi}(z|x) = \mathbb{1}_{\arg\min_{z'\in\mathcal{I}} g_\xi(x,z')}(z)$, where a tie is broken arbitrarily.

For the fixed predictor ξ, the retriever that minimizes the empirical risk given, $\hat{\theta}(\xi)$, and the retriever that minimizes the population risk, $\theta(\xi)$, over the class Θ are defined as

$$\hat{\theta}(\xi) = \arg\min_{\theta\in\Theta}\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big),\ \theta(\xi) = \arg\min_{\theta\in\Theta}\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}g_\xi(X,Z)\big].$$

Here, the probability is defined using the softmax operator for a given $\theta\in\Theta$ as

$$p_{\theta,\mathcal{I}}\big(z|x\big) = \frac{\exp\big(r_\theta(x,z)\big)}{\sum_{z'\in\mathcal{I}}\exp\big(r_\theta(x,z')\big)},\quad \forall\, z\in\mathcal{I}, x\in\mathcal{X}.$$

Hardness of retrieval: We assume the $\mathrm{g}_\xi(x,z)$ function is in the Sobolev space with κ derivatives as defined in Section A. The following is the restatement of Assumption 3.1 but for any $\xi\in\Xi$ and not just the optimal one ξ^*.

Assumption B.1 (Complexity of g_ξ)**.** For any $\xi\in\Xi$, there exists a baseline $b_\xi : [-1,1]^{d_x}\to\mathbb{R}$ such that the function $\mathrm{gap}_\xi : [-1,1]^{d_x+d_z}\to\mathbb{R}$ with baseline b_ξ, as defined by $\mathrm{gap}_\xi = (g_\xi(x,z) - b_\xi(x))$ lies in the Sobolev space with κ derivatives and $L_\infty([-1,1]^{d_x+d_z})$ norm.

As noted earlier this means that the predictor loss has a possibly 'complex' component $b_\xi(x)$, and a relatively 'smooth' component $gap_\xi(x,z)$ that ensures two retrieved examples that are close gives similar loss for the predictor for any given sample $x\in\mathcal{X}$. As $gap_\xi(x,z)$ solely determines the optimal retrived set, it's smoothness defines the hardness of retrieval.

Excess risk decomposition: The excess error in retriever learning is given as

$$\begin{aligned}
&R_{\ell,\mathcal{I}}(\xi,\hat{\theta}(\xi)) - R_{\ell,\mathcal{I}}(f^\ell_{\mathrm{opt},\mathcal{I}})\\
&= \underbrace{\sum_{\theta=\theta(\xi),\hat{\theta}(\xi)}\Big|\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big) - \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}g_\xi(X,Z)\big]\Big|}_{\text{retriever generalization error}}\\
&+ \underbrace{R_{\ell,\mathcal{I}}(\xi,\theta(\xi)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big]}_{\text{retriever approximation error}} + \underbrace{\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big] - R_{\ell,\mathcal{I}}(f^\ell_{\mathrm{opt},\mathcal{I}})}_{\text{error from predictor }\xi}.
\end{aligned}$$

B.1.1. Generalization error:

We now proceed to bound the generalization error using the Radamacher complexity. With probability at least $(1-\delta)$ for any $\delta>0$,

$$\begin{aligned}
&\Big|\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_{\hat{\theta}(\xi)}(\cdot|X)}g_\xi(X,Z)\big] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_{\hat{\theta}(\xi)}(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\Big|\\
&\le 2\mathbb{E}_{\boldsymbol{\sigma}}\Big[\max_{\theta\in\Theta}\frac{1}{n}\sum_{i\in[n]}\sigma_i\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\Big] + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}\\
&\le 2\times\inf_{\varepsilon\in[0,c_\xi/2]}\big(4\varepsilon + \tfrac{12}{\sqrt{n}}\int_\varepsilon^{c_\xi/2}\sqrt{\log(\mathcal{N}(\Theta,\nu,\|\cdot\|_{2,[n],\xi}))}d\nu\big) + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}
\end{aligned}$$

Using covering number bound with chaining we obtain the final inequality, where

$$c_\xi = \sup_{\theta\in\Theta}\Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\big)^2\Big)^{1/2},$$

and $\mathcal{N}(\Theta,\nu,\|\cdot\|_{2,[n],\xi})$ denote the covering number of the retriever function Θ with error ν in L_2 norm w.r.t. the set $\{(x_i,y_i) : i\in[n]\}$ and ξ fixed,

$$\|\mathbf{u}\|_{2,[n],\xi} = \Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} u_{i,z}\ell\big(h_\xi(x_i,z),y_i\big)\big)^2\Big)^{1/2}, \forall\mathbf{u}\in\mathbb{R}^{n\times|\mathcal{I}|}.$$

The generalization error in retriever learning depends on the covering number of Θ (which we shall see is dependent on the embedding space of the retrieved examples).

As $\theta(\xi)$ is a fixed retriever, we do not need to take any union bound over the retriever space. Therefore, we have

$$\left|\mathbb{E}_X\left[\mathbb{E}_{Z\sim p_{\theta(\xi)}(\cdot|X)} g_\xi(X,Z)\right] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_{\theta(\xi)}(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\right| \le 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}.$$

B.1.2. Approximation error:

The approximation error of learning the retriever depends on the hardness of the function $\min_{z\in\mathcal{I}} g_\xi(X,z)$. We recall that this term is approximated using softmax over $r_\theta(X,Z)$.

We want to approximate the term $\min_{z\in\mathcal{I}} g_\xi(x,z)$ for all $x\in\mathcal{X}$, by $\sum_{z\in\mathcal{I}} p_{\theta,\mathcal{I}}(z|x) g_\xi(x,z)$. We can break down the approximation into two parts. First we show that the function $\mathrm{softmax}(-\tau\times g_\xi(x,z))$ approximates $\min_z g_\xi(x,z)$ for large τ. In particular, if $\tau = O(\log(|\mathcal{I}|)/\delta)$ then softmax approximates max with error δ (see, (McSherry & Talwar, 2007; Epasto et al., 2020)). Next, we show that $p_{\theta,\mathcal{I}}\big(z|x\big)$ can approximate $\mathrm{softmax}(-\tau\times g_\xi(x,z))$ well in L_2 norm.

We define

$$\tilde{p}_\xi(z|x) = \frac{\exp(-\tau g_\xi(x,z))}{\sum_{z'}\exp(-\tau g_\xi(x,z'))} = \frac{\exp(-\tau(g_\xi(x,z)-b_\xi(x)))}{\sum_{z'}\exp(-\tau(g_\xi(x,z')-b_\xi(x)))}.$$

Here recall that $b_\xi(x)$ is the baseline function in Assumption 3.1. An example of such baseline is $b_\xi(x)=\min_{\tilde{z}} g_\xi(x,\tilde{z})$ the loss under the optimal retrieved sample for each $x\in\mathcal{X}$.

We have for any $\theta\in\Theta$

$$\begin{aligned}
&R_{\ell,\mathcal{I}}(\xi,\theta(\xi)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big]\\
&\le R_{\ell,\mathcal{I}}(\xi,\theta) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big]\\
&= \mathbb{E}_X\big[\sum_{z\in\mathcal{I}}(p_{\theta,\mathcal{I}}(z|x)-\tilde{p}_\xi(z|x))g_\xi(x,z)\big] + \mathbb{E}_X\big[\sum_{z\in\mathcal{I}}\tilde{p}_\xi(z|x) - \min_{z\in\mathcal{I}} g_\xi(x,z)\big]\\
&\le \mathbb{E}_X\big[\|g_\xi(x,\cdot)\|_\infty \|p_{\theta,\mathcal{I}}(\cdot|x)-\tilde{p}_\xi(\cdot|x)\|_1\big] + \frac{\log(|\mathcal{I}|)}{\tau^2}\\
&\le \mathbb{E}_X\big[\|g_\xi(x,\cdot)\|_\infty \|r_\theta(x,\cdot)+\tau\mathrm{gap}_\xi(x,\cdot)\|_\infty\big] + \frac{\log(|\mathcal{I}|)}{\tau^2}\\
&\le \ell_{\max}\|r_\theta+\tau\mathrm{gap}_\xi\|_\infty + \frac{\log(|\mathcal{I}|)}{\tau^2}
\end{aligned}$$

In the first inequality, the first term in the final inequality is simply using norm bounds for inner product while the second term in the final inequality follows from Theorem 3.1 in (Epasto et al., 2020) which originates from (McSherry & Talwar, 2007). The second inequality uses the fact that softmax functions over K classes follow $\|sm(x)-sm(y)\|_\infty \le \|x-y\|_1$ (see (henrikl , https://math.stackexchange.com/users/351007/henrikl)). In the final inequality, we bound the results using the L_2 norm bound of the inner product, and use $\ell_{\max}$ to bound the norm of g_ξ.

