

## SUMMARY OF THE APPENDIX

This appendix contains additional details, including mathematical proofs, experimental details and additional results. The appendix is organized as follows:

- Section A contains proof of Proposition 1.
- Section B lists the statistics of the datasets and training details.
- Section C introduces the results of some readout functions which aims at comparing removing methods in a different way, such as relearn time, activation distance and membership inference attack results.
- Section D are the results of removing mechanisms on other datasets and model architectures, which is a supplement to the Figure 2.
- Section E shows the results of removing mechanisms with limited remain data, which is a supplement to the Figure 3.

### A PROOF OF PROPOSITION 1

A property of Fisher matrix  $F$  is that it can be used to approximate KL-divergence between two distributions  $p(y|x, w), p(y|x, w')$  (by Taylor expansion of  $\text{KL}(w, w')$  with respect to  $w'$ ),

$$\text{KL}(w, w') = \mathbb{E}_{x,y} p(y|x, w) \log \frac{p(y|x, w)}{p(y|x, w')} \approx \frac{1}{2} (w - w')^T F (w - w').$$

*Proof.* First we recall that the stationaries of  $\mathcal{L}(w, D)$  satisfy normal equation,

$$XX^T w^* = |D| F w^* = b,$$

where  $X = [x_1, x_2, \dots, x_{|D|}]$ ,  $b = \sum_{i=1}^{|D|} y_i x_i$ . Denote  $b_r = \sum_{i=1}^{|D_r|} y_i x_i$  and  $b_f = \sum_{i=1}^{|D_f|} y_i x_i$ .

We approximate KL-divergence with Fisher matrix,

$$\begin{aligned} \text{KL}(w_r^*, \hat{w}_r) &\approx \frac{1}{2} (w_r^* - \hat{w}_r)^T F (w_r^* - \hat{w}_r) \leq \frac{\lambda}{2|D|} \sum_j (w_{r,j}^* - \hat{w}_{r,j})^2 \\ &= \frac{\lambda}{2|D|} \left( \sum_{j \in M} w_{r,j}^{*2} + \sum_{j \notin M} (w_{r,j}^* - w_j^*)^2 \right) \end{aligned}$$

By plugging in the weights (from normal equations), we have,

$$\begin{aligned} \sum_{j \in M} w_{r,j}^{*2} &= \sum_{j \in M} \frac{1}{F_{r,jj}^2} b_{r,j}^2 \leq c_1 \sum_{j \in M} \frac{1}{F_{r,jj}^2}, \\ \sum_{j \notin M} (w_{r,j}^* - w_j^*)^2 &= \sum_{j \notin M} \left( \frac{1}{F_{jj}} b_i - \frac{1}{F_{r,jj}} b_{r,j} \right)^2 \\ &= \sum_{j \notin M} \left( \frac{1}{F_{jj}} b_{f,j} - \frac{F_{f,jj}}{F_{jj} F_{r,jj}} b_{r,j} \right)^2 \\ &\leq \sum_{j \notin M} \frac{2}{F_{jj}^2} b_{f,j}^2 + \sum_{j \notin M} \frac{2}{F_{jj}^2} \left( \frac{F_{f,jj}}{F_{r,jj}} b_{r,j} \right)^2 \\ &\leq c_2 \sum_{j \notin M} \frac{2}{F_{r,jj}^2} + c_1 \sum_{j \notin M} \frac{2}{F_{jj}^2} \left( \frac{F_{f,jj}}{F_{r,jj}} \right)^2, \end{aligned}$$

where  $c_1 = \max_j b_{r,j}^2$ ,  $c_2 = \max_j b_{f,j}^2$ . The last inequality is based on the fact  $F_{r,jj} \leq F_{jj}$ . Putting them together we have,

$$\begin{aligned} \sum_{j \in M} w_{r,j}^{*2} + \sum_{j \notin M} (w_{r,j}^* - w_j^*)^2 &\leq c_1 \sum_{j \in M} \frac{1}{F_{r,jj}^2} + c_2 \sum_{j \notin M} \frac{2}{F_{r,jj}^2} + c_1 \sum_{j \notin M} \frac{2}{F_{jj}^2} \left( \frac{F_{f,jj}}{F_{r,jj}} \right)^2 \\ &\leq c + 2c_1 \sum_{j \notin M} \frac{1}{F_{jj}^2} \left( \frac{F_{f,jj}}{F_{r,jj}} \right)^2, \end{aligned}$$

where  $c = \max\{c_1, 2c_2\} \sum_j \frac{1}{F_{r,jj}^2}$ .

