

# Supplementary Material: Hierarchical Debiasing and Noisy Correction for Cross-domain Video Tube Retrieval

ANONYMOUS AUTHORS

In the supplementary material, we begin with a comprehensive description of our base video tube retrieval model in Section 1. We then detail the loss functions in Section 2 and implementation specifics in Section 3. In Section 4, we conduct granular ablation studies. Section 5 visualizes the processing of pseudo labels. Lastly, we analyze the limitations of our approach in Section 6.

## 1 MORE MODEL DETAILS

Our base video tube retrieval model STCAT [3] is built upon four essential components: the cross-modal transformer encoder (Section 1.1), the template generator (Section 1.2), the cross-modal transformer decoder (Section 1.3), and two parallel prediction heads (Section 1.4).

### 1.1 Cross-modal Encoder

The encoder harnesses cross-modal interactions between video and text, intricately correlating their corresponding semantics to achieve precise and consistent feature alignment across the two modalities. Given the visual features  $F_v$  and textual features  $F_s$ , both are first passed through a projection layer to embed them into a uniform channel dimension  $C$ . The projected visual embeddings are denoted as  $p_v = \{p_{vt}\}_{t=1}^T$ , where each  $p_{vt} \in R^{N_v \times C}$ , and the textual embeddings as  $p_s \in R^{N_s \times C}$ . The encoder comprises  $M$  stacked blocks, each integrating a spatial interaction layer (Section 1.1.1) and a temporal interaction layer (Section 1.1.2) that both utilize the transformer encoder structure [10].

**1.1.1 Spatial Interaction Layer.** To address local context within individual frames,  $T$  learnable tokens,  $p_l = \{p_l^t \in R^C\}_{t=1}^T$ , are introduced, where token  $p_l^t$  corresponds to frame  $t$ . The joint input sequence  $x_t$  for frame  $t$  is processed by a spatial interaction layer to generate contextualized visual-textual representations:

$$x_t = [p_l^t, p_{vt}^1, p_{vt}^2, \dots, p_{vt}^{N_v}, p_s^1, p_s^2, \dots, p_s^{N_s}]. \quad (1)$$

**1.1.2 Temporal Interaction Layer.** To integrate the global context of the video throughout the encoding process, we employ another learnable token,  $p_g \in R^C$ . The temporal interaction Layer models interactions across the frames using the input sequence  $x_g$ , which includes:

$$x_g = [p_g, p_l^1, p_l^2, \dots, p_l^T]. \quad (2)$$

To ensure the sequence maintains its temporal relevance, positional encoding is added to  $x_g$ .

The spatial interaction layer exclusively models the local context at the frame level, focusing on spatial intra- and inter-modality relationships within each frame. In contrast, the temporal interaction layer allows the local frame tokens to engage with the entire video content, supported by a global token that aggregates the overall video-text context. After processing, the encoder outputs contextualized multi-modal features,  $F_{vl} \in R^{T \times (N_v + N_s) \times C}$ , along with a global embedding,  $p_g$ , representing the entire video, and individual local embeddings,  $p_l = \{p_l^t\}_{t=1}^T$ , for each video frame.

---

Author's address: Anonymous Authors.

## 1.2 Template Generation

The template is composed of a content term (Section 1.2.1) and a position term (Section 1.2.2), both generated by a template generator that utilizes the encoded local and global tokens. The content term is shared by all frames, while the position term is characterized by per frame.

**1.2.1 Content Term.** The content term  $q_c \in R^C$  is generated through the global visual-linguistic context:

$$q_c = W_c p_g + b_c, \quad (3)$$

where  $W_c$  and  $b_c$  are both learnable parameters, strategically designed to optimize semantic representation.

**1.2.2 Position Term.** The position term  $q_p = \{q_p^t\}_{t=1}^T$  is defined for each frame  $v_t$  where each  $q_p^t$  is a 4-dimensional vector  $(x_t, y_t, w_t, h_t)$ , serving as a reference anchor to the grounding region in that frame. Drawing from [13], we first modulate the local embeddings  $p_l = \{p_l^t\}_{t=1}^T$  using the token  $p_g$ :

$$\gamma^c = \tanh(W_\gamma p_g + b_\gamma), \quad \beta^c = \tanh(W_\beta p_g + b_\beta), \quad (4)$$

where  $W_\gamma, b_\gamma, W_\beta$ , and  $b_\beta$  are learnable parameters. Then the position term  $q_p^t$  is computed by modulating  $p_l^t$ :

$$q_p^t = \text{Sigmoid}(f_p(\gamma^c \circ p_l^t + \beta^c)), \quad (5)$$

where  $f_p$  is a learnable mapping function from  $R^C$  to  $R^4$ .

