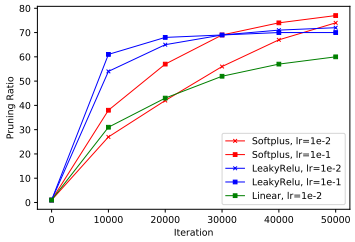
Review1: Thank you for the review and we appreciate the insightful suggestions. We will improve the paper accordingly:

In the related work section, we will first introduce structured and unstructured pruning and summarize existing papers in these two categories, respectively. We also will include two latest papers as follows:

Gomez *et al.* [2019] proposed to keep neurons with high magnitude and prune only neurons with smaller magnitude in a stochastic way. The accuracy is maintained by reducing the dependency of important neurons on unimportant neurons. Liu *et al.* [2018] did comprehensive experiments showing that training-from-scratch on the right sparse architecture yields better results than pruning from pre-trained models. Such a result implies that the pruning process is actually finding the right network structure, thus bridging the gap between pruning and architecture search.

Our paper can be further positioned as a structure learning procedure where a neuron's importance is learned by the attached trainable auxiliary parameter. Unimportant neurons will be pruned if the corresponding auxiliary parameter decreases below a threshold, and important neurons will be recovered otherwise.

For the experiment part, thank you for referring the AutoSlim paper. We did additional experimental comparison on MobileNet v1 and v2 with the AutoSlim paper. Our model achieves better performance than AutoSlim under extreme cases, where FLOPs are compressed down to 150M and 209M, respectively. We will include more neuron pruning results and discussion on ResNet50, SuffleNet, and MNasNet to replace the LeNet5 part in the camera ready version. We also include the performance of different STEs on VGG-like below, and will add more ablation discussion and results. Compared to Softplus STE, LeakyRelu converge faster but with lower CR, and linear STE converge slower with lower CR. In addition, figure 2(c) in the paper shows the comparison with and without forcing the sign.



Model	MobileNet v1			MobileNet v2		
	Base	AutoSlim	Ours	Base	AutoSlim	Ours
FLOPs	325M	325M	325M	300M	305M	300M
Top1-Err	31.6	**28.5**	28.8	28.2	**25.8**	26.1
FLOPs	150M	150M	150M	209M	207M	209M
Top1-Err	36.7	32.1	**31.8**	30.2	27	**26.7**

Review2: Thank you for the comments of the paper especially on the clarity and potential improvement. We will mention before Eq. (9) that R_n refers to n-norm regularization. We included some theoretical analysis on the correlation gradient direction consistency in the paper which ensures the effectiveness of gradient descent. We will also include the necessary convergence discussion in the appendix.

Review3: We would like to thank reviewer for the valuable thoughts and detailed comments on the paper presentation.

Originality: The significant difference from other mask based pruning lies on the weak-correlation between the weights and auxiliary parameters. We decoupled the correlation between update rule and weight magnitude which is harmful to pruning task, as is discussed in Sec3.3 and figure 2(c). Theoretical analysis and experimental results show such weak correlation leads to a more compact and more accurate model.

a) We agree that the masking formulation has been adopted by existing papers to learn sparse NNs, but optimizing the binary masks effectively, stably and efficiently requires significant research efforts. The modified update rule results in a better local minima of the auxiliary parameter network for the pruning task. Moreover, we have compared with [1] in the introduction section that, "In contrast to Louizos et al.[2018], our method avoids inefficient and high variance single step Monte-Carlo sampling and places no assumptions on the prior distribution". We also show in Table 1 that our model significantly outperforms [1]. b) The update rule is carefully designed as mentioned above. The way of calculating the partial derivations follow coarse gradients of predefined straight through estimators as discussed in Sec3.2. Specifically, our Tensorflow implementation employs gradient overriding function for coarse gradient calculation.

Quality: 1) We agree that the high scalability and training-from-scratch ability mentioned in 4) and 5) validate the significance of the proposed method. We will reorganize the list of contributions based on your nice suggestion. 2) The update rule is not following an accurate gradient w.r.t. m, but a direction that forms an acute angle with the original gradient direction. Here we kind of abuse the word "gradient" since it is the actual direction that auxiliary parameters will be updated on. We will rephrase it as "updating rule" or "modified gradient" to avoid potential misunderstanding. 3) Thank you for pointing out the inaccurate naming of "sensitivity consistency". We will rename it "sensitivity decoupling", since the update rule of auxiliary parameters is decoupled with the magnitude of weights. 4) Thank you for your suggestion to improve the equations' consistency. We will add the regularization term $\lambda\mathcal{R}(W)$ of weight W to Eq.(3). 5) We will add the following additional explanation on recoverable pruning to Sec3.4: Recovering a weight w_{ij} requires the gradient of m_{ij} moving toward positive direction. The weight decay will decrease the magnitude of w_{ij} and provide a negative gradient to m_{ij}, which reduces the recoverability. 6) There is a typo in the sentence. We will rephrase it to "The goal is to find the minimum subset $\{w : w \in \mathcal{W}\}$ that preserves the model accuracy."