□

## B EXPERIMENT SETTINGS

We list the data statistics and training setups in Table 3 and Table 4.

Dataset	CIFAR10	CIFAR100	MNIST	Tiny-ImageNet
# images	50K/10K	50K/10K	60K/10K	100K/10K
# class	10	100	10	200
Img Size	32*32	32*32	28*28	64*64

Table 3: Statistics on datasets.

Experiments	CIFAR10/100	MNIST	Tiny-ImageNet
Training epochs	160	30	160
Batch size	128	128	32
Init learning rate	0.1	0.1	0.1
Optimizer	SGD	SGD	SGD
Learning rate scheduler	step	N/A	step
Learning rate decay (epoch)	[80, 120]	N/A	[80, 120]
Learning rate decay factor	10	N/A	10
Momentum	0.9	0.9	0.9
Warmup epochs	0	0	20

Table 4: Detailed experiment setups.

## C ADDITIONAL READOUT FUNCTIONS

We use different evaluate metrics to compare different removing mechanisms, include: (i) Re-learn time (in epochs) for unlearned model to recover performance on  $D_f$  while training on the whole dataset  $D$ , (ii) Activation distance between unlearned model  $\hat{w}_r$  and model trained without forget set  $w_r^*$ , (iii) Success rate of membership inference attack on the forget data  $D_f$ . The membership inference is formulated as a binary classification task, and we use two-layer fully connected network of width 256 and 128. All the evaluate metrics is ideally as the same as the model trained without forget data  $w_r^*$ .

### C.1 RELEARN TIME

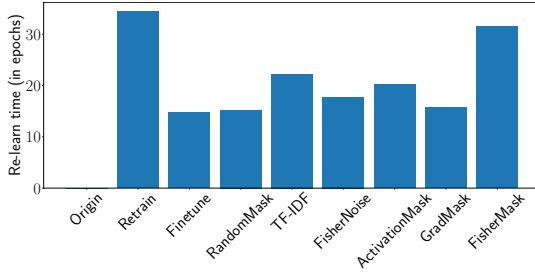


Figure 6: Re-learn time (in epochs) for removing mechanisms.

Figure 6 shows the results of relearn time (in epochs) on dataset  $D$ . We count the epochs for fine tuning the unlearned model to achieve the same loss on the forget data  $D_f$  as original model. We experiment on the 4 main settings and calculate the average epochs for each removing mechanism. And the results show that FisherMask method performs well on the re-learn time readout function, and TF-IDF, ActivationMask method have a longer re-learn time comparing with Finetune baseline. The results indicate that although we can unlearn the information to make the accu-

racy of unlearn category approach zero, there still remains information inside the model to recover performance quickly when original dataset is provided.

Settings\Criterions	Original		RansomMask		Finetune		TF-IDF		FisherNoise		ActivationMask		GradMask		FisherMask	
	U	R	U	R	U	R	U	R	U	R	U	R	U	R	U	R
ResNet20_CIFAR10	1.89	0.19	0.78	0.19	0.80	0.19	0.79	0.19	1.34	0.42	0.80	<b>0.18</b>	0.79	0.19	<b>0.76</b>	<b>0.18</b>
GoogLeNet_CIFAR100	1.84	0.48	1.45	0.48	1.61	0.48	1.43	0.48	1.58	1.64	0.88	0.49	0.81	0.50	<b>0.77</b>	<b>0.48</b>
MNIST_VGG16	2.00	<b>0.02</b>	1.19	0.05	0.88	<b>0.02</b>	<b>0.61</b>	<b>0.02</b>	1.74	1.18	1.30	<b>0.02</b>	1.37	0.03	1.54	<b>0.02</b>
DenseNet_Tiny-ImageNet	1.79	<b>0.68</b>	1.54	0.70	1.52	0.70	1.50	0.70	<b>1.04</b>	0.84	1.47	0.70	1.28	0.70	1.07	0.70

Table 5: Results of activation distance between unlearned model with various unlearn mechanisms and re-trained model without seeing forget set. Activation distance is computed as:  $\mathbb{E}_{x \sim p(x)} [\|f_{\hat{w}_r}(x) - f_{w_r^*}(x)\|_1]$ .  $U$  and  $R$  indicate forget and remain set, respectively.