As the above bound hold for any $\tau>0$, by optimizing of τ and θ we obtain,

$$R_{\ell,\mathcal{I}}(\xi,\theta(\xi)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big] \le \inf_{\theta\in\Theta}\inf_{\tau>0} \ell_{\max}\|r_\theta+\tau\mathrm{gap}_\xi\|_\infty + \frac{\log(|\mathcal{I}|)}{\tau^2}. \tag{29}$$

Note the above bound hold for any $\theta\in\Theta$. Therefore, if there exists a $\theta\in\Theta$ such that the function $r_\theta(x,z)$ approximates the function $-\tau\mathrm{gap}_\xi(x,z)$ well, then we end up with small approximation error. For that purpose, we need to impose some smoothness condition on the gap function, $\mathrm{gap}_\xi(x,z)$ for $(x,z)\in\mathcal{X}\times\mathcal{I}$, to provide approximability results using MLP.

B.1.3. Instantiation of MLP retriever

We consider Θ to be the class of MLP defined in Equation (28).

Generalization error for MLP retriever: To bound the generalization error, we need to first bound the covering number of $\mathcal{N}(\Theta,\nu,\|\cdot\|_{2,[n],\xi})$, for $\Theta=\mathrm{MLP}(\mathbb{R}^{d_x+d_z},\mathbb{R};W,L)$. Here, $\mathcal{X}\subseteq\mathbb{R}^{d_x}$ and $\mathcal{I}\subseteq\mathbb{R}^{d_z}$ (i.e. the retrieved space is embeded in $\mathbb{R}^{d_z}$). We first want to bound the covering number $\mathcal{N}(\Theta,\nu,\|\cdot\|_{2,[n],\xi})$ with a covering number of $\mathrm{MLP}(\mathbb{R}^{d_x+d_z},\mathbb{R};W,L)$.

We have for a fixed data set $\mathcal{S}_n := \{(x_1, y_1), \ldots, (x_n, y_n)\}$, predictor ξ, and for two $\theta, \theta' \in \Theta$

$$\begin{aligned}
&\Big(\frac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}}(p_\theta(z|x_i) - p_{\theta'}(z|x_i))\ell\big(h_\xi(x_i,z),y_i\big)\big)^2\Big)^{1/2}\\
&\le \ell_{\max}\Big(\frac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}}|p_\theta(z|x_i) - p_{\theta'}(z|x_i)|\big)^2\Big)^{1/2}\\
&\le \ell_{\max}\Big(\frac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}}|p_\theta(z|x_i) - p_{\theta'}(z|x_i)|\big)^2\Big)^{1/2}\\
&\le \ell_{\max}\Big(\frac{1}{n}\sum_{i\in[n]}\big(\max_{z\in\mathcal{J}}|r_\theta(x_i,z) - r_{\theta'}(x_i,z)|\big)^2\Big)^{1/2}\\
&\le \ell_{\max}\sup_{x\in\mathcal{S}_n, z\in\mathcal{J}}|r_\theta(x,z) - r_{\theta'}(x,z)|
\end{aligned}$$

Now consider a $\|\cdot\|_{\infty,n|\mathcal{J}|}$ norm cover of Θ, Θ_{cov} with cardinality $\mathcal{N}(\Theta, \nu/\ell_{\max}, \|\cdot\|_{\infty,n|\mathcal{J}|})$. Here, $\|\cdot\|_{\infty,n|\mathcal{J}|}$ is defined as $\|u\|_{\infty,n|\mathcal{J}|} = \sup_{x_i\in\mathcal{S}_n}\sup_{z\in\mathcal{J}}|u_{i,z}|$, $\forall \mathbf{u} \in \mathbb{R}^{n\times|\mathcal{J}|}$.
For any $\theta \in \Theta$, there exists a $\theta'_\theta \in \Theta_{cov}$ such that $\sup_{x\in\mathcal{S}_n, z\in\mathcal{J}}|r_\theta(x,z) - r_{\theta'_\theta}(x,z)| \le \nu/\ell_{\max}$. This means, that Θ_{cov} forms a ν-cover in the $\|\cdot\|_{2,[n],\xi}$ norm. In other words, we have $\mathcal{N}(\Theta, \nu, \|\cdot\|_{2,[n],\xi}) \le \mathcal{N}(\Theta, \nu/\ell_{\max}, \|\cdot\|_{\infty,n|\mathcal{J}|})$.
In absence of norm bounds for the MLP weights and biases, direct covering number bound is not readily available. Therefore, we will use pseudo-dimension of the class Θ from (Bartlett et al., 2019) to bound the covering number $\mathcal{N}(\Theta, \nu, \|\cdot\|_{\infty,n|\mathcal{J}|})$ using (Zhang, 2023). In particular, if the pseudo-dimension of Θ is d_{VC}, then we have $\log\mathcal{N}(\Theta, \nu, \|\cdot\|_{\infty,n|\mathcal{J}|}) \le 1 + \log(1 + d_{VC}) + d_{VC}\log(\max\{2, en|\mathcal{J}|/d_{VC}\nu\})$ as per Theorem 5.11 in (Zhang, 2023). From Theorem 6 in (Bartlett et al., 2019) we know that for the class $\mathrm{MLP}(\mathbb{R}^d, \mathbb{R}; W, L)$ the pseudo-dimension is $O(LNlog(M))$, where N is the number of parameters, and M is the number of computation units. For fully connected network, we have $N = O(LW^2)$, and $M = O(LW)$. So the final generalization error is $O\big(\frac{\ell_{\max}LW\sqrt{\log(LW)\log(n|\mathcal{J}|)}}{\sqrt{n}}\big)$ for large enough L (we will set L as a function of the data size n which will satisfy this). This is obtained by setting $\varepsilon = c/\sqrt{n}$ for a constant c, and $\delta = 1/n$.

Approximation error for MLP retriever: Let $\Omega = [-1, 1]^{d_x+d_z}$. Our excess risk bounds closely follow the work of (Siegel, 2023) which generalizes (Yarotsky, 2017).[2]
We consider Θ to be the class of multi-layer-perceptron (MLP), a.k.a. fully connected Deep Neural Network with Relu nonlinearity as defined in Section A. From Theorem 1 in (Siegel, 2023) by taking $p = q = \infty$ in the theorem statement, under Assumption B.1 we get that

$$\inf_{f\in\mathrm{MLP}(\mathbb{R}^{d_x+d_z},\mathbb{R};W,L)}\|f - \mathrm{gap}_\xi\|_{L_\infty(\Omega)} \le C\|\mathrm{gap}_\xi\|_{W^\kappa(L_\infty(\Omega))}L^{-2\kappa/(d_x+d_z)}$$

for $\Omega \in [-1, 1]^{d_x+d_z}$, $W = 25(d_x + d_z) + 31$ and $C = c(\kappa, d_x + d_z) < \infty$ (independent of L).
Therefore, we have for $\Theta = \mathrm{MLP}(\mathbb{R}^{d_x+d_z}, \mathbb{R}; 25(d_x + d_z) + 31, L)$ under Assumption B.1 we have

$$R_{\ell,\mathcal{J}}(\xi, \theta(\xi)) - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_\xi(X, z)\big] \le C\tau\|g_\xi\|_{L_\infty(\Omega)}\|\mathrm{gap}_\xi\|_{W^\kappa(L_\infty(\Omega))}L^{-2\kappa/(d_x+d_z)} + \frac{\log(|\mathcal{J}|)}{\tau^2}.$$

