Rethinking Token Reduction for Large Vision-Language Models

Supplementary Material

8. Implementation Details

8.1. Fixed Compression Matrix

The training algorithm for P_{raw} is shown in Algorithm 2, where compression projection matrix P_{raw} is optimized to minimize the objective as described in Equation (5).

There are several hyperparameters involved in the training of P_{raw}, including α in Equation (5) and σ_{raw} in the initialization of P_{raw}. We set $\alpha = 1$ and $\sigma_{\text{raw}} = 0.1$ in all experiments. The learning rate is set to 10 and we train 500 epochs for each image-text pair.

Algorithm 2 Training algorithm for MetaCompress.

Require: $(I_{\text{IMG}}, I_{\text{TXT}})$: the image-text pair; $E_{\text{TXT}}(\cdot)$: the language encoder; $V_{\text{IMG}}(\cdot)$: the image encoder; $\text{LLM}(\cdot,\cdot)$: the vision-language decoder; P_{raw}: the learnable compression matrix with shape $m \times n$.

Initialize P_{raw} with Gaussian distribution $\mathcal{N}(0, \sigma_{\text{raw}}^2)$.
$X_{\text{TXT}} \leftarrow E_{\text{TXT}}(I_{\text{TXT}})$
$X_{\text{IMG}} \leftarrow V_{\text{IMG}}(I_{\text{IMG}})$
$\boldsymbol{y} \leftarrow \text{LLM}(X_{\text{TXT}}, X_{\text{IMG}})$
while *not* converged **do**
  $\tilde{X}_{\text{IMG}} \leftarrow \sigma(P_{\text{raw}}) X_{\text{IMG}}$ # with gradients
  $\tilde{\boldsymbol{y}} \leftarrow \text{LLM}(X_{\text{TXT}}, \tilde{X}_{\text{IMG}})$ # with gradients
  Compute the final loss and gradient $\nabla_{P_{\text{raw}}}$ w.r.t. P_{raw}.
  Update P_{raw} with SGD optimizer.
end while

8.2. MetaCompress

To adapt to arbitrary compression rate, the stride k in the pooling operation is set to a float value $\frac{n}{m}$, which can be easily implemented by the fractional max pooling operation [22]. We set the kernel size $s = 3$ for all experiments.

9. Properties of MetaCompress

Now, we analyze the properties of MetaCompress when $W_q = W_k = W$ and are drawn from $\mathcal{N}(0, \sigma_w^2)$ in Equation (9).

We start by considering when kernel size $k = 1$, meaning that $\text{Pool}(X)$ is a down-sampling operation to X, and we let $\tilde{X}$ denotes the down-sampled image sequence. In this case, Equation (9) can be simplified as

$$P = \tilde{X} S X^\top, \tag{11}$$

where S is a positive semi-definite matrix. Therefore, the expectation $\mathbb{E}[p_{i,j}] = 0$ for all positions that $\tilde{x}_i \neq x_j$, and for $\tilde{x}_i = x_j = x$

$$\mathbb{E}[p_{i,j}] = \mathbb{E}[xSx^\top] = \mathbb{E}[xW\ \text{diag}(\omega) W^\top x^\top]. \tag{12}$$

Here, we notice that $y = xW$ is still a random vector, with all elements subject to $\mathcal{N}(0, d\sigma_c^2\sigma_w^2)$. Hence,

$$\mathbb{E}[y\ \text{diag}(\omega) y^\top] = d d_c \sigma_c^2 \sigma_w^2. \tag{13}$$

Considering that the embedding dimension of LVLMs is a large number (*e.g.*, 4096 for LLaVA-1.5-7b), $\sigma(P_{\text{raw}})$ is close to the down-sampling projection to the input image sequence controlled by the stride s.

Further, when the kernel size $k > 1$, the expectation of $p_{i,j}$ is still zero when $\tilde{x}_i$ is not captured by the pooling kernel located in x_i. To sum up, the initialization to Equation (9) converting MetaCompress to a interpretable pooling operation to the input image sequence.

However, as training progresses, W_q diverges from W_k, breaking the positive semi-definiteness of matrix S, enabling MetaCompress to further explore more effective compression strategies, ultimately enhancing performance. Besides, we choose the compression embedding dimension d_c to be smaller than the original embedding dimension d to reduce the computational cost and number of parameters.

Essentially, Equation (9) is a specialized form of the dot-product attention $XW_Q W_K^\top X^\top$, making it easier to optimize and less prone to over-fitting to the training dataset, as we only adopt a few-shot subset for training efficiency.

