We sincerely thank all the reviewers for their careful readings and valuable comments. We will address the reviewers' comments point by point.

Table A: **Updated part of ImageNet results.** In the paper for our original experiment on ImageNet, we use the same hyper-parameters of that for CIFAR10. In the new experiment we only change the learning rate of RL for agent from 0.1 to 0.2. The resulting policy is used to train Res50 and Res200 for Ours (without EB) and Ours (4× or 2× EB). By leveraging EB with a batch size ratio r ($r\times$EB), the mini-batch size is r times larger while the number of iterations is not changed (line 240). *Ours (without EB)* denotes our approach without using EB. *Ours (4× or 2× EB)* denotes our approach using 4 times the mini-batch size for Res50 and 2 times the mini-batch size for Res200. Please note that *Adv. Aug* used 8 times the mini-batch size.

Approach	Res50	Res200
OHL (without EB)	21.07 / 5.68	-
Adv. Aug (8×EB)	20.60 ± 0.15 / 5.53 ± 0.05	18.68 ± 0.18 / 4.70 ± 0.05
Ours (without EB)	20.61 ± 0.17 / 5.49 ± 0.08	18.64 ± 0.16 / 4.67 ± 0.07
Ours (4× or 2× EB)	**20.36 ± 0.15 / 5.41 ± 0.07**	**18.56 ± 0.14 / 4.62 ± 0.05**

R#1 Q1: In Table 2, why did the authors omit AdvAA? The authors should add this comparison.

A: **1)** In the caption of Table 2, we explained that AdvAA is not included here since the EB used in AdvAA is not used in the other compared approaches in Table 2, making it unfair to compare AdvAA with other approaches and ours. **2)** AdvAA has evaluated its performance for different EB ratios $r \in \{2, 4, 8, 16, 32\}$. The test accuracy improves rapidly with the increase of r up to 8. The further increase of r does not bring a significant improvement. So $r = 8$ is finally used in AdvAA. Following **R#4**'s advice on the mini-batch size, we tried to increase the batch size, but we can only use 4×EB for Res50 and 2×EB for Res200 due to the limited resources. As shown in Table. A, our approach can still achieve performance on a par with AdvAA without applying EB. And we've achieved better performance with even smaller r on both Res50 and Res200 backbones. We will update the results and some claims accordingly in our revision. Thanks for the kind suggestion.

R#3 Q1: In figure 1, the change of rank cannot indicate that the model is unstable ...

A: Sorry for the confusion. We will change into: ... the compromised evaluation process would distort the ranking for augmentation strategies since the rank for the models trained with too few iterations are known to be inconsistent with the final models trained with sufficient iterations.

R#3 Q2: The performance improvements are not very significant when comparing to recent methods. On ImageNet, resnet-50 is only 0.3 better than OHL. Also, the std is very large.

A: **1)** On CIFAR10, the performance gains of our AWS are substantial. On CIFAR100 and ImageNet, AWS still achieves superior performance over many recent works without tuned hyper-parameters, i.e., using the same hyper-parameters as CIFAR10's. **2)** On ImageNet, OHL stated in the paper that they manually tuned the agent learning rate and used the best one but we did not. We conducted a new search by simply doubling the agent learning rate of ours. Table. A shows AWS achieves around 0.5% accuracy improvement over OHL. **3)** Besides, we would like to point out that, rather than being 'very large', our std is comparable with AdvAA's, while many other works do not report their std.

R#3 Q3: In table 1 ... no std reported.

A: Sorry for the confusion. In line 229 we stated that our test error rates with std are given in the supplementary (line 52), which conflicted with the caption of table 1. We will revise the caption of Table 1.

R#4 Q1: 20.73+0.17 = 20.9, which is close to 21.07 from OHL.

A: **1)** This comparison would be unfair. Please note that, 20.73+0.17 is the worst case for our AWS, while 21.07 could be the mean or even best performance for OHL since it did not report std. **2)** The updated result in Table. A shows that AWS achieves around 0.5% accuracy improvement over OHL. Please refer to **R#3 Q2 2)** for more details.

R#4 Q2: Evaluate the key observation for other tasks ...

A: Thanks for the valuable advice. We've evaluated our key observation in the 'pre-train fine-tune' paradigm. We first pre-train the Res18 on CIFAR100 for 200 epochs without augmentation. Then we transfer it to CIFAR10 to train (fine-tune) 100 epochs and conduct a similar experiment in Sec. 3.1.1. Figure. A is similar to Fig. 2 in our paper, showing that data augmentation is more effective in the later training process. Furthermore, we agree that further research can be conduct based on this work. As our future work, we will evaluate our key observation for more different tasks, such as object detection and semantic segmentation.

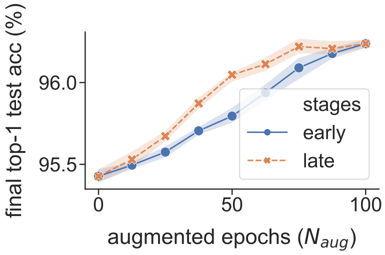

Figure A: The investigation on a transfer learning task.