This follows from the following series of inequalities:

$$\begin{aligned}
&R_{\ell,\mathcal{J}}(\xi, \theta(\xi)) - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_\xi(X, z)\big]\\
&\le \mathbb{E}_X\big[\|g_\xi(x,\cdot)\|_\infty\big]\mathbb{E}_X\big[\|r_\theta(x,\cdot) + \tau\mathrm{gap}_\xi(x,\cdot)\|_\infty\big] + \frac{\log(|\mathcal{J}|)}{\tau^2}\\
&= \tau\mathbb{E}_X\big[\|g_\xi(x,\cdot)\|_\infty\big]\|\tilde{r}_\theta - \mathrm{gap}_\xi\|_{L_\infty(\Omega)} + \frac{\log(|\mathcal{J}|)}{\tau^2}\\
&\le C\tau\mathbb{E}_X\big[\|g_\xi(x,\cdot)\|_\infty\big]\|\mathrm{gap}_\xi\|_{W^\kappa(L_\infty(\Omega))}L^{-2\kappa/(d_x+d_z)} + \frac{\log(|\mathcal{J}|)}{\tau^2}\\
&\le C'\ell_{\max}\tau L^{-2\kappa/(d_x+d_z)} + \frac{\log(|\mathcal{J}|)}{\tau^2}
\end{aligned}$$

[2]We note (Siegel, 2023) works with $\Omega = [0, 1]^d$, and as mentioned therein, it can extend to bounded domain, e.g. $[a, b]^d$ which includes our setting. Furthermore, one can extend to non-integer Sobolev and Besov spaces following (Siegel, 2023).

The first inequality is what we derived earlier. The second equality, replaces $\tilde{r}_\theta = -\tau r_\theta$. The second last inequality follows by optimizing $\tilde{r}_\theta$ over the class Θ, as we see then $-\tau r_\theta$ also lies in Θ, and applying Theorem 1 in (Siegel, 2023). The final inequality combines $C' = C\|\text{gap}_\xi\|_{W^\kappa(L_\infty(\Omega))}$ and bounds $\mathbb{E}_X\left[\|g_\xi(x,\cdot)\|_\infty\right] \leq \ell_{\max}$.

As the choice of τ is not algorithmic, we can optimize for τ. In particular, we choose $\tau = cL^{-2\kappa/3(d_x+d_z)}\log^{1/3}(|\mathcal{I}|)$ to obtain the approximation error bound as $O(\ell_{\max}L^{-4\kappa/3(d_x+d_z)}\log^{1/3}(|\mathcal{I}|))$, where we treat the remaining terms that are independent of τ and L as constants.

Excess risk for MLP retriever learning: Adding approximation and generalization error we bound the excess risk as

$$\text{Excess Risk} \leq \underbrace{\mathbb{E}_X\left[\min_{z\in\mathcal{I}} g_\xi(X,z)\right] - R_{\ell,\mathcal{I}}(f^\ell_{\text{opt},\mathcal{I}})}_{\text{error from predictor } \xi} + \underbrace{O(\ell_{\max}L^{-\frac{4\kappa}{3(d_x+d_z)}}\log^{1/3}(|\mathcal{I}|))}_{\text{retriever approximation error}} + \underbrace{O\big(\frac{\ell_{\max}LW\sqrt{\log(LW)\log(n|\mathcal{I}|)}}{\sqrt{n}}\big)}_{\text{retriever generalization error}} \tag{30}$$

$$= \underbrace{\mathbb{E}_X\left[\min_{z\in\mathcal{I}} g_\xi(X,z)\right] - R_{\ell,\mathcal{I}}(f^\ell_{\text{opt},\mathcal{I}})}_{\text{error from predictor } \xi} + \underbrace{\tilde{O}(\ell_{\max}n^{-\frac{2\kappa}{3(d_x+d_z)+4\kappa}})}_{\text{retriever combined error}} \tag{31}$$

Here, we choose $L = n^{\frac{3(d_x+d_z)}{6(d_x+d_z)+8\kappa}}$ and use the data-store size $|\mathcal{I}| = poly(n)$. Note we have $W = O(d_x + d_z)$ which is combined with the constants.

B.2. Learning the predictor

We now quantify the excess risk of a predictor ξ for a fixed predictor θ. For a fixed predictor θ, the task of the predictor is to minimize

$$\mathbb{E}_{(X,Y)\sim\mathcal{D}_{XY}}[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}\ell(h_\xi(X,Z),Y)] = \mathbb{E}_{((X,Z),Y)\sim\mathcal{D}_{XY}\times p_\theta(\cdot|X)}\big[\ell(h_\xi(X,Z),Y)|X]\big]$$

The predictor now learns from the joint distribution $\mathcal{D}_{XY}\times p_\theta(\cdot|X)$. We assume that the *hardness* of the classification task performed by the predictor varies with the selected retriever θ.

Similar to retriever learning, for a fixed retriever θ, the predictor that minimizes the empirical risk given, $\hat{\xi}(\theta)$, and the predictor that minimizes the population risk, $\xi^*(\theta)$, over the class Ξ are defined as

$$\hat{\xi}(\theta) = \arg\min_{\xi\in\Xi}\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big),\ \xi^*(\theta) = \arg\min_{\xi\in\Xi}\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)}g_\xi(X,Z)\big],$$

where $g_\xi(X,Z) = \mathbb{E}_{Y|X}\ell(h_\xi(X,Z),Y)$. We also define the predictor over the class Ξ with 'optimal' retrieval (possibly outside of Θ) that minimizes the population risk as ξ^* as $\xi^* = \arg\min_{\xi\in\Xi}\mathbb{E}_X\left[\min_{z\in\mathcal{I}} g_\xi(X,z)\right]$.

Usefulness of data-store: We start with characterization of the prediction task in presence of the data-store $\mathcal{I}$. We consider that there exists a score function $h_* : \mathcal{X}\times\mathcal{Z}\to\mathbb{R}^{|\mathcal{Y}|}$, and corresponding probability distribution

$$p^y_*(x,z) = \frac{\exp(h^y_*(x,z))}{\sum_{y'}\exp(h^{y'}_*(x,z))}, \tag{32}$$

that approximates well $p^y_{\mathsf{D}_{XY}}(x) := \mathbb{P}_{Y\sim\mathsf{D}_{XY}}(y|X=x)$ for all $x\in\mathcal{X}$ and $y\in\mathcal{Y}$. Furthermore, this score function h_* lies coordinate wise in the Sobolev space. The Assumption 3.2 captures the above. We restate the assumption here for convenience.

Assumption B.2 (Retrieval quality)**.** There exists a score function $h_* : \mathcal{X}\times\mathcal{Z}\to\mathbb{R}^{|\mathcal{Y}|}$ such that

1. for each $y\in\mathcal{Y}$, the function h^y_* lies in the Sobolev space with $\kappa_\mathcal{I}$ derivatives and finite $L_\infty([-1,1]^{d_x+d_z})$ norm,
2. for any $x\in\mathcal{X}$ there exists a retrieved example $z^*(x)\in\mathcal{I}$ such that for $p^y_*(x,z)$ as defined in Equation (32)
$$\max_{y\in\mathcal{Y}}\sup_{x\in\mathcal{X}}|p^y_*(x,z(x)) - p^y_{\mathsf{D}_{XY}}(x)| \leq c_\mathcal{I}|\mathcal{I}|^{-\gamma_\mathcal{I}}.$$

The tuple $(\gamma_\mathcal{I}, d_z, \kappa_\mathcal{I})$ defines the usefulness of the data-store $\mathcal{I}$. In particular, the higher the $\gamma_\mathcal{I}$ the closer the approximation, and the higher the $\kappa_\mathcal{I}$ and smaller the embedding dimension d_z the 'easier' the score function used for this approximation.