Settings\Criterions	Original	RansomMask	Finetune	TF-IDF	FisherNoise	ActivationMask	GradMask	FisherMask
ResNet20_CIFAR10	0.82	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>
GoogLeNet_CIFAR100	1.00	0.22	0.52	0.23	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>
MNIST_VGG16	1.00	0.43	0.40	0.25	<b>0.01</b>	0.56	0.31	0.41
DenseNet_Tiny-ImageNet	0.71	0.31	0.32	0.31	<b>0.00</b>	0.21	0.06	<b>0.00</b>

Table 6: Recall of membership inference attack on forget set  $D_f$ .

### C.2 ACTIVATION DISTANCE

Table 5 shows the results on subset  $U$  and  $R$  of activation distance for models combining different removing strategies. We use the activation distance (Golatkar et al., 2020b) to measure the distance between unlearned model  $\hat{w}_r$  and retrained model  $w_r^*$ :  $\mathbb{E}_{x \sim p(x)} [\|f_{\hat{w}_r}(x) - f_{w_r^*}(x)\|_1]$ , where  $f_w(x)$  is the activation (post-softmax) on  $x$  with parameter  $w$ .

From the results, we can find that unlearned model with FisherMask method is closest to the re-trained model on CIFAR10/100 dataset. ActivationMask and GradMask methods achieve a similar performance on both datasets, while TF-IDF, FisherNoise, RandomMask and Finetune methods show a good performance on CIFAR10, whereas do not on CIFAR100. On dataset MNIST,

TF-IDF method achieves a better performance on both subset  $U$  and  $R$ . Other methods have a larger activation distance on dataset  $U$  and all methods have a similar good performance on dataset  $R$  except FisherNoise. On dataset Tiny-ImageNet, FisherNoise method achieves a better forget performance on dataset  $U$  but has a largest activation distance on remain dataset  $R$ . And FisherMask still has a relatively good results on both  $U$  and  $R$  dataset compared to the rest methods.

### C.3 MEMBERSHIP INFERENCE ATTACK

Here we use Membership Inference to test the ability of data deletion and evaluate how much could the model leak the information about forget set. Consider the adversary attempts to query the model and guess a particular sample (or a particular class) was used to train the model. And the unlearning mechanism fails if the adversary inspect the existence of deleted data from the unlearned model. We consider the scenario where the adversary only has black-box access to the model, which means the adversary can only get the input and output without knowing the model architecture.

The attack model is trained with the shadow training technique following Shokri et al. (2017), and the goal is to recognize the intrinsic differences in the behavior of target model and distinguish from members and non-members according to the model output. We use 20 shadow models to imitate the behavior of target model and each of them is trained on a similar dataset as the target model to strengthen the attack model. We randomly split the training set in half to create member dataset and non-member dataset for each shadow model. And the attack model is trained on a syntactic dataset which is constructed by the output probabilities of shadow models labeled with the ground truth about the membership.

Here we present the membership inference attack results on Table 6. We list the recall on the forget dataset  $D_f$ . Considering the original model is trained on the dataset  $D$ , the attack model should have a high recall on the training subset  $D_f$ . And if the unlearn method could unlearn information successfully, then the attack model could fail to gain information about  $D_f$ . The results show that FisherNoise method could unlearn completely on all the four datasets. FisherMask and GradMask method have similar performance, both of them unlearn completely on all datasets except MNIST. ActivationMask performs well on the CIFAR10/100 dataset, but can not avoid information leaking on the remain datasets. The rest baselines RandomMask, Finetune and TF-IDF can only forget completely on CIFAR10 dataset.

