We thank all the reviewers for their thorough feedback and valuable suggestions! We will revise our paper accordingly.

To Reviewer 1:

Thanks for the positive review!

To Reviewer 2:

"Effectiveness of proposed algorithm in training neural networks":
The goal of our paper is not to propose a new algorithm that outperforms current ones in training neural networks, but rather to analyze gradient descent on tensor decompositions beyond the lazy training regime. Tensor decomposition problems are closely related to the training of neural networks, e.g., the population loss of one-hidden-layer networks is a sum of tensor decompositions (Ge et al., 2017), but our algorithm cannot be directly applied to neural network training. For the tensor decomposition problem stated in our paper, our modifications to the vanilla objective and the vanilla GD are mostly motivated by theoretical challenges: reparameterize the objective to avoid high-order saddle points; re-initialize one component to escape bad local minimum; etc. Some of the changes, e.g., re-initialization of components, are extendable to neural network training, while others are more restricted to tensor decompositions.

"Numerical experiments":
We will add numerical experiments to verify our lower bound for lazy training on tensor decomposition problems (Theorem 1). We modified vanilla GD mostly because of the theoretical challenges in analyzing the optimization of tensor decompositions. We do not claim our algorithm outperforms SGD/Adam in training neural networks.

We will also fix the typos. Thanks for pointing them out.

To Reviewer 3:

"Concern on the over-parameterization:"
Recovering a rank r tensor using exactly r components is NP-Hard, so it's natural to use more components to fit the ground truth tensor. Besides, in this paper, instead of optimizing the degree of over-parameterization, we focus on studying the optimization of GD in over-parameterized tensor decompositions beyond the lazy training regime. Overparameterization plays an essential rule in the training of neural networks; it's also known that the training of networks can be cast as mixture of tensor decompositions (Ge et al., 2017). Therefore, we view this work as a first step towards understanding the training of over-parameterized neural networks.

"Non-standard objective function and optimization algorithm":
We modified the standard objective function and vanilla GD to overcome challenges in theoretical analysis. Most of these changes are well justified: reparameterize the objective to avoid high order saddle points; re-initialize one component to escape bad local minimum; regularize the objective to control parameter norms. Some others might be artifacts of our analysis: update separately on U and $C, \hat{C}$; switch the scalar mode when a component grows large. Proving similar guarantees on a more standard objective and a cleaner algorithm is an important future direction.

"Theorem 3 requires $r < d$?":
Theorem 3 holds for $r > d$ if we replace r by d in the bounds of m, λ, and K. However, in this setting, there are no benefit of using this approach compared to lazy training.

We will also fix the grammar issues. Thanks for pointing that out.

To Reviewer 4:

"Why do we need a_i?":
Each a_i is initialized as $+1$ or -1 and then fixed throughout the training. We need positive and negative a_i's so that our model can fit a "non-positive-definite" ground truth tensor, particularly when the order l is even. For example, if the ground truth tensor is $-v^{\otimes 4}$ for some vector v, our model cannot fit it if all a_i's are $+1$.

"Numerical experiments: compare normal GD and the proposed algorithm":
We modified the normal GD to overcome challenges in theoretical analysis: re-initialize one component to escape bad local minimum; update separately on U and $C, \hat{C}$ to contract B subspace. In the numerical experiments, from a random initialization, normal GD can also successfully optimize the model as the modified algorithm. However, it will require a significantly different proof as the proof needs to show why the trajectory of gradient descent does not go through any spurious local minima. We leave that as a future direction.

References

Ge, R., Lee, J. D., and Ma, T. (2017). Learning one-hidden-layer neural networks with landscape design. *arXiv preprint arXiv:1711.00501*.