Language-free Training for Zero-shot Video Grounding -Supplementary Material-

Dahye Kim[1] Jungin Park[1] Jiyoung Lee[2] Seongheon Park[1] Kwanghoon Sohn[1,3*]
[1]Yonsei University [2]NAVER AI Lab [3]Korea Institute of Science and Technology (KIST)
{dadaday, newrun, sam121796, khsohn}@yonsei.ac.kr lee.j@navercorp.com

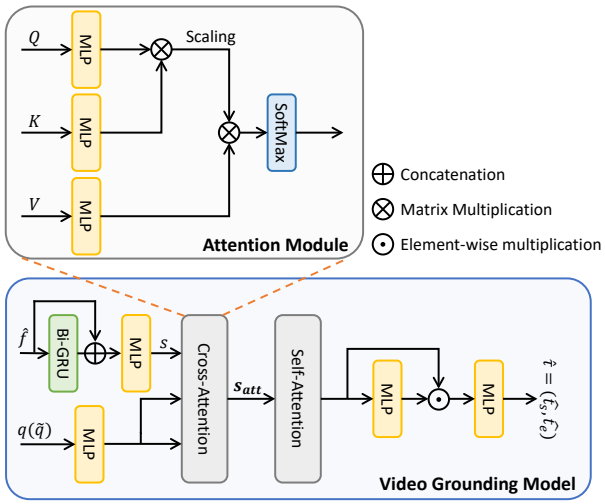


Figure 1. Overall architecture of our video grounding model.

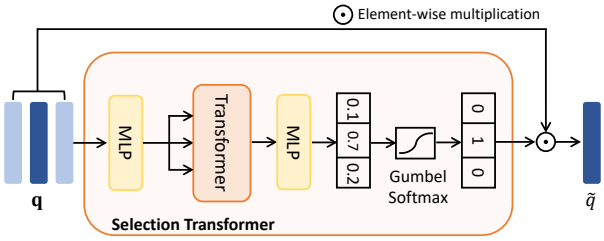


Figure 2. Selection transformer architecture.

1. Implementation Details

Video grounding model. We illustrate the architecture of our video grounding model in Fig. 1. The grounding model takes the video features $\hat{f}$ and the pseudo language feature $\tilde{q}$ while training and the real language feature q at inference. For the video features, we apply a Bi-GRU [1] and concatenate with $\hat{f}$ followed by a single MLP to obtain the encoded video features $s \in \mathbb{R}^{T\times d}$. We also apply a single MLP to project the language feature (q or $\tilde{q}$) to the d-dimensional feature space. With the projected video and language features, we obtain the language-guided video feature s_{att} using the cross-modal attention module. To take the global context into account, we employ the self-attention module for s_{att}. We use 3 multi-head attention (MHA) layers with 4 heads for the cross-attention and 2 MHA with 4 heads for the self-attention. Finally, we regress the start and end time stamps $(\hat{t}_s, \hat{t}_e)$ with two MLP layers:

$$a = \text{Softmax}(\texttt{MLP}_1(\hat{s}_{att})) \in \mathbb{R}^T, \quad (1)$$

$$\tilde{s} = \text{Avg}(a \odot \hat{s}_{att}), \quad (2)$$

$$(\hat{t}_s, \hat{t}_e) = \texttt{MLP}_2(\tilde{s}), \quad (3)$$

where $\hat{s}_{att}$ is the self-attended feature, $\text{Avg}(\cdot)$ is the average pooling along the temporal axis, and $\odot$ denotes the element-wise multiplication.

Selection transformer. We introduce the selection transformer to choose a single dominant pseudo language feature $\tilde{q}$ from the candidates $\mathbf{q}$. We employ a low-capacity transformer [8] to consider computational costs. Specifically, we first project the candidate features $\mathbf{q}$ using a single MLP and feed them into the transformer. The output of the transformer is fed into the additional MLP followed by Gumbel-softmax estimator [3] to obtain discrete logits. We finally take the single pseudo language feature q by multiplying the discrete logits to the initial candidates $\mathbf{q}$. The overall architecture of the selection transformer is shown in Fig. 2.

2. Additional Experimental Results

Effect of the number of frame embeddings. To verify the effectiveness of the number of pseudo language candidates N used in the selection transformer, we provide the performance according to various values of mIoU, as shown in Tab. 1. The results show that the performance increases

*Corresponding author

N	R@0.3	R@0.5	R@0.7	mIoU
1	50.2	34.84	15.66	33.49
2	51.21	34.07	17.81	34.48
4	51.03	35.74	17.65	34.37
9	52.95	37.24	19.33	36.05
16	50.92	34.87	16.33	33.80

Table 1. The ablation study of the number of the pseudo language candidates N in the selection transformer.

