We do agree this assumption might restrict the applicability of our results. Nevertheless, even this setting is not consistent with the sample size.

Advantages of adopting a functional approximation perspective

We would like to thank Reviewer 2 for raising this question, and we made improvements in a revised version if we get a chance.

Response to the significance of $c_1$. Sorry for the confusion. We will emphasize the definition of $\epsilon(f^\ast, \ell)$, and the notation $\lambda_{m_{\ell}}$ and $\lambda_{m_{\ell}+1}$ in a revised version if possible. Here $\lambda_{m_{\ell}}$ and $\lambda_{m_{\ell}+1}$ (introduced in the paragraph above Theorem 4) are the $\ell$-th and $\ell+1$-th largest distinct eigenvalues of the integral operator, and $(\lambda_{m_{\ell}} - \lambda_{m_{\ell}+1})$ is the eigengap. Once the distribution $\rho$ is fixed, the eigenvalues of the integral operator is also fixed – they do not change with the sample size $n$. Thus, for fixed $\rho$ and $f^\ast$, $c_1 = \Theta(\sqrt{\log(1/\delta)/n})$.

Response to the concern on fixed second layer. We would like to thank Reviewer 3 for raising this question, and we made improvements in a revised version if possible. Here $\ell$ is fixed, the eigenvalues of the integral operator is also fixed – they do not change with the sample size $n$. Thus, for fixed $\rho$ and $f^\ast$, $c_1 = \Theta(\sqrt{\log(1/\delta)/n})$. Each hidden unit $j$ has a corresponding weight $a_j$, and we introduce a pairing hidden unit $j'$.

Besides, only the first three terms in the upper bound of $\|y_0^\ast - (I - nK)^s y\|$ in Lemma 5 remain.

Response to other issues. Due to space limit we collectively respond to other issues here. We fixed grammars and typos mentioned by Reviewer 2, and carefully went through the entire article to fix others in a revised version; we also clarified notations at their first appearances as pointed out by Reviewer 3.

Summary. We would like to thank the entire review team for their efforts and insightful comments. In particular, we would like to thank Reviewer 1 for the positive comments, and Reviewers 2 and 3 for sharing their concerns on the significance of our results compared with existing literature. We sincerely apologize for the lack of clarity of our original submission in distinguishing our results from existing work. We will take more care of communicating the improvements in a revised version if we get a chance.

Overall, the advantages of taking such a functional approximation perspective are at least three-fold: It can help us spot the fact that some existing convergence guarantees are diminishing in the sample size $n$.

Below, we first detail the significance of these three advantages, and then provide clarifications on the specific issues raised by the reviewers.

Advantages of adopting a functional approximation perspective

We showed in Theorem 2 that the existing rate characterizations in the influential line of work [ADH+19, DLL+18, DZPS18] (where DZPS18 refers to arXiv:1810.02054) approach zero (i.e., $\rightarrow 0$) as the sample size $n \rightarrow \infty$. In fact, even the rate derived in the state-of-the-art work on training over-parametrized neural networks (NNs) [OS19] approaches zero as $n \rightarrow \infty$; see Corollary 2.2. in [OS19]. However, in many applications the volumes of the datasets are huge – the ImageNet dataset has 14 million images. For those applications, a non-diminishing convergence rate is more desirable.

Recall that $f^\ast$ denotes the underlying function that generates output labels/responses (i.e., $y$)’s given input features (i.e., $x$’s). For example, $f^\ast$ could be a constant function or a linear function, i.e., $f^\ast(x) \equiv c$ or $f^\ast(x) = \theta^\top x$. Clearly, the difficulty in learning $f^\ast$ via training neural networks should crucially depend on the properties of $f^\ast$ itself. Our Theorem 4 and Corollary 2 essentially say that the training convergence rate is determined by how $f^\ast$ can be decomposed into the eigenspaces of some integral operator. Our results are also validated by a couple of existing empirical observations: (1) The spectrum of the MNIST data concentrates on the first a few eigenspaces; and (2) the training is slowed down if labels are partially corrupted [Zhang et al. 2016] (arXiv:1611.03530). One important practical implication of our results is: in order to speed up training, the practitioners could “adapt” the eigenspaces of the underlying integral operator of GD by designing better feature engineering method so that the underlying true function could be well projected onto a few eigenspaces.

It has been empirically observed that linear over-parameterization $m = \Theta(n)$ is sufficient for GD to converge [ZBHB16]. However, the state-of-the-art theoretical results on network over-parameterization is $m = \Theta(n^2)$ but at a price of having diminishing convergence rate (Corollary 2.2. in [OS19]). In our work, we show (in Corollary 2) that if $f^\ast$ can be decomposed into a finite number of eigenspaces of the integral operator, then $m = \Theta(n^2)$ is sufficient and a constant convergence rate can be achieved. Moreover, we conjecture (not mentioned in our original submission) that with a slightly different network initialization, the over-parameterization level might be improved to $m = \Theta(n \log(1/\delta))$. In particular, for each hidden unit $j$ (where $j = 1, \cdots, m$), we introduce a pairing hidden unit $j'$. We initialize $w_j$ and $a_j$ as there were in our original submission, and set $w_j' = w_j^0$ and $a_j' = -a_j^0$ for each $j$.

By Eq. (4), we know $\hat{y}_i(0) = 0$ for $i = 1, \cdots, m$; thus, we do not need to set $v$ to be small in order to control $\|\hat{y}_i(0)\|$. In summary, we can bound the second kernel function by zero mapping. Then we can follow the line of analysis in the current paper to conclude.