Appendix of
Unified Knowledge Maintenance Pruning and Progressive Recovery with Weight Recalling for Large Vision-Language Models

Zimeng Wu[1,2], **Jiaxin Chen**[1,2*], **Yunhong Wang**[1,2]

[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China
[2]School of Computer Science and Engineering, Beihang University, Beijing, China
{zimengwu, jiaxinchen, yhwang}@buaa.edu.cn

In this document, we additionally report more implementation details of the experiments in Sec. A, provide detailed principles of error in Fisher Information theory with externally sourced calibration data in Sec. B, present ablation studies on certain hyperparameters and similar comparative settings in Sec. C, and demonstrate representative generation examples in Sec. D.

A. Implementation Details

In our experiments, we set $\mathcal{T}$ in Eq.(8) to 0.4. In Eq.(12), both r_1 and r_2 are set to 8, with the scaling factor α set to 16. In Eq.(13), we set β_1 to 1e6, β_2 to 1e9 and 1e8 for the LLM encoder and decoder components, respectively. Additionally, Tab. A presents detailed configurations for the 3 phases for distillation. Within each sub-phase, the learning rate is set to 2e-5 with 50 warm-up steps and a batch size of 20. We employ the AdamW optimizer with a weight decay of 0.05. All the other settings are consistent with the pre-trained model.

B. Error in Fisher Information with Externally Sourced Calibration Data

We first demonstrate the standard derivation based on the Fisher Information theory. Let θ denote the parameters of a well-trained model, which, in our context, will be pruned according to gradient-based importance estimation. Let $f(X;\theta)$ be the probability density function conditioned on θ given the observation of X. With $\log f(x;\theta)$ twice differentiable *w.r.t.* θ, we get:

$$\begin{aligned}\frac{\partial^2}{\partial\theta^2}\log f(X;\theta) &= \frac{\frac{\partial^2}{\partial\theta^2}f(X;\theta)}{f(X;\theta)} - (\frac{\frac{\partial}{\partial\theta}f(X;\theta)}{f(X;\theta)})^2 \\ &= \frac{\frac{\partial^2}{\partial\theta^2}f(X;\theta)}{f(X;\theta)} - (\frac{\partial}{\partial\theta}\log f(X;\theta))^2.\end{aligned} \tag{a}$$

When X is actually distributed as $f(X;\theta)$, the expectation of the first term in Eq.(a) can be derived as follows:

$$\begin{aligned}E\left[\frac{\frac{\partial^2}{\partial\theta^2}f(X;\theta)}{f(X;\theta)}\middle|\theta\right] &= \int_{\mathbb{R}}\frac{\frac{\partial^2}{\partial\theta^2}f(x;\theta)}{f(x;\theta)}f(x;\theta)dx \\ &= \int_{\mathbb{R}}\frac{\partial^2}{\partial\theta^2}f(x;\theta)dx = \frac{\partial^2}{\partial\theta^2}\int_{\mathbb{R}}f(x;\theta)dx \\ &= \frac{\partial^2}{\partial\theta^2}1 = 0,\end{aligned} \tag{b}$$

where the third equality relies on the feasibility of switching the order of the integration and the twice differentiation. Then apply the expectation observed by X to both sides of Eq.(a), and with the aid of Eq.(b), we obtain the negative format for Fisher Information:

$$E\left[\frac{\partial^2}{\partial\theta^2}\log f(X;\theta)\middle|\theta\right] = -E\left[(\frac{\partial}{\partial\theta}\log f(X;\theta))^2\middle|\theta\right]. \tag{c}$$

Finally, since $\mathcal{L}$ in Eq.(2) is a special case of the negative log-likelihood, the diagonal element of Hessian matrix H_{ii} can be derived as Eq.(d) according to Eq.(c).

