We thank all reviewers for their constructive comments. We are encouraged that all voted to accept, and the acknowledgement of the importance of our work [R1, R2] and the comprehensiveness of our studies [R1, R2, R3]. We address specific comments below and will incorporate them to the updated version.

To R1: Q: Writing. Sorry for the hardness to follow in L212-L216. Recall that the basic cell in NSA is Aggregation-ReLU-Conv-BN (see L111). Particularly, the aggregation module is to combine the input data from multiple edges via a weighted sum (see Figure 5 in Appendix). To avoid introducing unaffordable architecture dependent parameters, we employ architecture dependent aggregation and BN in NSA-id, following the style of the class-conditional BN widely used in conditional generative modeling [*1]. Namely, we build an individual set of trainable aggregation coefficients and BN affine parameters for each architecture. We'll rewrite this part and add a figure to depict this explicitly. Regarding the mutual information (MI), a standard measure of uncertainty [33], we first calculate the MI of normal test samples and OOD (or adversarial) ones, based on which we directly distinguish the normal ones from OOD (or adversarial) ones. The underlying notion is that OOD (or adversarial) samples commonly deviate from the manifold of normal ones, thus have high uncertainty. We compute and report the AUC of such a binary classification (L301).

Q: Track the values of var(μ). Thanks for the advice. We sampled 1000 training images from CIFAR-10 and computed var(μ) of the last BN layer of a NSA and a NSA-i trained given $S = 5000$ architectures. We calculated the average variance over all the channels and spatial locations, and the results of NSA and NSA-i are 0.00214 and 0.00082, respectively, which testify the effectiveness of NSA-i. We'll track the full dynamics of var(μ) in the final version.

Q: Ensemble gain of NSA-id. At first, we clarify the ensemble gain of NSA-id is substantially more evident than that of NSA-i. We have also discussed the potential reasons of the quick saturation of ensemble accuracy in L221-224. In short, the introduced new parameters are rare, thus cannot adequately improve the weights diversity for ensembling.

Q: Root cause of mode collapse. Intuitively, the expectation w.r.t. architecture in NSA's training loss forces the shared weights to be robust against architecture variability. Given such weights, the trained NSA may behave consistently under diverse architectures, incurring mode collapse. To verify this, we assembled the 5 individuals with unshared weights introduced in L277, and got striking 2.36% error and 0.003 ECE on CIFAR-10, confirming the above intuition.

Q: Extension to DARTS search space. Although the investigated search space is simpler than that in DARTS, the issues of BN and weight sharing are shared between the spaces, and are observed frequently by the NAS community [46, *2]. We think the discovered phenomena and proposed solutions are insightful for general NAS, while a systematic investigation on general NAS is one of our future work.

Q: Training time. We clarify that we didn't perform searching. NSA's training time is almost identical to that of WRN-28-10†, e.g., 0.6 day on a GTX 2080Ti for 300 epochs (L115-116). The additional computations induced by the complicated connections are only summations in the aggregation modules, which are negligible as compared to the time-consuming convolutions.

Q: Comparison to DARTS and ENAS. DARTS and ENAS build networks with the parameter-efficient separable convolutions, while NSA adopts the regular convolutions following WRN. Thus, comparing NSA with DARTS and ENAS in the aspect of parameter number is not fair. Currently, the comparable baselines WRN-28-10† and *Average of individuals* are outperformed by NSA evidently. And we leave the application to DARTS space as future work.

To R2: Q: Extension and broader impact of NSA. Thanks for the advice. We'll try to extend NSA to regression tasks for uncertainty quantification and improve the broader impact in the final version. We'll add the NeurIPS paper.

To R3: Q: Regarding $p(\alpha)$. As stated in L98-101, $p(\alpha)$ is a uniform distribution over S randomly pre-fetched architectures by the ER-0.3 model. $p(\alpha)$ affects the architecture samples in the training (Eq. 3). When $p(\alpha)$ has larger support, the optimized weights may be more helpful for architecture generalizing, but more under-fitting (see Table 1).

Q: Regarding Eq.1 and Eq.3. Eq.1 is the loss commonly used for training network with stochastic architectures, as in SNAS [44], and the sampled architecture α is shared among all the instances in the mini-batch. Eq.3 uses instance specific architectures to compute the training loss, namely, sampling an individual architecture α_i for each instance (x_i, y_i) in the mini-batch. The loss in Eq.3 is averaged over all the instances.

Q: Regarding Figure 4. The performance drop in Figure 4 may stem from the facts that the 500 used architectures are randomly sampled and we perform only uniform ensemble instead of weighted ensemble. Thus assembling more base learners may not give rise to rigidly better predictions. But the overall trend of NSA-id is substantially superior to that of NSA-i. At last, we clarify that the first five ensemble accuracies of NSA-id in Figure 4 are 0.9613, 0.9648, 0.9658, 0.9659, 0.9659, while those of NSA-i are 0.9616, 0.9641, 0.9635, 0.9634, 0.9636. The comparisons confirm the claim "ensemble gain is more obvious compared to NSA-i".

[*1] Takeru Miyato and Masanori Koyama. cGANs with Projection Discriminator.

[*2] Zhang et al. Deeper Insights into Weight Sharing in Neural Architecture Search.