## 1.3 Cross-modal Decoder

Given the templates  $\{q_c^t, q_p^t\}_{t=1}^T$  from the template generator, the object query for each frame  $v_t$  is structured as  $Q_t = [C_t; P_t]$ , incorporating both a content query  $C_t$  and a position query  $P_t$ . At each decoder block,  $C_t$  and  $P_t$  are initially generated from  $q_c^t$  and  $q_p^t$  based on:

$$C_t = q_c^t, \quad P_t = \text{Linear}(\text{PE}(q_p^t)), \quad (6)$$

where  $PE$  denotes the sinusoidal position encoding applied to the position template  $q_p^t = (x_t, y_t, w_t, h_t)$ . The content query  $C_t$  is then processed through both self-attention and cross-attention layers, while the position query  $P_t$  acts as the position encoding. As per recent advancements in decoder design [5, 11, 15], the position term  $q_p$  and subsequently  $P_t$  are updated layer by layer through a shared prediction head. The refined  $C_t$  and  $P_t$  are finally input into a prediction head to compute the final object tube.

To separate spatial and temporal feature aggregation, we employ a dual-decoder architecture, each focusing on bounding-box prediction and temporal boundary estimation, respectively. Adhering to the DETR [1], each decoder comprises  $M$  stacked blocks. Each block has a self-attention layer (Section 1.3.1) for modeling the temporal interactions across the entire video and a cross-attention layer (Section 1.3.2) for probing the encoded multi-modal feature within the corresponding frame.

**1.3.1 Self-Attention Layer.** In the self-attention layer, each content query  $C_t$  for frame  $v_t$  aggregates the global temporal context by attending to all other frames. We also add a sinusoidal time encoding to each content query  $C_t$  to reflect the original temporal positions of different  $C_t$ .

**1.3.2 Cross-Attention Layer.** The cross-attention layer aggregates the encoded cross-modal representation  $F_{vl}^t \in R^{(N_v+N_s) \times C}$  for each frame  $v_t$  to enrich the content query  $C_t$ .  $C_t$  cross-attends to  $F_{vl}^t$  using  $P_t$  as a positional anchor to

guide the decoder focus on the region most likely containing the target object. The output from the cross-attention layer is then processed through a shared prediction head, which generates the relative position adjustments  $(\Delta x_t, \Delta y_t, \Delta w_t, \Delta h_t)$  to update  $q_p^t$ .

#### 1.4 Prediction Head

After the decoding process, the refined content query  $C_t$  is used to predict both spatial and temporal localizations. The prediction head is divided into two branches: the bounding box branch, which predicts a 4-dimensional coordinate offset  $(\Delta x_t, \Delta y_t, \Delta w_t, \Delta h_t)$ , and the temporal branch, which estimates the start  $p_t^s$  and end  $p_t^e$  probabilities for each frame. The final bounding box for frame  $v_t$  is computed as:

$$b_t = (x_t + \Delta x_t, y_t + \Delta y_t, w_t + \Delta w_t, h_t + \Delta h_t). \quad (7)$$

## 2 MORE LOSS DETAILS

### 2.1 Supervised Loss

Each video-query pair has a ground-truth bounding box sequence  $B = \{b_t\}_{t_s}^{t_e}$ , and the corresponding start and end timestamps  $\{t_s, t_e\}$ . For spatial localization, we employ the smooth L1 loss, denoted as  $L_1$ , along with the generalized IoU loss [7], represented by  $L_{\text{IoU}}$ , with both being exclusively applied to the bounding boxes within  $\{t_s, t_e\}$ .

