DeCafNet: Delegate and Conquer for Efficient Temporal Grounding in Long Videos

Supplementary Material

We present DeCafNet, an efficient algorithm that uses a *delegate-and-conquer* strategy to achieve accurate and efficient temporal grounding in long videos. In this supplementary material, we provide additional details about our architecture, experimental results, ablation studies, and implementation specifics.

1. Additional Architectural Details

To enable temporal grounding using features extracted by both the sidekick and expert encoders, we introduce DeCaf-Grounder. DeCaf-Grounder consists of the following key components: query-aware temporal aggregation, multi-scale temporal refinement, and classifier & regressor. In this section, we provide additional details about the multi-scale temporal refinement component.

Recall that, DeCaf-Grounder produces multi-scale features via query-aware temporal aggregation, $\{\mathbf{Z}^l\}_{l=0}^{L}$. The features capture temporal information from local to global scales, i.e., $\mathbf{Z}^0$ represents the most local scale, encoding one clip per feature, and $\mathbf{Z}^L$ is the most global scale, encoding 2^L clips per feature.

Since the features generated by the sidekick and expert encoders are at different temporal resolutions, this mismatch can result in inconsistencies in $\mathbf{Z}^l$ across varying scales. We aggregate information across scales to improve temporal grounding and focus on grounding-relevant information to maximize efficiency. Overall, multi-scale temporal refinement consists of four steps: **transform-expand-aggregate-pool**, as shown in Figure 1.

Transform. To explicitly capture grounding-specific information, we transform $\{\mathbf{Z}^l\}$ to $\{\mathbf{p}^l\}$ via a FFN classifier,

$$\mathbf{p}^l = \text{FFN}(\mathbf{Z}^l) \in \mathbb{R}^{T/2^l}. \tag{1}$$

where $\mathbf{p}^l$ has the same temporal length as $\mathbf{Z}^l$. It explicitly denotes if the ground truth moments happen at the temporal position represented by features in $\mathbf{Z}^l$. This also reduces the feature dimension to 1. The FFN classifier is trained via Focal Loss as explained in the main paper.

Expand. To combine $\{\mathbf{p}^l\}$, we need to first align their temporal lengths. We apply linear interpolation to expand each $\mathbf{p}^l$ to length T,

$$\hat{\mathbf{p}}^l = \text{linear-interpolate}(\mathbf{p}^l) \in \mathbb{R}^T. \tag{2}$$

All $\{\hat{\mathbf{p}}^l\}$ have the same temporal length T. Thus, we can concatenate them to obtain $\hat{\mathbf{P}} = \text{concat}[\hat{\mathbf{p}}^0, \ldots, \hat{\mathbf{p}}^T] \in \mathbb{R}^{T\times L}$.

Ψ_D	Ψ_E	TFLOPS	Mem (G)	Time (Sec)
100%	0%	64.8	40.1	1.9
0%	100%	2071.8	700.4	48.0
100%	30%	686.3 ↓ 67%	250.2 ↓ 64%	15.3 ↓ 68%
100%	50%	1100.7 ↓ 47%	390.3 ↓ 44%	24.3 ↓ 49%

Table 1. Average Encoder Computation measured on Ego4D-GoalStep [6] dataset. Column 1, 2 show the amount of clips processed by each encoder. With saliency selection (row 3, 4), DeCafNet significantly reduces TFLOPs by 47% and 67% compared to the feature-extraction cost in prior works that process all clips with expert encoder Ψ_E (row 2).

Aggregate. With $\hat{\mathbf{P}}$, we employ a temporal convolution to synchronize grounding information across scales,

$$\mathbf{H} = \text{convolution}(\hat{\mathbf{P}}) \in \mathbb{R}^{T\times C}, \tag{3}$$

where $\mathbf{H}$ is the output of temporal convolution, encoding refined grounding information. C is the size of feature dimension.

Pool. To combine $\mathbf{H}$ with the initial features $\{\mathbf{Z}^l\}$, we continue to compute a multi-scale feature pyramid from $\mathbf{H}$ via simple average pooling,

$$\mathbf{U}^l = \text{average-pooling}(\mathbf{H}) \in \mathbb{R}^{T/2^l\times C}, \tag{4}$$

where $\mathbf{U}^l$ is obtained by pooling $\mathbf{H}$ on temporal dimension by a factor of 2^l. Finally, we concatenate it with $\mathbf{Z}^l$ to obtain $\mathbf{Z}^l_{\text{refine}}$ as explained in the main paper.