10. More Results

10.1. Performance of Fixed Compression Matrix

Because we train the compression matrix P_{raw} for a single image on the training dataset, which is a straightforward optimization problem, we do not compare it with other methods. Here, we present the overall accuracy about compressing LLaVA-Next-7b on the MT-VQA-v2 dataset for reference. The accuracy of the base setting is 82.44, and when reducing 90% of the image token the accuracy decreases to 80.89.

10.2. More comparison Results

Table 4 presents the comparison results of different token reduction method with the compression rate of 70%. Our method achieves the best overall performance, the same as the results in Table 1.

As a supplement, Table 5 compares the effectiveness of token reduction methods. Here, 'Spatial' represents applying spatial pooling to the image sequence (the kernel size k is set to the same as the stride s). 'ToMe' [5] is a token merging

Table 4. The comparison of visual token reduction methods on three MT-VQA benchmarks with the reduction rate of 70%. The best and the second-best results are highlighted in bold and underline, respectively.

Model	Method	MT-VQA-v2				MT-GQA				ConvBench			
		Acc_1	Acc_2	Acc_3	Avg	Acc_1	Acc_2	Acc_3	Avg	S_1	S_2	S_3	Avg
LLaVA-1.5-7b	Base	76.72	77.51	77.30	77.18	61.76	64.07	65.35	63.73	4.33	5.72	5.55	5.20
	Random	72.72	73.57	73.11	73.13	57.79	61.07	63.09	60.65	4.17	4.73	5.51	**4.80**
	Sample	72.96	73.71	73.33	73.33	58.48	61.12	62.30	60.63	4.16	5.03	3.81	4.33
	FastV	69.30	69.57	69.41	69.43	54.79	57.65	60.03	57.49	3.99	5.03	3.47	4.16
	PruMerge	72.79	73.90	73.42	73.37	57.89	59.54	61.81	59.75	3.64	4.68	4.68	4.33
	Ours	75.67	76.63	76.46	**76.25**	58.62	60.96	63.64	**61.07**	3.99	5.03	4.85	4.62
LLaVA-1.5-13b	Base	78.35	79.47	78.92	78.91	62.47	65.21	67.22	64.97	4.33	7.11	5.72	5.72
	Random	73.63	74.56	73.96	74.05	57.74	61.24	63.33	60.77	4.03	4.89	5.24	4.72
	Sample	73.81	74.79	74.51	74.37	58.51	61.29	62.82	60.87	3.64	5.03	5.37	4.68
	FastV	73.85	75.18	74.58	74.54	57.99	60.75	63.51	60.75	4.37	6.92	5.10	5.46
	PruMerge	73.80	75.16	74.57	74.51	57.67	60.45	62.18	60.10	4.51	6.41	5.37	5.43
	Ours	74.03	76.98	76.31	**75.77**	59.48	61.91	65.25	**62.21**	4.33	6.93	5.37	**5.55**
LLaVA-NeXT-7b	Base	80.20	80.86	80.71	80.59	63.83	66.68	67.94	66.15	7.95	11.46	7.58	9.00
	Random	76.18	77.43	77.64	77.08	61.96	63.95	66.29	64.07	7.97	9.19	6.93	8.03
	Sample	76.60	77.93	77.96	77.50	62.28	64.61	66.17	64.35	7.63	8.32	4.33	6.76
	FastV	75.96	76.86	76.39	76.40	61.54	64.37	65.97	63.96	0.00	0.00	2.50	0.83
	Ours	77.75	78.06	78.54	**78.12**	63.38	64.69	67.59	**65.22**	7.63	9.88	7.45	**8.32**
LLaVA-NeXT-13b	Base	81.02	82.32	81.64	81.66	65.45	67.32	69.12	67.30	12.48	13.17	7.97	11.21
	Random	77.30	78.77	78.65	78.24	62.89	64.64	67.22	64.92	11.44	11.27	8.15	10.29
	Sample	77.51	79.15	79.03	78.56	63.95	64.54	67.20	65.23	10.40	14.04	6.76	10.40
	FastV	75.78	77.16	76.66	76.53	62.10	64.37	65.06	63.84	20.00	5.00	3.33	9.44
	Ours	78.14	80.98	80.21	**79.78**	64.86	65.89	67.59	**66.11**	10.92	13.86	7.11	**10.63**
XComposer-2.5-7b	Base	78.80	81.12	81.24	80.39	60.21	63.38	65.01	62.87	12.48	7.00	7.97	9.15
	Random	74.63	77.23	77.44	76.43	56.96	61.59	63.63	60.73	15.42	10.23	7.97	11.21
	Sample	75.07	77.68	78.04	76.93	57.79	61.17	63.36	60.77	15.94	11.79	6.07	11.27
	FastV	77.93	80.18	80.05	79.39	58.95	61.34	62.67	60.99	12.50	4.17	12.50	9.72
	Ours	78.24	80.53	80.79	**79.85**	60.11	62.28	64.14	**62.18**	15.77	12.13	6.24	**11.38**

Table 5. Comparison results of different token merging methods for LLaVA-1.5-7b.