Excess risk decomposition The excess risk decomposition is given as below.

$$R_{\ell,\mathcal{I}}(\hat{\xi}(\theta),\theta) - R_{\ell,\mathcal{I}}(f^\ell_{\text{opt},\mathcal{I}})$$

$$
\begin{aligned}
&\leq \sum_{\xi=\xi^*(\theta),\hat{\xi}(\theta)} \Big|\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big) - \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_\xi(X,Z)\big]\Big| \\
&+ R_{\ell,\mathcal{J}}(\xi^*(\theta),\theta) - R_{\ell,\mathcal{J}}(f^\ell_{\mathrm{opt},\mathcal{J}}) \\
&\leq \sum_{\xi=\xi^*(\theta),\hat{\xi}(\theta)} \Big|\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big) - \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_\xi(X,Z)\big]\Big| \\
&+ \underbrace{R_{\ell,\mathcal{J}}(\xi^*(\theta),\theta) - R_{\ell,\mathcal{J}}(\xi^*,\theta)}_{\leq 0} + R_{\ell,\mathcal{J}}(\xi^*,\theta) - R_{\ell,\mathcal{J}}(f^\ell_{\mathrm{opt},\mathcal{J}}) \\
&\leq \underbrace{\sum_{\xi=\xi^*(\theta),\hat{\xi}(\theta)} \Big|\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big) - \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_\xi(X,Z)\big]\Big|}_{\text{generalization error}} \\
&+ \underbrace{R_{\ell,\mathcal{J}}(\xi^*,\theta) - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi^*}(X,z)\big]}_{\text{retriever error}} + \underbrace{\mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi^*}(X,z)\big] - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{f^\ell_{\mathrm{opt},\mathcal{J}}}(X,z)\big]}_{\text{predictor error}}
\end{aligned}
$$

Note that in the second inequality as the retriever is fixed (and not optimized with predictor), we can substitute the predictor $\xi^*(\theta)$ with ξ^* to obtain an upper bound.

B.2.1. APPROXIMATION ERROR:

We specialize our analysis for the log-loss bounded by $\ell_{\max} > 0$ give as

$$
\ell(h_\xi(x,z),y) = \min(\ell_{\max}, -\log(p_\xi(y|x,z))) = \min(\ell_{\max}, \log(\sum_{y'\in\mathcal{Y}} \exp(h^{y'}_\xi(x,z))) - h^y_\xi(x,z)). \tag{33}
$$

We now need to bound the predictor error $(\mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi^*}(X,z)\big] - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{f^\ell_{\mathrm{opt},\mathcal{J}}}(X,z)\big])$ for the bounded log-loss. We want to relate this term to the $p^y_*(x,z)$ for which we have good control over its complexity. We first need a lower bound for $\mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{f^\ell_{\mathrm{opt},\mathcal{J}}}(X,z)\big]$ as a function of $p^y_*(x,z)$. We proceed as follows:

$$
\begin{aligned}
&\mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{f^\ell_{\mathrm{opt},\mathcal{J}}}(X,z)\big] \\
&\geq \mathbb{E}_X\big[\sum_{y\in\mathcal{Y}} p^y_{\mathsf{D}_{XY}}(X)\min(\ell_{\max}, -\ln(p^y_{\mathsf{D}_{XY}}(X)))\big] - (|\mathcal{Y}|-1)\exp(-\ell_{\max}) \\
&\geq \mathbb{E}_X\big[\sum_{y\in\mathcal{Y}} p^y_{\mathsf{D}_{XY}}(X)\min(\ell_{\max}, -\ln(p^y_*(X,z^*(X)))\big] - (|\mathcal{Y}|-1)\exp(-\ell_{\max}) \\
&\quad - \mathbb{E}_X\big[\max_{y\in\mathcal{Y}} \ell_{\max}|p^y_*(X,z^*(X)) - p^y_{\mathsf{D}_{XY}}(X)|\big] \\
&\geq \mathbb{E}_X\big[\sum_{y\in\mathcal{Y}} p^y_{\mathsf{D}_{XY}}(X)\min(\ell_{\max}, -\ln(p^y_*(X,z^*(X)))\big] - (|\mathcal{Y}|-1)\exp(-\ell_{\max}) - c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}} \\
&= \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] - (|\mathcal{Y}|-1)\exp(-\ell_{\max}) - c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}
\end{aligned}
$$

In the first inequality, applying Proposition A.10 in Appendix A to our setting with $C = \ell_{\max}$ and $K = |\mathcal{Y}|$ we obtain the lower bound. The second inequality relies on the mean-value bound,

$$
|\min(C,-\log(x)) - \min(C,-\log(y))| \leq \max_{x'\in[x,y]} \big|\tfrac{\delta}{\delta x}\min(C,-\log(x))|_{x=x'}(x-y)\big| \leq C|x-y|.
$$

Next inequality is obtained by Assumption B.2 with $z^*(x)$ is ad defined therein. The final inequality substitutes $g_{h^*}(x,z^*(x)) = \mathbb{E}_{Y|X=x}[\ell(h_*(x,z^*(x)),y)]$ where $h_*(x,z)$ is the score function used in Equation (32).

We now derive an upper bound for the predictor error part of our excess risk bound. Let $\xi \in \Xi$ be an arbitrary predictor

$$
\begin{aligned}
\text{Predictor Error} &\triangleq \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi^*}(X,z)\big] - \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{f^\ell_{\mathrm{opt},\mathcal{J}}}(X,z)\big] \\
&\leq \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi^*}(X,z)\big] - \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}} \\
&\leq \mathbb{E}_X\big[\min_{z\in\mathcal{J}} g_{\xi}(X,z)\big] - \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}} \\
&\leq \mathbb{E}_X\big[g_{\xi}(X,z^*(X))\big] - \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}
\end{aligned}
$$

The second inequality follows by substituting the lower bound of $\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{f^{\ell}_{\text{opt},\mathcal{I}}}(X,z)\big]$. As ξ^* optimizes ℓ-risk over Ξ, we can substitute with the arbitrary predictor ξ to obtain an upper bound. The final inequality is obtained by substituting $z^*(X)$ instead of minimizing with respect to $z \in \mathcal{I}$. Note that the final inequality holds for all $\xi \in \Xi$ as the initial choice of ξ was arbitrary.

Bounding the term $\mathbb{E}_X\big[g_\xi(X,z^*(X))\big] - \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big]$, is similar to bounding the ℓ-risk for classification with the data distribution $\mathbb{P}(X=x,Z=z,Y=y) = \mathbb{P}_{\mathsf{D}_{XY}}(X=x,Y=y)\mathbb{1}(z=z^*(X))$. Our strategy is to bound ℓ-risk with L_∞ distance between the score functions $h^y_{\xi^*}(x,z)$ and the score function $h^y_*(x,z)$ which lies in the Sobolev space as given in the Assumption B.2. In particular, we have the following L_∞ norm bound.

$$\begin{aligned}
&\mathbb{E}_X\big[g_\xi(X,z^*(X))\big] - \mathbb{E}_X\big[g_{h_*}(X,z^*(X))\big] \\
&= \mathbb{E}_{XY}\big[\ell(h^Y_\xi(X,z^*(X))) - \ell(h^Y_*(X,z^*(X)))\big] \\
&\le \mathbb{E}_{XY}\big[|h^Y_\xi(X,z^*(X)) - h^Y_*(X,z^*(X))| + \max_{y\in\mathcal{Y}} |h^y_\xi(X,z^*(X)) - h^y_*(X,z^*(X))|\big] \\
&\le 2\mathbb{E}_X\big[\max_{y\in\mathcal{Y}} |h^y_\xi(X,z^*(X)) - h^y_*(X,z^*(X))|\big]
\end{aligned}$$

The first inequality follows by substituting the bounded log-loss, and using the fact that for any two $s, s' \in \mathbb{R}^K$, $|\log(\sum_k exp(s_k)) - \log(\sum_k exp(s'_k))| \le \max_k |s_k - s'_k|$.