2. Computation Efficiency on Ego4D-Goalstep

In Table 2 of the main paper, we have reported computation efficiency on Ego4D-NLQ dataset. In Table 1 of this supplementary material, we also show the computation on Ego4D-Goalstep dataset. Row 2 shows the feature extraction cost of all prior works that process all clips via expert encoder Ψ_E. Row 3 and 4 show the computation of our saliency selection method with the sidekick encoder Ψ_D. Since the computation cost is linear to the number of video clips, we similarly reduce TFLOPS by 67% and 47%, demonstrating our *delegate-and-conquer* approach has significantly lower computation cost than prior methods.

3. Additional Experimental Results

Table 2, 3 show complete model results on Ego4D-NLQ and Ego4D-Goalstep datasets. Their settings are consistent

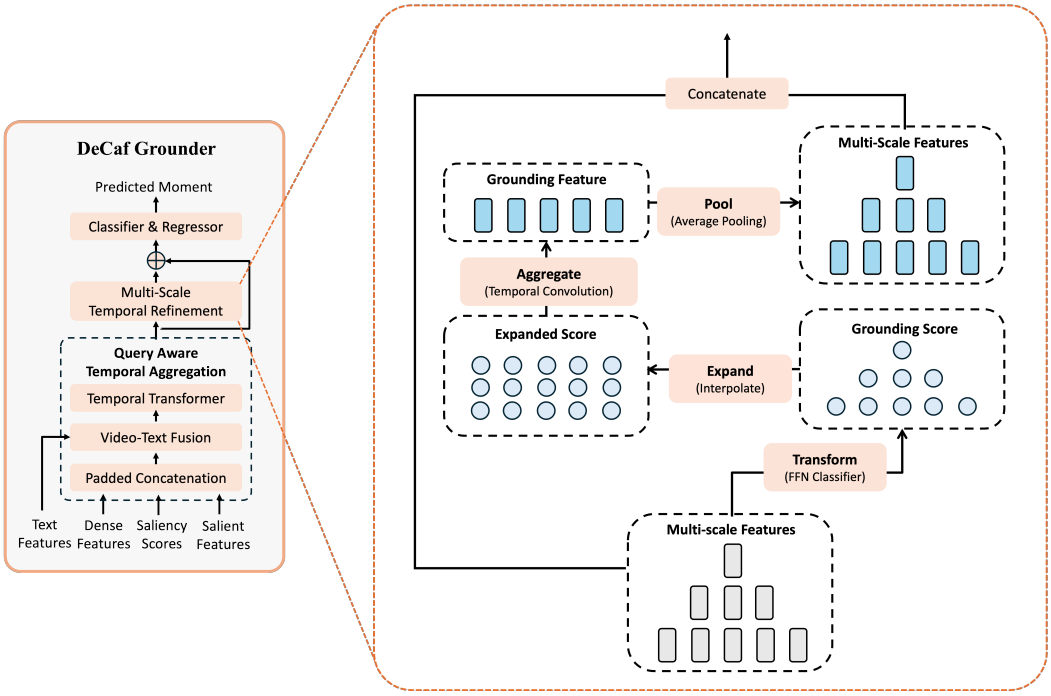

Figure 1. Details of multi-scale temporal refinement. The multi-scale features produced by the temporal transformer are transformed into grounding scores using an FFN classifier. To synchronize grounding information across different scales, we utilize linear interpolation and temporal convolution. Finally, average pooling is applied to effectively combine the synchronized features with the input features.

	R1@0.3	R1@0.5	R5@0.3	R5@0.5	AVG
RGNet [1]	18.28	12.04	34.02	22.89	21.81
SnAG [4]	15.87	11.26	38.26	27.16	23.14
DeCafNet-30%	18.07	12.41	37.68	27.47	23.91
DeCafNet-50%	**18.10**	**12.55**	**38.85**	**28.27**	**24.44**
DeCafNet-100%	**19.07**	**12.98**	**41.57**	**30.42**	**26.01**
RGNet[1] †	20.63	12.47	41.67	25.08	24.96
DeCafNet-30% †	**21.13**	15.04	**42.42**	31.22	27.45
DeCafNet-50% †	20.81	**15.04**	42.40	**31.68**	**27.48**
DeCafNet-100% †	**22.21**	**15.52**	**45.63**	**33.93**	**29.32**

Table 2. Complete Model Results on Ego4D-NLQ dataset. † denotes the models are pretrained on NaQ dataset [5].