## D RESULTS OF REMOVING MECHANISMS

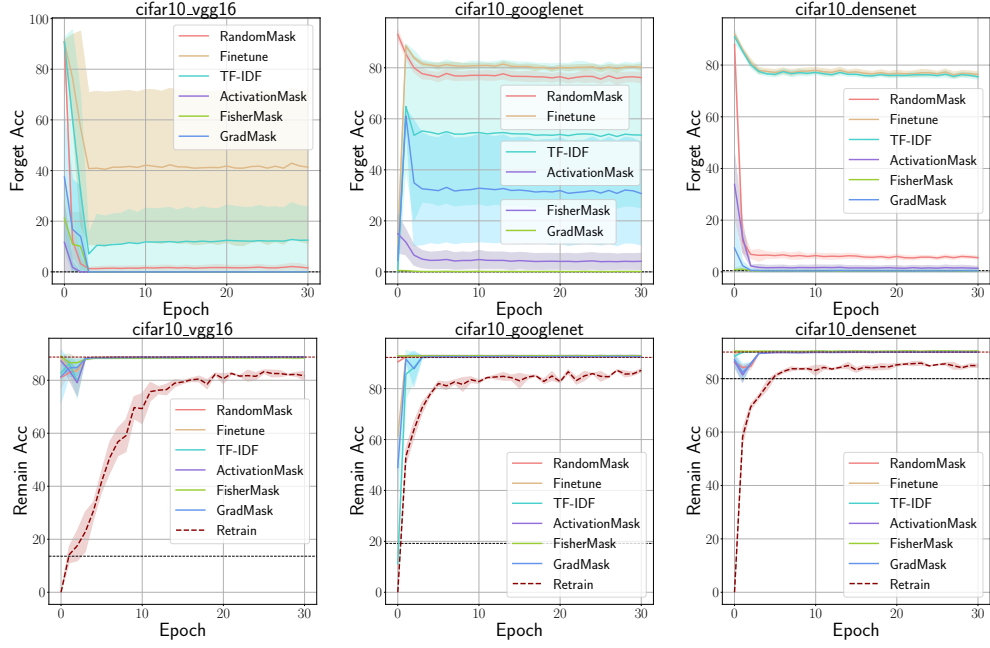


Figure 7: Results of different architectures on CIFAR10 dataset.

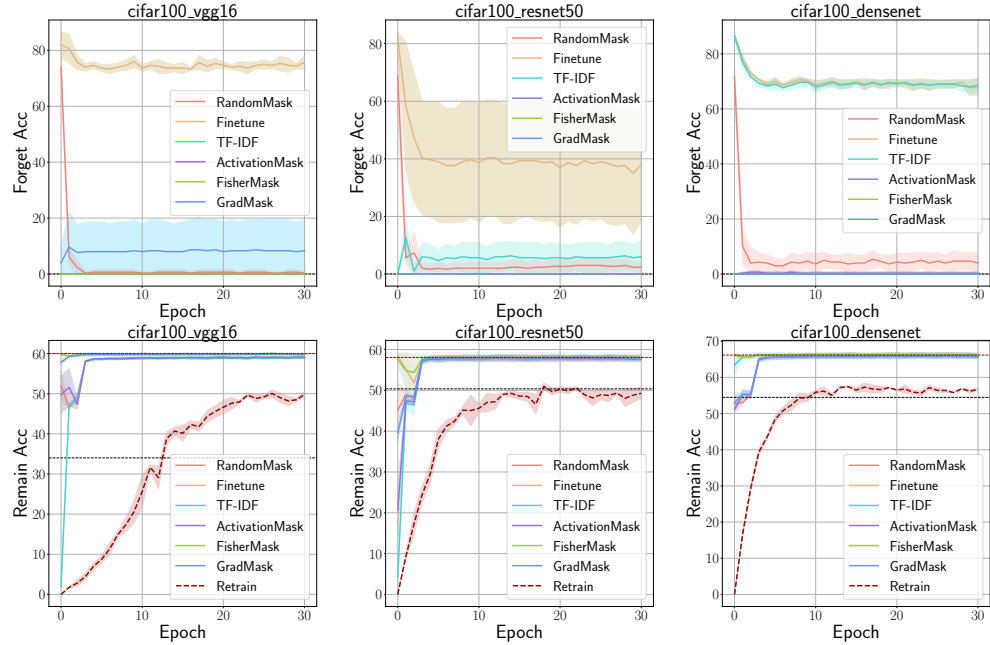


Figure 8: Results of different architectures on CIFAR100 dataset.