$$H_{ii} = E\left[\frac{\partial^2}{\partial\theta^2}\mathcal{L}\middle|\theta\right] = E\left[(\frac{\partial}{\partial\theta}\mathcal{L})^2\middle|\theta\right]. \tag{d}$$

However, when X is externally sourced or conditionally filtered, which means X is distributed as $f(X;\psi)$, the expectations are all conditioned with ψ instead of θ. In this case, Eq.(b) becomes:

$$E\left[\frac{\frac{\partial^2}{\partial\theta^2}f(X;\theta)}{f(X;\theta)}\middle|\psi\right] = \int_{\mathbb{R}}\frac{\frac{\partial^2}{\partial\theta^2}f(x;\theta)}{f(x;\theta)}f(x;\psi)dx \neq 0. \tag{e}$$

This leads to an increased error in Eq.(c) and subsequently in Eq.(d). Therefore, we discard the estimation of the second-order term and instead aim to enhance the accuracy of the first-order term using Eq.(8).

C. Further Ablations

On Modality-wise Normalization method.

We investigate four different modality-wise normalization methods (*i.e.* M-normalize in Tab. B), including the commonly used min-max normalization, standardization and the

*Corresponding Author

Phase	VE[0:23:2]	VE[24:39:2]	LE[1:12:2]	LE[13:24:2]	LD[1:12:2]	LD[13:24:2]	Data Volumn
τ_1	✓						20K
	✓	✓					20K
τ_2	✓	✓	✓				80K
	✓	✓	✓	✓			40K
	✓	✓	✓	✓	✓		80K
	✓	✓	✓	✓	✓	✓	80K
τ_3	✓	✓	✓	✓	✓	✓	275K

Table A: Layers activated and data volume utilized for the 3 phases in Eq. (13). VE, LE, LD for visual encoder, language encoder and language decoder, respectively. $[x : y : 2]$ represents extracting a range of layers from x to y with a step size of 2.

Module	Setting	Method	Acc(%)
UKMI-UN	M-normalize	w/o normalize	25.43
		mean w/ iterative	35.15
		mean w/o iterative	34.52
		min-max	33.55
		standardize	34.79
UKMI-TS	gradient terms	first w/ thresh	35.15
		first w/o thresh	30.89
		second w/ thresh	5.48
		second w/o thresh	3.59
		mix w/ thresh	34.60
		mix w/o thresh	32.06
UKMI-EI	angle distribution	w/o entropy	29.62
		cosine	35.15
		arccosine	33.97
LPD	loss function	MSE+KL+Task	33.41
		KL+Task	31.01
		Task	30.08
		w/o progressive	28.67

Table B: Effect of different normalization methods, different Taylor importance terms, different angle distribution calculation methods on UKMI, and different loss functions on LPD by GQA at 50% pruning ratio. UN for the unifying on modules and modalities, TS for the Taylor selection over next token prediction, EI for the entropy injection for LLM component. Acc for Accuracy. Settings utilized by our method are underlined.

choice of whether to perform iterative normalizing. These methods help reduce the imbalance between modalities and thereby enhance performance. Among them, our employed scheme of mean normalization, combined with iterative normalization achieves the best results.

On Terms of Taylor Importance.

We conduct experiments with various combinations of Taylor Importance estimation and the sub-task filtering. As shown in Tab. B, using the second-order term alone provides little insight into the actual parameter importance within the model, which is consistent with our earlier discussion on the necessity of retaining the first-order term for LVLMs. While the second-order term can indeed aid in estimating parameter importance when the first-order term is dominant, it is still less effective than discarding the second-order term and

Threshold	Backpropagated Tokens	CIDEr	SPICE
$\mathcal{T} = 1.0$	9263	94.69	12.78
$\mathcal{T} = 0.8$	6801	97.23	13.10
$\mathcal{T} = 0.7$	6111	97.72	13.05
$\mathcal{T} = 0.6$	5339	97.74	13.07
$\mathcal{T} = 0.5$	4367	**98.06**	13.11
$\mathcal{T} = 0.4$	3178	98.01	**13.12**
$\mathcal{T} = 0.3$	2195	96.49	12.92