Setting	MT-GQA			
	Acc_1	Acc_2	Acc_3	Avg
Base	61.76	64.07	65.35	63.73
Sample	54.60	57.07	59.31	56.99
Spatial	51.05	54.62	56.24	53.97
ToMe	53.64	56.91	57.62	56.06
VisionZip	55.08	57.82	59.89	57.60
Ours	55.95	58.71	60.64	58.43

method proposed for ViTs rather than LVLMs, and thus performs ineffectively in our setting. 'VisionZip' [63] is a hybrid token compression method that integrates both token pruning and merging, yet it does not take MT-VQA scenarios into account and therefore also underperforms our method.

10.3. More Ablation Results

The results in Table 6 provide additional ablation studies across various LVLMs, further supporting the results in Table 3 and demonstrating that each objective contributes positively to the overall performance.

10.4. More Transfer Results

Table 7 reports more transfer validation experiments across LVLMs on MT-VQA-v2 and MT-GQA, showing that MetaCompress does not heavily depend on the specific training dataset, thereby demonstrating robust generalization ability. Furthermore, Table 8 reports transfer results on video question answering task. In detail, we transformed the video QA dataset Video-MME [21] into a 3-turn dialogue version, referred to as MT-Video-MME (including 500 dialogs for validation), and conducted comparative evaluations against baseline methods at a 70% compression rate. Since XComposer-2.5-7B natively supports video input, we directly use the

Table 6. Additional ablation study of training MetaCompress for various LVLMs using different loss terms on MT-GQA. Gradient clipping is only applied for the '$\mathcal{L}_{\text{collapse}}$ + *Grad Clip*' setting.

$\mathcal{L}_{\text{pred}}$	$\mathcal{L}_{\text{entropy}}$	$\mathcal{L}_{\text{collapse}}$	$\mathcal{L}_{\text{collapse}}$ + *Grad Clip*	**LLaVA-1.5-7b**	**LLaVA-NeXT-7b**	**XComposer-2.5-7b**
✓	✗	✗	✗	56.63	61.98	56.77
✓	✓	✗	✗	57.99	62.42	58.01
✓	✗	✓	✗	52.26	56.34	52.53
✓	✗	✗	✓	57.57	62.13	58.24
✓	✓	✗	✓	58.43	62.70	58.68

Table 7. Transfer validation experiments. Average accuracy is reported for cross-dataset transfer between MT-VQA-V2 and MT-GQA, all under a 90% token reduction rate.

Settings	**LLaVA-1.5-7b**	**LLaVA-1.5-13b**	**LLaVA-NeXT-7b**	**LLaVA-NeXT-13b**	**XComposer-2.5-7b**
MT-VQA-v2	70.65	72.94	75.18	75.26	75.76
MT-GQA → MT-VQA-v2	69.06	71.89	73.61	73.25	74.41
MT-GQA	58.43	59.48	62.70	63.12	58.68
MT-VQA-v2 → MT-GQA	57.45	58.78	61.43	62.60	58.06

Table 8. Transfer results on MT-Video-MME. Average accuracy is reported across different methods.

Metrics	**Base**	**Random**	**Sample**	**FastV**	**Ours**
Acc_1	44.3	26.8	25.5	26.2	28.5
Acc_2	46.7	25.3	26.4	27.3	27.3
Acc_3	48.2	30.7	31.3	31.6	34.6
Avg	46.4	27.6	27.7	28.4	30.1

pre-trained weights obtained from training on the small combined dataset of MT-VQA-v2 and MT-GQA (only around 20k samples in total as we mentioned) to evaluate on the MT-Video-MME benchmark. As shown in the table, MetaCompress outperforms baseline approaches even without any task-specific training on MT-Video-MME, further demonstrating its strong transferability.

11. More Visualizations

Figures 6 and 7 show the visualization of the generated compression projection for LLaVA-1.5-7b and LLaVA-1.5-13b on MT-GQA with a compression rate of 90% (we randomly select two images as examples). The row and column indices in the figures represent the original and reduced token indices, respectively, with darker colors indicating higher retention weights. As observed, MetaCompress performs pruning and merging operations at different positions, but is primarily based on equidistant down-sampling, with specific adaptations for certain tokens.