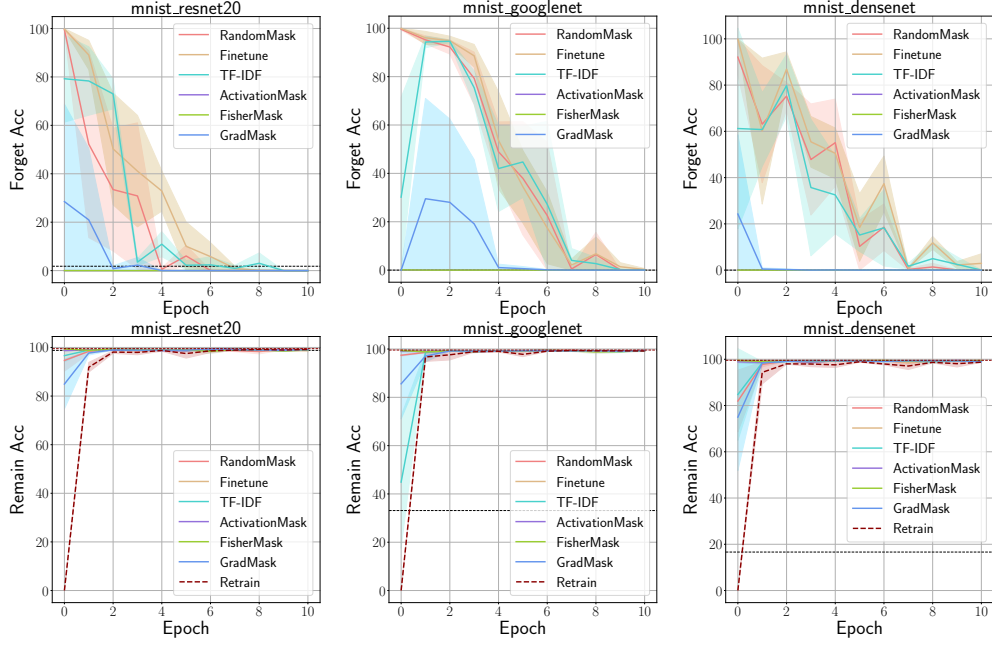


Figure 9: Results of different architectures on MNIST dataset.

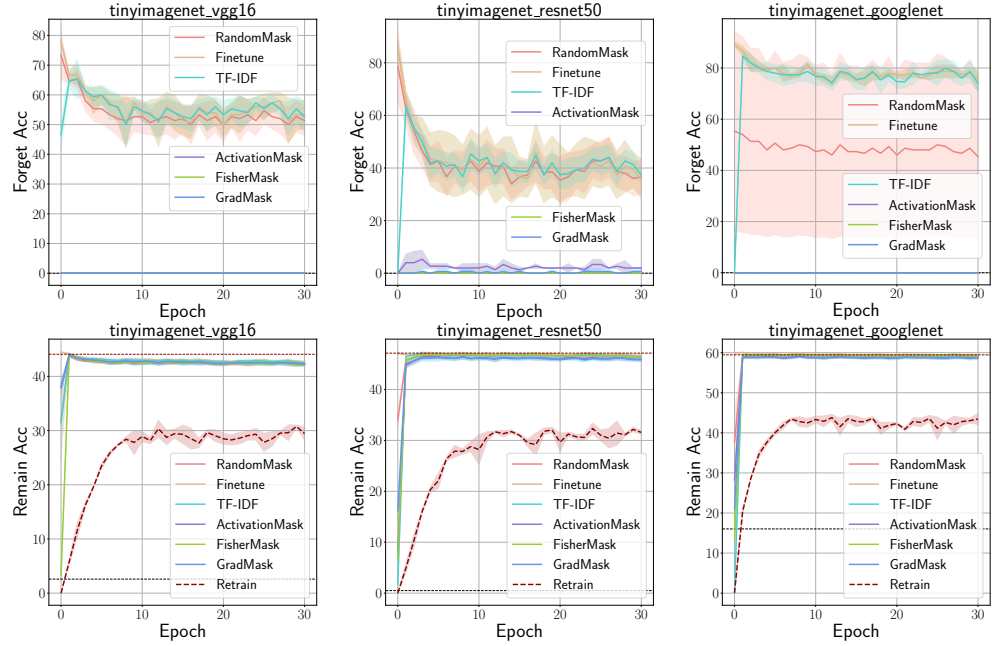


Figure 10: Results of different architectures on Tiny-ImageNet dataset.