We note that the above holds for all ξ. This gives the general approximation error bound as

$$\text{Predictor Error} \le \inf_{\xi\in\Xi} 2\mathbb{E}_X\big[\max_{y\in\mathcal{Y}} |h^y_\xi(X,z^*(X)) - h^y_*(X,z^*(X))|\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{I}}|\mathcal{I}|^{-\gamma_{\mathcal{I}}}. \tag{34}$$

B.2.2. Generalization error:

The generalization error can be bounded in a similar manner as the retriever learning. The key difference here is that the predictor is learnt over the space Ξ while the retriever is fixed.

$$\begin{aligned}
&|\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_{\hat{\xi}(\theta)}(X,Z)\big] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_{\hat{\xi}(\theta)}(x_i,z),y_i\big)| \\
&\le 2\mathbb{E}_{\boldsymbol{\sigma}}\Big[\max_{\xi\in\Xi}\frac{1}{n}\sum_{i\in[n]}\sigma_i\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\Big] + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}} \\
&\le 2\times\inf_{\varepsilon\in[0,c_\theta/2]}\big(4\varepsilon + \tfrac{12}{\sqrt{n}}\int_\varepsilon^{c_\theta/2}\sqrt{\log(\mathcal{N}(\Xi,\nu,\|\cdot\|_{2,[n],\theta}))}d\nu\big) + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}
\end{aligned}$$

The final inequality again follows using covering number based bounds with chaining. We have used for a fixed retriever θ

$$c_\theta = \sup_{\xi\in\Xi}\Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big)\big)^2\Big)^{1/2},$$

and $\mathcal{N}(\Xi,\nu,\|\cdot\|_{2,[n],\theta})$ denote the covering number of the retriever function Ξ with error ν in L_2 norm w.r.t. the set $\{(x_i,y_i): i\in[n]\}$ and θ fixed,

$$\|\mathbf{u}\|_{2,[n],\theta} := \Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{I}} p_\theta(z|x_i)u_{i,z}\big)^2\Big)^{1/2},\ \forall \mathbf{u}\in\mathbb{R}^{n\times|\mathcal{I}|}.$$

As $\xi^*(\theta)$ is fixed for a fixed θ we can directly bound, without any union over the learner/predictor space,

$$|\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_{\xi^*(\theta)}(X,Z)\big] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_{\xi^*(\theta)}(x_i,z),y_i\big)| \le 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}.$$

Note the predictor approximation error is independent of retriever learning as it is compared with respect to the Bayes optimal retriever (i.e. $min_{z\in\mathcal{I}} g_\xi(x,z)$).

B.2.3. Instantiation of MLP predictor

As a concrete example, we now consider the space $\Xi = \text{MLP}(\mathbb{R}^{d_x+d_z}, \mathbb{R}^{\mathcal{Y}}; W, L)$.

Approximation error of MLP predictor: Our approximation results rely mainly on the results in (Siegel, 2023). The key difference here is the output is now $|\mathcal{Y}|$ dimensional. We find MLP of depth L and width at most $W' = O(d_x+d_z)$, to individually approximate the functions $h^y_*(x,z)$ for each $y\in\mathcal{Y}$. Later we can join these networks in parallel to obtain a final network with depth L and width at most $O((d_x+d_z)|\mathcal{Y}|)$. In principle these networks may share sub-networks

(e.g. the bit extraction networks, the sub-domain indexation network for $p = q$ in (Siegel, 2023)) used for constructing the approximation. However, this is out of scope for this work, and we leave this open.

From Theorem 1 in (Siegel, 2023) by taking $p = q = \infty$ in the theorem statement, under Assumption B.2 we get that for each $y \in \mathcal{Y}$ there exists a MLP $f_y \in \text{MLP}(\mathbb{R}^{d_x+d_z}, \mathbb{R}; W, L)$ such that

$$\|f_y - h^y_*\|_{L_\infty(\Omega)} \leq C_y \|h^y_*\|_{W^\kappa(L_\infty(\Omega))} L^{-2\kappa_{\mathcal{J}}/(d_x+d_z)}$$

for $\Omega \in [-1,1]^{d_x+d_z}$, $W = 25(d_x+d_z)+31$ and $C_y = c(\kappa_{\mathcal{J}}, d_x+d_z) < \infty$ (independent of L). By concatenating the networks f_y for $y \in \mathcal{Y}$ in parallel (c.f. Lemma 5 in (Siegel, 2023)), and using the first layer to share the (d_x+d_z) input to these parallel networks we obtain a MLP $f_{\text{opt}} \in \text{MLP}(\mathbb{R}^{d_x+d_z}, \mathbb{R}^K; W_{\mathcal{Y}}, L+1)$, $W_{\mathcal{Y}} = O(|\mathcal{Y}|(d_x+d_z))$, such that we have

$$\|f^y_{\text{opt}} - h^y_*\|_{L_\infty(\Omega)} \leq \big(\max_{y\in\mathcal{Y}} C_y \|h^y_*\|_{W^\kappa(L_\infty(\Omega))}\big) L^{-2\kappa_{\mathcal{J}}/(d_x+d_z)}.$$

By using $\xi = f^y_{\text{opt}}$ in our bounds we obtain the predictor error as

$$\text{Predictor Error} \leq 2\big(\max_{y\in\mathcal{Y}} C_y \|h^y_*\|_{W^\kappa(L_\infty(\Omega))}\big) L^{-2\kappa_{\mathcal{J}}/(d_x+d_z)} + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}} \tag{35}$$

Generalization error for MLP predictor: We now bound the generalization error for Ξ which is the class of multi-layer perceptron (MLP) with Relu nonlinearity given as $\text{MLP}(\mathbb{R}^{(d_x+d_z)}, \mathbb{R}^{|\mathcal{Y}|}; W, L)$.

The first step is to bound the covering number $\mathcal{N}(\Xi, \nu, \|\cdot\|_{2,[n],\theta})$ norm with the covering number $\mathcal{N}(\Xi, \nu, \|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$. Where $\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|}$ is defined as $\|u\|_{\infty,n|\mathcal{J}||\mathcal{Y}|} = \sup_{x_i\in\mathcal{S}_n} \sup_{z\in\mathcal{J}} \sup_{y\in\mathcal{Y}} |u_{i,z,y}|$, $\forall \mathbf{u} \in \mathbb{R}^{n\times|\mathcal{J}|\times|\mathcal{Y}|}$.

We have for a fixed data set $\mathcal{S}_n := \{(x_1,y_1),\ldots,(x_n,y_n)\}$ and retriever ξ, and two predictors $\xi, \xi' \in \Xi$

$$\begin{aligned}
&\Big(\tfrac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}} p_\theta(z|x_i)(\ell\big(h_\xi(x_i,z),y_i\big) - \ell\big(h_{\xi'}(x_i,z),y_i\big))\big)^2\Big)^{1/2} \\
&\leq \Big(\tfrac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\big(\ell\big(h_\xi(x_i,z),y_i\big) - \ell\big(h_{\xi'}(x_i,z),y_i\big)\big)^2\Big)^{1/2} \\
&\leq \Big(\tfrac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\big(|h^{y_i}_\xi(x_i,z) - h^{y_i}_{\xi'}(x_i,z)| + \max_{y\in\mathcal{Y}}|h^y_\xi(x_i,z) - h^y_{\xi'}(x_i,z)|\big)^2\Big)^{1/2} \\
&\leq \Big(\tfrac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_\theta(z|x_i)\big(|h^{y_i}_\xi(x_i,z) - h^{y_i}_{\xi'}(x_i,z)| + \max_{y\in\mathcal{Y}}|h^y_\xi(x_i,z) - h^y_{\xi'}(x_i,z)|\big)^2\Big)^{1/2} \\
&\leq \sqrt{2}\sup_{x\in\mathcal{X}}\sup_{y\in\mathcal{Y}}\sup_{z\in\mathcal{J}}|h^y_\xi(x,z) - h^y_{\xi'}(x,z)|
\end{aligned}$$

The first inequality follows from Cauchy-Schwartz. For the case of bounded log-loss, we obtain the second inequality using the fact that for any two $s, s' \in \mathbb{R}^K$, $|\log(\sum_k exp(s_k)) - \log(\sum_k exp(s'_k))| \leq \max_k |s_k - s'_k|$.