12. Discussions

Most data-driven approaches for efficient model inference primarily focus on model pruning [43, 52], efficient attention mechanisms [39, 50], and efficient model architectures [33, 41, 56], particularly in designing vision encoders for stronger and more compact visual representations. However, these methods typically require fine-tuning the entire model, resulting in substantial computational overhead. In contrast, our proposed MetaCompress trains only a small number of lightweight linear projection layers (D_q, D_k, and w), yet it surpasses existing token reduction approaches. Owing to its efficiency, MetaCompress requires only a modest amount of training data (approximately 20k samples) while exhibiting strong transferability across datasets, as demonstrated in our transfer experiments. This generalization capability stems from the fact that MetaCompress is trained to preserve as much general visual information as possible for multi-turn dialogues rather than being specialized for specific image domains.

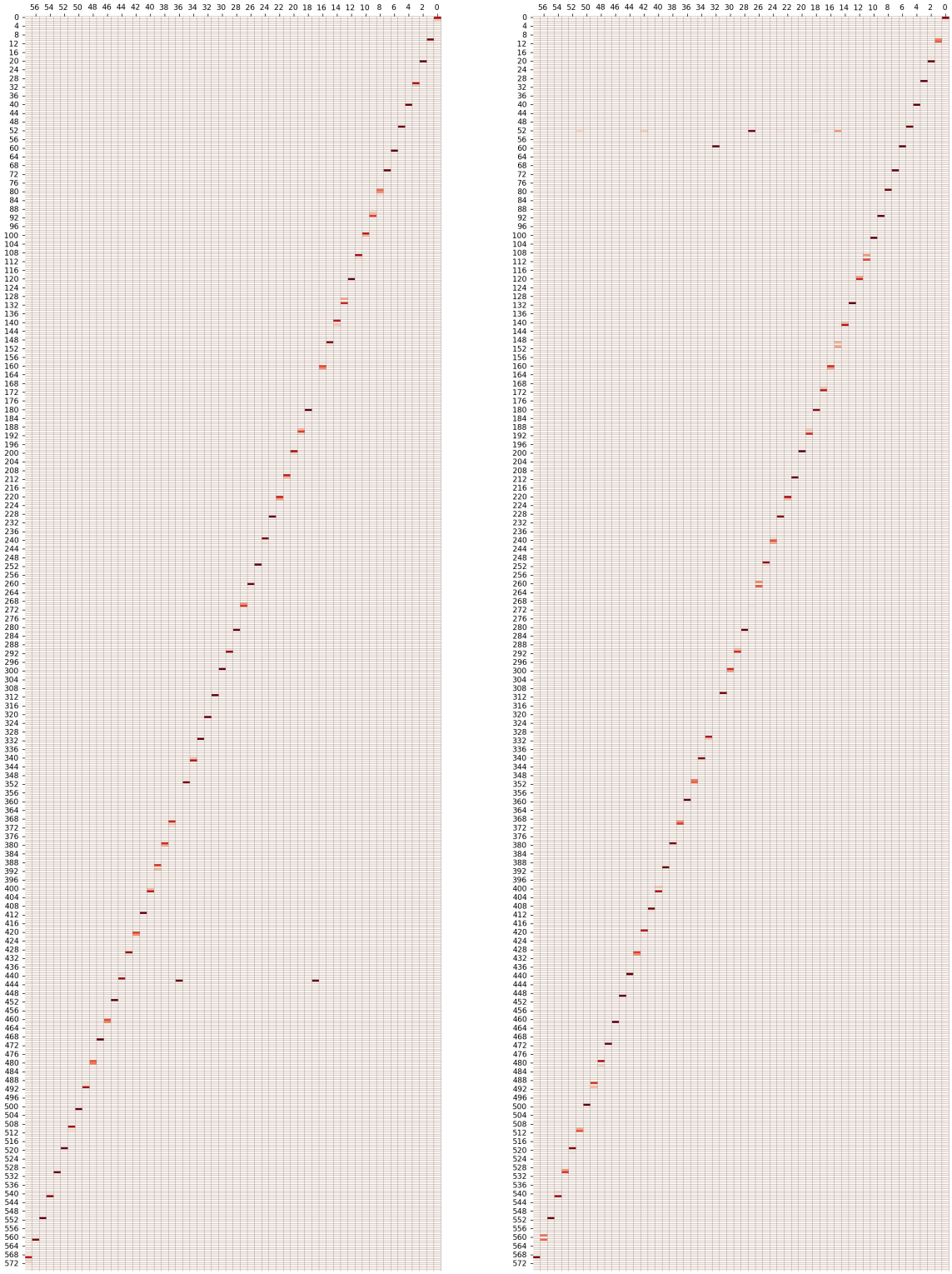

Figure 6. Visualization of the compression projection for LLaVA-1.5-7b on MT-GQA with the compression rate of 90%.