Table C: Ablation study on the threshold (*i.e.* $\mathcal{T}$) of Taylor selection on the image captioning task at 50% pruning ratio. $\mathcal{T} = 1.0$ for no filtering mechanism applied. Best in bold.

improving the accuracy of the first-order estimation by selecting poorly fitted sub-tasks. We also conduct experiments to investigate the impact of the hyperparameter $\mathcal{T}$ in Eq.(8). As summarized in Tab. C, appropriately filtering well fitted token prediction sub-tasks can mitigate the degradation during pruning, with the optimal performance observed around $\mathcal{T} = 0.4$ or 0.5.

On Angle Distribution of Entropy Injection for LLM Component.

To quantify the dispersion of angles between token and weight vectors, entropy of cosine values are employed in our method. Using the angles directly (*i.e.* $\arccos \phi_{i,s}$ in Eq.(9)) with the same number of bins yields suboptimal results, as shown in Tab. B. We attribute this to the non-linear interval division introduced by the arccosine transformation, which unnecessarily emphasizes certain directions.

On Loss Functions for Recovery.

Different loss functions in our proposed progressive learning strategy are also examined, as shown in Tab. B. Both the distribution supervision from output layers and the intermediate feature supervision contribute to facilitating model recovery. However, using all the three losses throughout the entire process will lead to worse recovery, as the MSE loss provides overly strong constraints, which introduces negative optimization objectives in the long term, particularly at high pruning ratios.

D. Generation Examples

We present some generation results of pruned models on image captioning and VQA tasks, as shown in Fig. A and

LLM-pruner global:	a car driving down a street
LLM-pruner local:	a car window with a house in the background
ECoFLaP_sp:	a man sitting in a car looking out the window
UPop:	a man sitting in the back seat of a car
FLAP:	a man sitting in the back seat of a car
UKMP(Ours):	**a man sitting on a bench in front of a building**

LLM-pruner global:	a basket filled with food on a table
LLM-pruner local:	a basket of food and a book on a table
ECoFLaP_sp:	a basket filled with food on a table
UPop:	a basket of food on a table
FLAP:	a basket of bread on a table
UKMP(Ours):	**a basket of bread and a newspaper on a table**

LLM-pruner global:	a street sign on the side of a building
LLM-pruner local:	a street sign that reads,
ECoFLaP_sp:	a street sign in front of a building
UPop:	a building with a street sign on it
FLAP:	street signs in front of a building
UKMP(Ours):	**street signs on a pole in front of a building**

LLM-pruner global:	a hallway in a home
LLM-pruner local:	a hallway in a home
ECoFLaP_sp:	a bathroom with a tiled floor
UPop:	a bathroom with tiled walls and a tub
FLAP:	a bathroom with a tile floor
UKMP(Ours):	**a bathroom with white cabinets and a tiled floor**

Figure A: Comparison of generation results of models pruned by different methods on image captioning at 50% pruning ratio.

Fig. B, respectively. Inevitably, at high pruning ratios, there is degradation of both visual perception and language generation capabilities within models. Notably, models pruned by most other methods, though still able to generate meaningful image captions, suffer from misdescription of object relationships or failure in accounting for multiple objects in the image. In contrast, the model pruned by our UKMP demonstrates superior performance. As for the VQA tasks, models pruned by other methods often fail to understand the posed questions, resulting in irrelevant or even nonsensical answers. However, our UKMP better maintains the model's strong performance in comprehending various kinds of questions (*i.e.* general questions, special questions and alternative questions), counting, and external knowledge association.