Let Ξ_{cov} be a $\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|}$ norm cover for the space Ξ of cardinality $\mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$. That implies, for any $\xi \in \Xi$ there exists a $\xi'(\xi) \in \Xi_{cov}$ such that $\sup_{x\in\mathcal{X}}\sup_{y\in\mathcal{Y}}\sup_{z\in\mathcal{J}}|h^y_\xi(x,z) - h^y_{\xi'}(x,z)| \leq \nu$. Therefore, due to the above inequality, we have $\Big(\frac{1}{n}\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}} p_\theta(z|x_i)(\ell\big(h_\xi(x_i,z),y_i\big) - \ell\big(h_{\xi'}(x_i,z),y_i\big))\big)^2\Big)^{1/2} \leq \nu$. So Ξ_{cov} forms a cover of Ξ with respect to the $\|\cdot\|_{2,[n],\theta}$ norm. Hence, $\mathcal{N}(\Xi,\nu,\|\cdot\|_{2,[n],\theta}) \leq \mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$.

We need to bound $\mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$ next. Similar to the retrieval analysis, we first apply (Zhang, 2023) to bound the covering number $\mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$ with pseudo-dimension. However, we need slight reformulation of the function $h_\xi : \mathcal{X}\times\mathcal{Z} \to \mathbb{R}^{|\mathcal{Y}|}$ to apply the results therein. Let us define function $\tilde{h}_\xi : \mathcal{X}\times\mathcal{Z}\times\mathcal{Y} \to \mathbb{R}$, where for each $y \in \mathcal{Y}$ we have $\tilde{h}_\xi(x,y,z) = h^y_\xi(x,z)$. It is easy to see that $\mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|})$ covering of set Ξ remains unchanged due to this reformulation. In particular, if the pseudo-dimension of $\{\tilde{h}_\xi : \xi \in \Xi\}$ is $\tilde{d}_{VC}$, then we have $\log\mathcal{N}(\Xi,\nu,\|\cdot\|_{\infty,n|\mathcal{J}||\mathcal{Y}|}) \leq 1 + \log(1+\tilde{d}_{VC}) + \tilde{d}_{VC}\log(\max\{2, en|\mathcal{J}||\mathcal{Y}|/\tilde{d}_{VC}\nu\})$ as per Theorem 5.11 in (Zhang, 2023).

Next we derive the pseudo-dimension of the class $\{\tilde{h}_\xi : \xi \in \Xi\}$ using (Bartlett et al., 2019). One challenge here is that for the MLP we are considering the label y does not lie in the input space, rather this correspond to one co-ordinate of the $|\mathcal{Y}|$-dimensional output. This can be captured with the slight modification of Theorem 6 in (Bartlett et al., 2019), namely Theorem A.9 in Appendix A. By Theorem A.9 we have for $\Xi = \text{MLP}(\mathbb{R}^{d_x+d_z}, \mathbb{R}^{|\mathcal{Y}|}; L, W)$ the VC dimension of Ξ as $VCdim(\Xi) = O(L\log(|\mathcal{Y}|) + L^2W^2\log(LW))$. The final generalization bound obtained is

$O\big(\frac{\ell_{\max}\sqrt{(L\log(|\mathcal{Y}|)+L^2W^2\log(LW))\log(n|\mathcal{I}||\mathcal{Y}|)}}{\sqrt{n}}\big)$.

Excess risk of predictor learning: We can now combine the generalization and approximation errors to obtain the final excess risk. The final excess risk is upper bounded as

$$\text{Excess Risk} \le \underbrace{R_{\ell,\mathcal{I}}(\xi^*,\theta) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X,z)\big]}_{\text{error from retriever } \theta} + \underbrace{O(L^{-2\kappa_{\mathcal{I}}/(d_x+d_z)} + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{I}}|\mathcal{I}|^{-\gamma_{\mathcal{I}}})}_{\text{predictor approximation error}}$$
$$+ \underbrace{O\big(\frac{\ell_{\max}\sqrt{(L\log(|\mathcal{Y}|)+L^2W^2\log(LW))\log(n|\mathcal{I}||\mathcal{Y}|)}}{\sqrt{n}}\big)}_{\text{predictor generalization error}} \tag{36}$$
$$= \underbrace{\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi}(X,z)\big] - R_{\ell,\mathcal{I}}(f^{\ell}_{\text{opt},\mathcal{I}})}_{\text{error from predictor } \xi} + \underbrace{\tilde{O}(|\mathcal{Y}|^{\frac{2\kappa_{\mathcal{I}}}{(d_x+d_z)+2\kappa_{\mathcal{I}}}} n^{-\frac{\kappa_{\mathcal{I}}}{(d_x+d_z)+2\kappa_{\mathcal{I}}}})}_{\text{predictor combined error}} \tag{37}$$

We have retriever set grow polynomially with data, $|\mathcal{I}| = \Omega(n^s)$, and we let $\ell_{\max} = \log(|\mathcal{Y}|) + s'\log(n)$. For $s \ge \frac{\kappa_{\mathcal{I}}}{((d_x+d_z)+2\kappa_{\mathcal{I}})\gamma_{\mathcal{I}}}$, and $s' \ge \frac{\kappa_{\mathcal{I}}}{((d_x+d_z)+2\kappa_{\mathcal{I}})}$ the final error bound for predictor follows by setting $L = n^{\frac{(d_x+d_z)}{2(d_x+d_z)+4\kappa_{\mathcal{I}}}}|\mathcal{Y}|^{-\frac{d_x+d_z}{(d_x+d_z)+2\kappa_{\mathcal{I}}}}$. Note that the choice of L and W here are related to predictor size, and are independent of the choices in retriever size.

Moreover, here we see Assumption B.2 forces the quality of retriever set to become the bottleneck in predictor excess risk, if we have $|\mathcal{I}| = o(n^s)$ for $s = \frac{\kappa_{\mathcal{I}}}{((d_x+d_z)+2\kappa_{\mathcal{I}})\gamma_{\mathcal{I}}}$.

B.3. Joint learning of retriever and predictor

In this section, we consider the task of joint learning the predictor and retriever from the space Ξ and Θ, respectively. The empirical optimizer pair $(\hat{\xi}_{\text{joint}}, \hat{\theta}_{\text{joint}})$, and the population optimizer $(\xi^*_{\text{joint}}, \theta^*_{\text{joint}})$ for the joint task are given as.

$$\hat{\xi}_{\text{joint}}, \hat{\theta}_{\text{joint}} = \operatorname*{arg\,min}_{\xi\in\Xi,\hat{\theta}\in\Theta} \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_\theta(z|x_i)\ell\big(h_\xi(x_i,z),y_i\big),\ \xi^*_{\text{joint}}, \theta^*_{\text{joint}} = \operatorname*{arg\,min}_{\xi\in\Xi} \mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_\xi(X,Z)\big].$$

Recall, the optimal predictor with best possible retrieval is $\xi^* = \arg\min_{\xi\in\Xi}\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_\xi(X,z)\big]$. We let the optimal retriever for ξ^* as $\theta(\xi^*) = \arg\min_{\theta\in\Theta}\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_\theta(\cdot|X)} g_{\xi^*}(X,Z)\big]$.