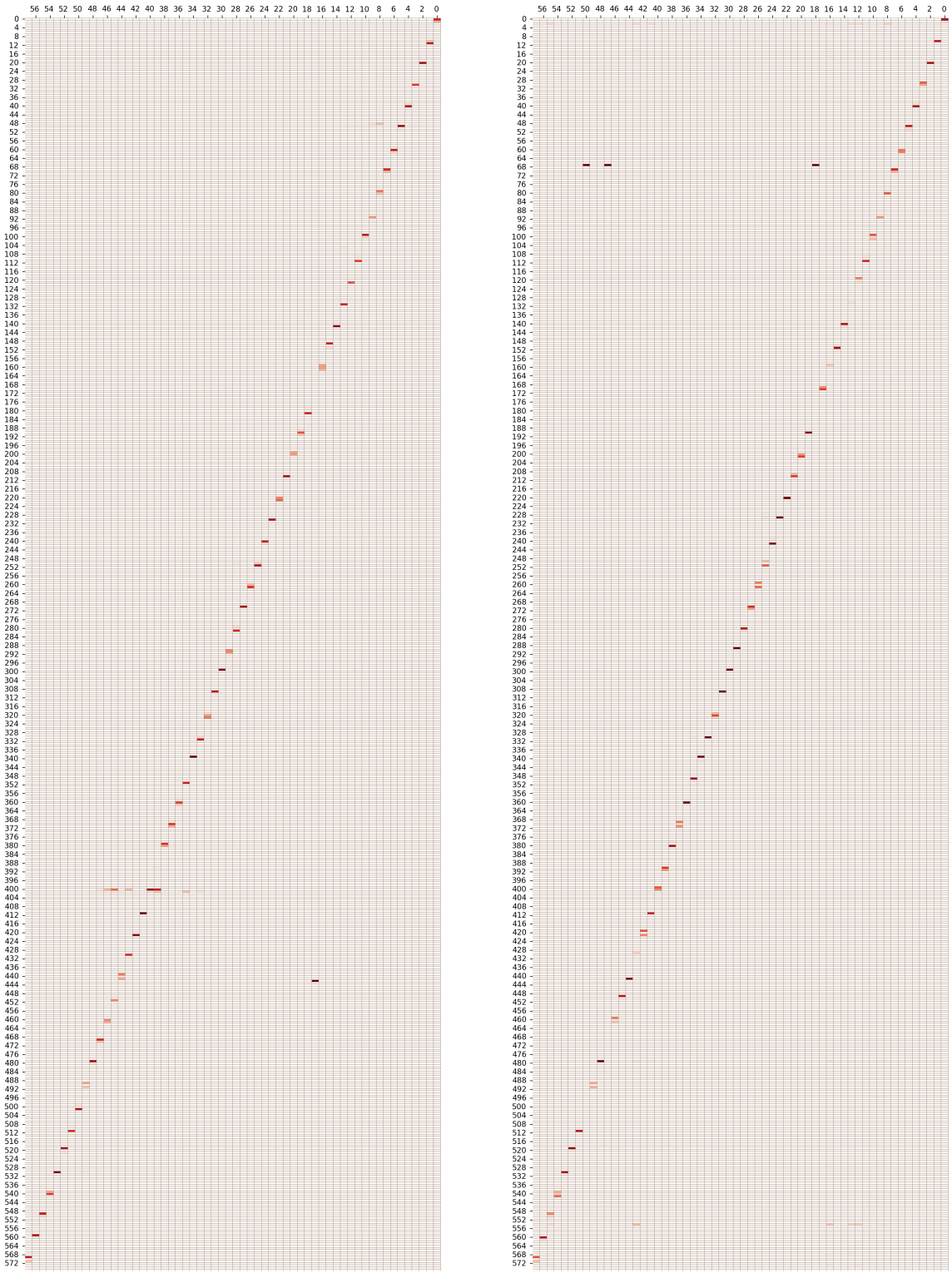

Figure 7. Visualization of the compression projection for LLaVA-1.5-13b on MT-GQA with the compression rate of 90%.