Frame Order	R@0.3	R@0.5	R@0.7	mIoU
Random	52.95	37.24	19.33	36.05
Ordered + e_{pos}	50.14	34.05	15.47	33.29

Table 2. The ablation study of the relative temporal order of the pseudo language candidates in the selection transformer.

as N increases in terms of Recall@0.7 and mIoU until the 9 candidates. We observe that a large number of candidates ($N = 16$) significantly degrades the performance. Not surprisingly, it seems that many candidates provide a large number of noisy and redundant candidates, making the selection transformer challenging to choose a good pseudo language feature. Especially, this problem can be maximized with the short temporal proposal which contains a small number of frames. Summarizing the results, we set $N = 9$ for all experimental results in the main paper.

Effect of temporal ordering. In our method, we generate the pseudo language feature from frames without any temporal context. We investigate the effect of the temporal ordering of frames in generating the pseudo language feature. We add temporal positional embeddings [8], e_{pos}, to the frame embeddings and feed them into the selection transformer. As shown in Tab. 2, the pseudo language features generated with temporal context slightly degrade the performance. We ascribe this mainly to the pretrained CLIP encoder featuring the image-based model that is not learned with temporal context.

Comparison with VideoCLIP. As shown in the main paper, the pretrained visual encoder of the image-based vision-language model (*i.e*. CLIP [7]) provides better performance than the video-language model (*i.e*. VideoCLIP [10]). We present additional qualitative comparisons to validate the effectiveness of the image-based vision-language model. As shown in Fig. 3, while the model with VideoCLIP sometimes fails to predict the time interval corresponding to given queries, the proposed model with CLIP predicts more confident time intervals. It demonstrates that the large-scale image-language model is effective enough when given a query describes static scenes.

Method	Sup.	TE (ms)	CML (ms)	ALL (ms)	R@0.5
LGI [5]	FS	1.63	7.75	9.38	59.46
WSTAN [9]	WS	1.4	43.14	44.55	29.35
CNM [11]	WS	0.91	8.7	9.61	35.43
Ours	ZS	6.73	7.65	14.38	37.24

Table 3. Speed and accuracy on the Charades-STA. Reported time costs include - TE (query embedding generation), CML (cross-modal learning for video grounding), and ALL (TE+CML).

Comparison with SOTA [6]. We present more qualitative comparisons with the SOTA method [6] evaluated on the Charades-STA [2] dataset. As shown in Fig. 4, our model produces more accurate predictions than PSVL and successfully predicts different time intervals according to the different queries for the same video. In addition, we depict qualitative results for the ActivityNet Captions [4] dataset as shown in Fig. 5.

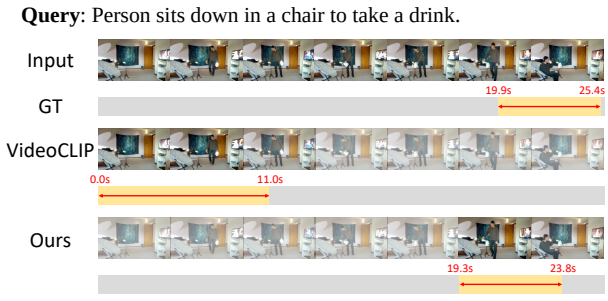
Computational Cost We measure inference time (ms) for each video in the Charades-STA dataset using a single RTX 2080ti with 11GB memory and report in Tab. 3. The inference time is separated into two parts: text embedding generation (TE); and crossmodal learning (CML) for video grounding. As shown in Tab. 3, our method requires a higher time cost for text embedding than previous works since we use CLIP text encoder. Meanwhile, our method achieves the fastest runtime by 7.65ms for cross-modal learning thanks to the lightweight model architecture. Our method requires 14.38ms in total, outperforming weakly-supervised approaches [9, 11].

Failure cases. We analyze some failure cases of our model as shown in Fig. 6. We observe that our model fails to localize the time stamps for the query where visual cues are hard to recognize or minor scene changes appear on consecutive frames. For example, while the action 'eat' is precisely detected, our model fails to recognize the object 'jar' (recognizes 'cup' instead), as shown in Fig. 6(a). As another example, our model produces a reasonable location for 'a person awakens in a bedroom' that has obvious scene changes, as shown in Fig. 6(b). However, the result shows that our method fails to localize the time interval for the query 'person looks over at picture' that corresponds to a static scene.