## E EXPERIMENT WITH LIMITED REMAIN DATA

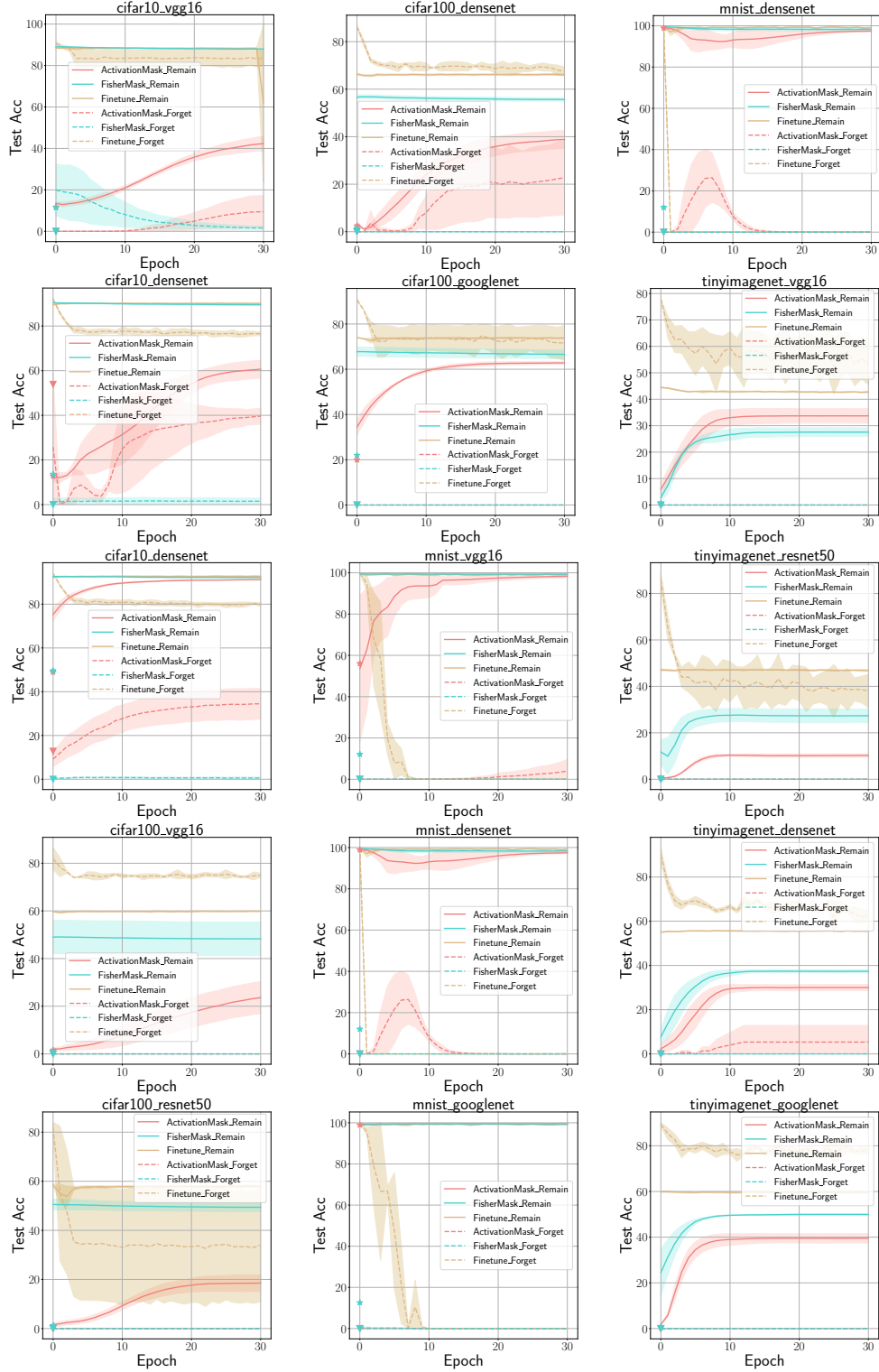


Figure 11: Results on different architectures with limited remain data.