The excess risk for the classes Θ and Ξ can be bounded as

$$\begin{aligned}
&R_{\ell,\mathcal{I}}(\hat{\xi}_{\text{joint}},\hat{\theta}_{\text{joint}}) - R_{\ell,\mathcal{I}}(f^{\ell}_{\text{opt},\mathcal{I}}) \\
&\le \sum_{(\theta,\xi)\in\{(\hat{\theta}_{\text{joint}},\hat{\xi}_{\text{joint}}),(\theta^*_{\text{joint}},\xi^*_{\text{joint}})\}} |R_{\ell,\mathcal{I}}(\xi,\theta) - R_{\ell,\mathcal{I},n}(\xi,\theta)| + R_{\ell,\mathcal{I}}(\xi^*_{\text{joint}},\theta^*_{\text{joint}}) - R_{\ell,\mathcal{I}}(f^{\ell}_{\text{opt},\mathcal{I}}) \\
&\le \sum_{(\theta,\xi)\in\{(\hat{\theta}_{\text{joint}},\hat{\xi}_{\text{joint}}),(\theta^*_{\text{joint}},\xi^*_{\text{joint}})\}} |R_{\ell,\mathcal{I}}(\xi,\theta) - R_{\ell,\mathcal{I},n}(\xi,\theta)| + R_{\ell,\mathcal{I}}(\xi^*,\theta(\xi^*)) - R_{\ell,\mathcal{I}}(f^{\ell}_{\text{opt},\mathcal{I}}) \\
&\le \underbrace{\sum_{(\theta,\xi)\in\{(\hat{\theta}_{\text{joint}},\hat{\xi}_{\text{joint}}),(\theta^*_{\text{joint}},\xi^*_{\text{joint}})\}} |R_{\ell,\mathcal{I}}(\xi,\theta) - R_{\ell,\mathcal{I},n}(\xi,\theta)|}_{\text{Generalization Error}} \\
&+ \underbrace{R_{\ell,\mathcal{I}}(\xi^*,\theta(\xi^*)) - \mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X,z)\big]}_{\text{retriever error}} + \underbrace{\mathbb{E}_X\big[\min_{z\in\mathcal{I}} g_{\xi^*}(X,z)\big] - R_{\ell,\mathcal{I}}(f^{\ell}_{\text{opt},\mathcal{I}})}_{\text{predictor error}}
\end{aligned}$$

Here, we substitute the pair $(\xi^*,\theta(\xi^*))$ for $(\xi^*_{\text{joint}},\theta^*_{\text{joint}})$ where the former may have higher loss than latter, but the predictor error is easily controlled. Also, note that the retriever $\theta(\xi^*)$ is optimized for predictor ξ^*. Therefore, we can bound the retriever error properly unlike the fixed predictor case. We next bound the generalization and approximation errors separately.

B.3.1. Generalization Error:

First, for the fixed (θ^*,ξ^*) pair we bound the generalization error as

$$|\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_{\theta^*}(\cdot|X)} g_{\xi^*}(X,Z)\big] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{I}} p_{\theta^*}(z|x_i)\ell\big(h_{\xi^*}(x_i,z),y_i\big)| \le 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}}.$$

Next, the generalization for the $(\hat{\xi}, \hat{\theta})$ error can be bounded as.

$$
\begin{aligned}
&|\mathbb{E}_X\big[\mathbb{E}_{Z\sim p_{\hat{\theta}}(\cdot|X)} g_{\hat{\xi}}(X,Z)\big] - \frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{J}} p_{\hat{\theta}}(z|x_i)\ell\big(h_{\hat{\xi}}(x_i,z),y_i\big)| \\
&\le 2\mathbb{E}_{\boldsymbol{\sigma}}\Big[\max_{(\theta,\xi)\in\Theta\times\Xi}\frac{1}{n}\sum_{i\in[n]}\sigma_i\sum_{z\in\mathcal{J}} p_{\theta}(z|x_i)\ell\big(h_{\xi}(x_i,z),y_i\big)\Big] + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}} \\
&\le 2\times \inf_{\varepsilon\in[0,c_{\max}/2]}\big(4\varepsilon + \tfrac{12}{\sqrt{n}}\int_{\varepsilon}^{c_{\max}/2}\sqrt{\log(\mathcal{N}(\Theta\times\Xi,\nu,\|\cdot\|_{2,[n]}))}d\nu\big) + 3\ell_{\max}\sqrt{\tfrac{\log(2/\delta)}{n}},
\end{aligned}
$$

The second inequality again follows using covering number based bounds with chaining. We have used for a fixed retriever θ

$$
c_{\max} = \sup_{\theta,\xi\in\Theta\times\Xi}\Big(\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}} p_{\theta}(z|x_i)\ell\big(h_{\xi}(x_i,z),y_i\big)\big)^2\Big)^{1/2},
$$

and $\mathcal{N}(\Xi,\nu,\|\cdot\|_{2,[n]})$ denote the covering number of the retriever function Ξ with error ν in L_2 norm w.r.t. the set $\{(x_i,y_i): i\in[n]\}$,

$$
\|\mathbf{u}\|_{2,[n]} := \Big(\sum_{i\in[n]}\big(\sum_{z\in\mathcal{J}} u_{i,z}\big)^2\Big)^{1/2},\ \forall \mathbf{u}\in\mathbb{R}^{n\times|\mathcal{J}|}.
$$

The term can be bounded using the retriever and predictor learning complexities as

$$
\sqrt{\log(\mathcal{N}(\Theta\times\Xi,\nu,\|\cdot\|_{2,[n]}))} \le \max_{\xi\in\Xi}\sqrt{\log(\mathcal{N}(\Theta,\nu/2,\|\cdot\|_{2,[n],\xi}))} + \max_{\theta\in\Theta}\sqrt{\log(\mathcal{N}(\Xi,\nu/2,\|\cdot\|_{2,[n],\theta}))}.
$$

This implies that the generalization error of joint learning is (orderwise) bounded by the sum of the generalization error of retriever and predictor learning.

B.3.2. Approximation error

Moreover, the approximation error of predictor and retriever decouples under our decomposition, and under Assumption B.1 and B.2. So the approximation error is also bounded by the sum of the approximation error of retriever with optimal predictor, and the approximation error of predictor learning. Our derived bounds approximation error of the retriever holds uniformly for all predictor, so it also holds for optimal predictor. This implies that the joint retriever and predictor learning error is bounded (orderwise) by the sum of the predictor and retriever errors derived earlier in Equation (29), and Equation (34) earlier.

Proof of Theorem 3.3. We define $f_{\mathcal{N}}(\nu;\mathcal{A},\mathcal{B}) = \sup_{b\in\mathcal{B}}\sqrt{\log(\mathcal{N}(\mathcal{A},\nu,\|\cdot\|_{2,n,b}))}$. Putting the approximation and generalization errors together we obtain the final excess risk bound as

$$
\begin{aligned}
&\Delta_{\ell,\xi}(\hat{\xi},\hat{\theta}) \\
&\le 3\ell_{\max}(\tfrac{1}{n}+\sqrt{\tfrac{\log(n)}{n}}) + \inf_{\varepsilon\in[0,\frac{\ell_{\max}}{2}]} 8\varepsilon + \tfrac{24}{\sqrt{n}}\int_{\varepsilon}^{\frac{\ell_{\max}}{2}} f_{\mathcal{N}}(\tfrac{\nu}{2};\Theta,\Xi) + f_{\mathcal{N}}(\tfrac{\nu}{2};\Xi,\Theta)d\nu \\
&+ \inf_{\theta\in\Theta}\inf_{\tau>0}\ell_{\max}\|r_{\theta}+\tau\mathrm{gap}_{\xi}\|_{\infty} + \frac{\log(|\mathcal{J}|)}{\tau^2} \\
&+ \inf_{\xi\in\Xi} 2\mathbb{E}_X\big[\max_{y\in\mathcal{Y}}|h^y_{\xi}(X,z^*(X)) - h^y_*(X,z^*(X))|\big] + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}.
\end{aligned}
$$

This completes the proof. □

B.3.3. Instantiation of MLP retriever and predictor

For the scenario where the retriever and predictor are MLP, we can reuse the earlier analysis to provide the excess risk bound here.