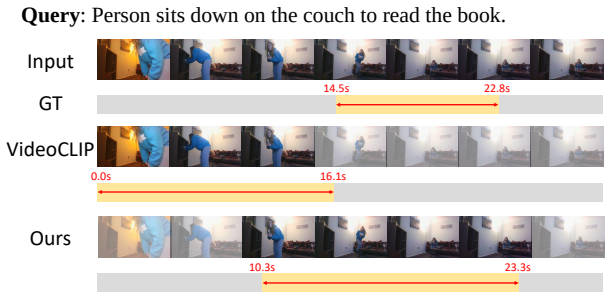
References

[1] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[2] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. *ICCV*, 2017.



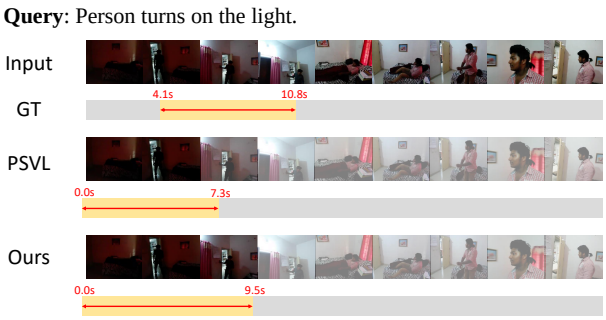
(a)



(b)

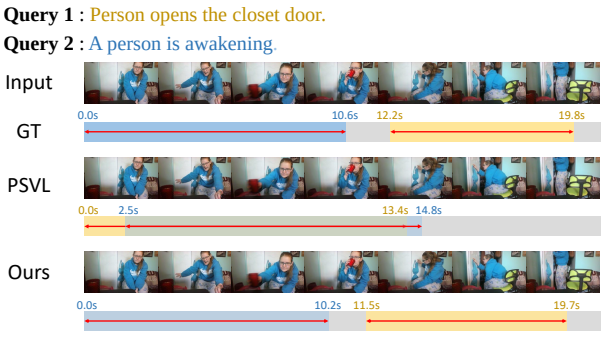
(c)

Figure 3. Qualitative comparisons corresponding to the language feature encoders on the Charades-STA dataset.



(a)

(b)



(c)

Figure 4. Qualitative comparisons between ours and PSVL on the Charades-STA dataset.

[3] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *ICLR*, 2017.

[4] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. *ICCV*, 2017.

[5] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. *CVPR*, 2020.

[6] Jinwoo Nam, Daechul Ahn, Dongyeop Kang, Seong Jong Ha, and Jonghyun Choi. Zero-shot natural language video localization. *ICCV*, 2021.

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *ICML*, 2021.

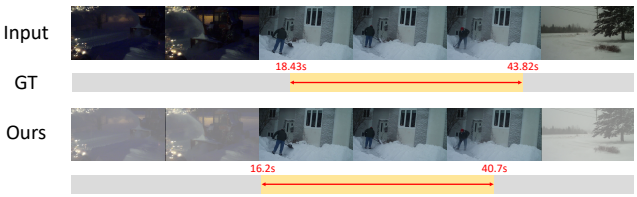
[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.

[9] Yuechen Wang, Jiajun Deng, Wengang Zhou, and Houqiang Li. Weakly supervised temporal adjacent network for language grounding. *IEEE TMM*, 2021.

[10] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *EMNLP*, 2021.

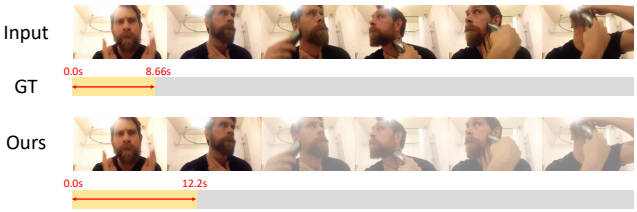
[11] Minghang Zheng, Yanjie Huang, Qingchao Chen, and Yang Liu. Weakly supervised video moment localization with contrastive negative sample mining. *AAAI*, 2022.

Query: A man is shoveling his walk way to his home.



(a)

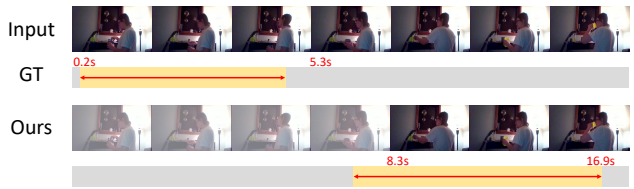
Query: A man is playing with his beard.



(b)

Figure 5. Qualitative comparisons between ground-truth intervals and ours on the ActivityNet Captions dataset.

Query: A person eats something from a jar.



(a)

Query 1: Person looks over at a picture.
Query 2: A person awakens in a bedroom.

(b)

Figure 6. Failure cases of ours comparing PSVL on the Charades-STA dataset.