	R1@0.3	R1@0.5	R5@0.3	R5@0.5	AVG
VSLNet [7]	11.70	-	-	-	-
SnAG [4]	18.34	15.12	45.95	38.55	29.49
RGNet [1]	21.26	15.71	47.15	37.85	30.49
DeCafNet-30%	20.01	16.22	44.70	37.34	29.56
DeCafNet-50%	**21.29**	**17.46**	**47.27**	**40.40**	**31.61**
DeCafNet-100%	**23.20**	**19.40**	**51.38**	**44.17**	**34.54**

Table 3. Complete Model Results on Ego4D-Goalstep dataset.

with those of Table 1, 3 in the main paper. We include the performance of DeCafNet-100% on both datasets, where we process all clips with both sidekick and expert encoders (rows in blue in Table 2 and Table 3). Compared to all prior methods that process all clips with the expert encoder, this model provides more diverse features to grounding models with *only 3% more TFLOPs* for running the sidekick encoder (row 1 vs row 2 in Table 1). It can be observed that, DeCafNet-100% greatly boosts the performance. In Table 2, it achieves an average recall of 26.01% on Ego4D-NLQ, higher than SnAG by 2.87%. In Table 3, it achieves an average recall of 34.54% on Ego4D-Goalstep, higher than SnAG by 4.05%.

Moreover, we also follow the setting in RGNet to pretrain models on the larger NaQ dataset [5], as shown in the second section of Table 2. First, we highlight that, our DeCafNet-50% without pretraining already achieves close performance to RGNet with pretraining, while using 47% less computations. After pretraining, DeCafNet outperforms RGNet by large margins and improves average recall by 2.49% to 4.36%. Pretraining also enhances our accuracy on saliency selection, therefore DeCafNet-30% now has similar performance as DeCafNet-50%.

4. Implementation Details

Our sidekick encoder has 12 spatio-temporal blocks and we initialize its weight from [2] to speed up training. For temporal convolution [3] in multi-scale temporal refinement,

we use 8 layers, where the dilation rate of the i-th convolution layer equals to 2^i. Since neither SnAG nor RGNet reports performance on the Ego4D-Goalstep dataset, we use their released codes to report performance on this dataset. We measure all computation cost using one 80GB A100 GPU. When the GPU cannot store all video clips in memory, we split the data into multiple batches and report the overall TFLOPS/Mem/Time summed over all batches. To evaluate on short temporal video grounding datasets, we use features released by SnAG and use the I3D feature for Charades-STA dataset.

5. Limitations

DeCafNet has established new SOTA for LVTG with greatly reduced computation. However, the overall recall values are relatively low, especially for R1@0.3 and R1@0.5. We found this is partly caused by ambiguity in text queries in the dataset. For example, for a text query of "Where was object X before I used it?", the object was often used for multiple times by the person. While the model can identify most of the temporal regions involving the object, it is often unclear about which region is the correct moment and gives them similar confidence. This leads to low R1@0.3 and R1@0.5, whereas R5@0.3, R5@0.5 are often much higher. The above mentioned ambiguity can potentially be mitigated by clarifying text queries, such as specifying, "Where was object X before I used it for the first time?".

References

[1] Tanveer Hannan, Md Mohaiminul Islam, and Thomas Seidl. Rgnet: A unified clip retrieval and grounding network for long videos. In *European Conference on Computer Vision*, 2024. 2

[2] Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *arXiv preprint arXiv:2206.01670*, 2022. 2

[3] Z. Lu and E. Elhamifar. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 2

[4] Fangzhou Mu, Sicheng Mo, and Yin Li. Snag: Scalable and accurate video grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18930–18940, 2024. 2

[5] Santhosh K. Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Naq: Leveraging narrations as queries to supervise episodic memory. In *Computer Vision and Pattern Recognition (CVPR), 2023 IEEE Conference on*. IEEE, 2023. 2

[6] Yale Song, Eugene Byrne, Tushar Nagarajan, Huiyu Wang, Miguel Martin, and Lorenzo Torresani. Ego4d goal-step: Toward hierarchical understanding of procedural activities. In *Advances in Neural Information Processing Systems*, 2023. 1

[7] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. *arXiv preprint arXiv:2004.13931*, 2020. 2