Proof of Theorem 3.4. Let us recall from Appendix B.1.3, in Equation 30 a retriever MLP with depth L_{ret}, and width is $W_{\mathrm{ret}} = O(d_x+d_z)$ gives an approximation error $O\left(\ell_{\max}L_{\mathrm{ret}}^{-\frac{4\kappa}{3(d_x+d_z)}}\log^{1/3}(|\mathcal{J}|)\right)$ and the generalization error $O\left(\frac{\ell_{\max}LW\sqrt{\log(LW)\log(n|\mathcal{J}|)}}{\sqrt{n}}\right)$

Similarly, from Appendix B.2.3, in Equation (35), a MLP predictor with depth L_{pred} and width $W_{\text{pred}} = O(|\mathcal{Y}|(d_x + d_z))$ has an approximation error $O\left(L_{\text{pred}}^{-2\kappa_{\mathcal{J}}/(d_x+d_z)} + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}\right)$, and a generalization error $O\left(\frac{\ell_{\max}\sqrt{(L_{\text{pred}}\log(|\mathcal{Y}|)+L_{\text{pred}}|\mathcal{Y}|\log(L_{\text{pred}}|\mathcal{Y}|))\log(n|\mathcal{J}||\mathcal{Y}|)}}{\sqrt{n}}\right)$

The combined error in this case is given as

$$\Delta_{\ell,\mathcal{J}}(\hat{\xi},\hat{\theta}) \le \tilde{O}\left(\frac{\ell_{\max}}{\sqrt{n}}\left(L_{\text{ret}} + L_{\text{pred}}|\mathcal{Y}|\right)\right) + O\left(\ell_{\max} L_{\text{ret}}^{-\frac{4\kappa}{3(d_x+d_z)}}\log^{1/3}(|\mathcal{J}|)\right) + O\left(L_{\text{pred}}^{-\frac{2\kappa_{\mathcal{J}}}{(d_x+d_z)}} + (|\mathcal{Y}|-1)\exp(-\ell_{\max}) + c_{\mathcal{J}}|\mathcal{J}|^{-\gamma_{\mathcal{J}}}\right).$$

This completes the proof. □

Finally, combining excess risk in Equation (31) and (37), the joint learning excess error rate is given as

$$\text{Joint Excess Risk MLP} \le \begin{cases} \tilde{O}\left(n^{-\frac{2\kappa}{3(d_x+d_z)+4\kappa}} + |\mathcal{Y}|^{\frac{2\kappa_{\mathcal{J}}}{(d_x+d_z)+2\kappa_{\mathcal{J}}}} n^{-\frac{\kappa_{\mathcal{J}}}{(d_x+d_z)+2\kappa_{\mathcal{J}}}}\right), & \text{if } |\mathcal{J}| = \Omega(n^{\frac{\kappa_{\mathcal{J}}}{((d_x+d_z)+2\kappa_{\mathcal{J}})\gamma_{\mathcal{J}}}}), \\ \tilde{O}\left(n^{-\frac{2\kappa}{3(d_x+d_z)+4\kappa}} + |\mathcal{J}|^{-\gamma_{\mathcal{J}}}\right), & \text{otherwise.} \end{cases} \tag{38}$$

Here κ is defined in Assumption B.1, and $(\kappa_{\mathcal{J}}, \gamma_{\mathcal{J}})$ are defined in Assumption B.2. Also, d_x is the embedding dimension of input $x \in \mathcal{X}$ and d_z is the embedding dimension of retrieved example $z \in \mathcal{J}$.

C. More experiments

Method	small			base			large		
	small	base	large	small	base	large	small	base	large
EMDR2	40.0	47.7	52.0	41.5	48.0	51.4	41.6	48.8	52.6
PDist	49.7	57.4	61.3	48.6	57.0	61.0	47.7	55.7	58.9
Cross-Entropy + PG	44.9	52.6	54.7	45.3	53.3	55.2	44.9	51.7	54.9
Cross-Entropy + TopK	48.9	56.8	60.9	47.9	55.5	59.6	46.7	54.3	58.2

Table 4. **Recall on NQ**. We measure the recall of answer string being present in the retrieved passage performance of RAMs across various training objectives and model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

C.1. Implementation details

Computing the objective (13), let alone its gradient, requires evaluating the reader and predictor over the entire data-store $\mathcal{J}$ making it prohibitively expensive. We explore two ways to approximately compute the objective:

Top-K approximation This approach involves constraining the summation to a specific subset. Periodically we compute $p_\theta(z|x)$ for all items $z \in \mathcal{J}$ based on the current value of θ. We use this to obtain a set of K documents $\mathcal{Z}(x_i)$ with the highest (stale) scores, i.e. $\mathcal{T}_K(p_\theta(\cdot|x_i))$ and evaluate the sum on this.

$$\mathcal{L}_{\mathcal{J},n}^{\text{CE+TopK}}(\theta;\xi,\mathcal{J}) = -\frac{1}{n}\sum_{i\in[n]}\sum_{z\in\mathcal{Z}(x_i)} p_{\theta,\mathcal{J}}(z|x_i)\cdot\log p_\xi(y_i|x_i,z) \tag{39}$$

This methodology is akin to those adopted by EMDR2 and PDist, with the set being refreshed every 500 training steps and the selection of $K = 64$.

Policy gradient Based on connection to RLHF/RLAIF, we propose to use policy gradient method (Sutton & Barto, 2018) to obtain an unbaised estimate of gradient with respect to θ efficiently. However, as policy gradients suffer from high variance (Burda et al., 2015; Grathwohl et al., 2021) we use a constant baseline (Williams, 1992) for variance reduction, i.e.

Method	small			base			large		
	small	base	large	small	base	large	small	base	large
EMDR2	46.6	54.7	62.4	46.1	55.7	61.6	46.0	53.9	59.5
PDist	59.6	68.6	72.8	59.1	61.9	72.2	56.4	59.3	69.3
Cross-Entropy + PG	58.1	60.7	70.7	56.9	66.1	64.2	54.2	61.4	61.3
Cross-Entropy + TopK	57.1	64.5	69.1	55.9	63.5	68.1	54.2	61.2	65.8

Table 5. **Recall on TriviaQA**. We measure the recall of answer string being present in the retrieved passage performance of RAMs across various training objectives and model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

small			base			large		
small	base	large	small	base	large	small	base	large
96.4M	170.9M	396.4M	258.8M	333.3M	558.9M	773.6M	848.1M	1073.7M

Table 6. **Parameters**. We report the model parameters in various configuration by RAMs across various model sizes. Top row specifies the predictor size and the second row specifies the retriever size.

our objective becomes

$$\begin{aligned}\mathcal{L}^{\text{CE+PG}}_{\mathcal{I},n}(\theta;\xi,\mathcal{I}) &= -\frac{1}{n}\sum_{i\in[n]}\sum_{j\in[K]} p_{\theta,\mathcal{I}}(z_j(x_i)|x_i)\cdot\big[\log p_\xi(y_i|x_i,z_j(x_i))-b\big]\\ \nabla_\theta\mathcal{L}^{\text{CE+PG}}_{\mathcal{I},n}(\theta;\xi,\mathcal{I}) &= -\frac{1}{n}\sum_{i\in[n]}\sum_{j\in[K]} \nabla_\theta\log p_{\theta,\mathcal{I}}(z_j(x_i)|x_i)\cdot\big[\log p_\xi(y_i|x_i,z_j(x_i))-b\big],\end{aligned} \tag{40}$$

where $z_j(x_i)\sim p_\theta(\cdot|x_i)$ are K i.i.d. samples from the retriever distribution. We use $K=64$ and $b=5$.

C.2. Training details

Dataset The versions of the open-domain QA datasets, we use are:

- TriviaQA: https://www.tensorflow.org/datasets/catalog/trivia_qa#trivia_qaunfilterednocontext
- NQOpen https://www.tensorflow.org/datasets/catalog/natural_questions_open

Optimization. For all of our experiments, we use ADAM weight decay optimizer with a short warm up period (2000 steps) and a linear decay schedule. We use the peak learning rate of 1×10^{-4}. The weight decay factor is 0.1. We chose batch sizes to be 64. The number of total training steps is as follows:

- No retriever, train predictor ξ: 40,000
- Fixed retriever θ_0, train predictor ξ: 20,000
- Fixed predictor $\xi^\star(\theta_0)$, train retriever θ: 20,000
- Jointly train predictor ξ and retriever θ: 40,000

Initializations We initialize models for different configurations as follows:

- No retriever, train predictor ξ: We initialize the predictor from public pretrained T5 checkpoint.
- Fixed retriever θ_0, train predictor ξ: We initialize the fixed retriever from public pretrained GTR checkpoint and predictor from public pretrained T5 checkpoint.
- Fixed predictor $\xi^\star(\theta_0)$, train retriever θ: We initialize the fixed predictor from the final checkpoint of previous run, i.e. "Fixed retriever θ_0, train predictor ξ". The retriever is initialized from public pretrained GTR checkpoint.
- Jointly train predictor ξ and retriever θ: We initialize the fixed retriever from public pretrained GTR checkpoint and predictor from public pretrained T5 checkpoint.