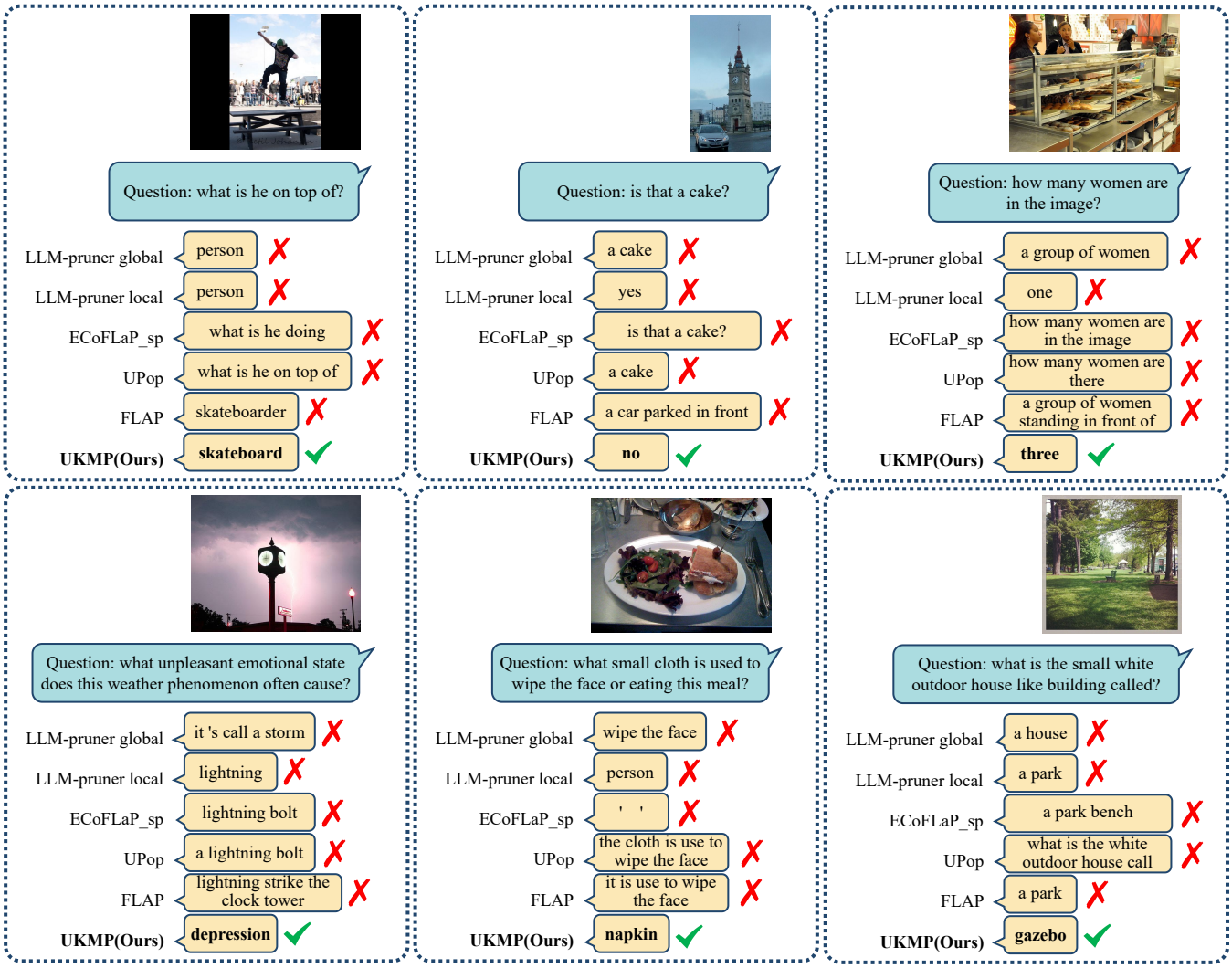

Figure B: Comparison of generation results of models pruned by different methods on VQA at 50% pruning ratio. Incorrect answers and correct answers are marked with red crosses and green checks, respectively.