As for temporal localization, following [8], we take  $t_s$  and  $t_e$  to formulate two target categorical distribution vectors:  $\tau_s \sim N(t_s, 1) \in R^n$  and  $\tau_e \sim N(t_e, 1) \in R^n$ , where  $N(\mu, \sigma)$  represents a quantized Gaussian distribution centered at  $\mu$  with standard deviation  $\sigma$ , discretizing the Gaussian distribution across the range  $[1, n]$ . We use the Kullback-Leibler divergence loss  $L_{\text{KL}}$  to measure the distance between our predictions  $\{\hat{\tau}^s, \hat{\tau}^e\}$  and true probability distributions  $\{\tau^s, \tau^e\}$ .

$$L_{\text{KL}} = D_{\text{KL}}(\hat{\tau}^s || \tau^s) + D_{\text{KL}}(\hat{\tau}^e || \tau^e). \quad (8)$$

$D_{\text{KL}}$  symbolizes the Kullback-Leibler divergence. Moreover, we apply the guided attention loss  $L_{\text{att}}(A)$  to encourage weights for time queries outside the temporal boundaries to be lower than those inside the boundaries [8]:

$$L_{\text{att}} = - \sum_{i=1}^n (1 - \delta_{t_s \leq i < t_e}) \log(1 - a_i) \quad (9)$$

where  $\delta$  is the Kronecker delta and  $a_i$  is the  $i$ th column of the attention matrix  $A$ . We also predict whether a frame falls within the ground-truth temporal interval, supervised by a binary cross-entropy loss  $L_{\text{act}}$ .

The supervised loss  $L_{\text{sup}}$  is a linear combination of previously introduced five losses:

$$L_{\text{sup}} = \lambda_1 L_1 + \lambda_{\text{IoU}} L_{\text{IoU}} + \lambda_{\text{KL}} L_{\text{KL}} + \lambda_{\text{att}} L_{\text{att}} + \lambda_{\text{act}} L_{\text{act}}. \quad (10)$$

### 2.2 Unsupervised Loss

The unsupervised loss function  $L_{\text{unsup}}$  replaces the ground truth in Equation (10) with pseudo labels, which can be either hard or soft. Specifically, for temporally extended bounding boxes, hard pseudo labels (definitive single-value predictions) are used to calculate the  $L_1$  and  $L_{\text{IoU}}$  losses. In contrast, for the temporal start and end points, soft pseudo labels are employed. These labels, which represent continuous distributions, are derived from the predictions of the teacher model and are utilized to compute the  $L_{\text{KL}}$  loss in conjunction with corresponding predictions from the student model. Meanwhile, the  $L_{\text{att}}$  loss computation remains unchanged. For the  $L_{\text{act}}$  loss, soft pseudo labels are similarly used, leveraging the probabilistic distributions.

Table 1. Ablation studies on the I2O-VTR dataset.

GBA	TEX	Declarative Sentences				Interrogative Sentences			
		m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5	m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
		39.23	18.01	26.91	13.95	39.32	14.83	21.25	9.34
✓		40.15	20.26	31.33	16.97	40.30	16.57	24.70	11.36
	✓	40.34	20.68	32.23	17.59	40.52	16.93	25.37	11.71
✓	✓	41.33	23.22	36.80	20.81	41.63	18.91	28.85	13.83

### 3 MORE IMPLEMENTATION DETAILS

Consistent with prior methods [3, 12], we use ResNet-101 [2] as the visual encoder and RoBERTa [6] as the textual encoder. Specifically, we extract visual features from the output of the fourth residual block of ResNet-101. In both the cross-modal encoder and decoder, the number of attention heads is set to 8, and the hidden dimension of the feed-forward networks in the attention layers is 2,048. The prediction heads, including the bounding box head, temporal boundary head, and actionness score head, all employ the MLP architecture. The visual discriminator utilizes a CNN architecture, while the textual, frame-level, and video-level discriminators employ the MLP architecture. We initialize part of the model parameters with pre-trained weights from [4], and the entire framework is optimized end-to-end during the training process.

### 4 MORE ABLATION STUDIES

Our model consists of two modules: Layered Feature Debiasing (LFD) and Pseudo Label Refinement (PLR). Among these, the Adversarial Feature Alignment (AFA) in LFD and the Threshold Filtering (TFT) in PLR have been validated by a substantial body of work related to domain adaptation. We have specifically adapted these two sub-modules as our foundational components, thus eliminating the need for their ablation. In subsequent ablation experiments, these two sub-modules will be retained. It is noteworthy that through our strategic modifications, these modules alone have achieved significant performance improvements. The remaining sub-modules, Graph Based Alignment (GBA) and Temporal Expansion (TEX), are ingeniously designed to cater to the unique characteristics of the UDA-VTR scenario. However, their effectiveness in domain adaptation has not yet gained recognition. We undertake a series of experiments to test various combinations of these modules. Through extensive experimentation, we have confirmed the effectiveness of each module, both individually and in conjunction, thereby solidifying their utility in enhancing the domain adaptation capabilities of our approach.

The experimental results for the I2O-VTR and R2M-VTR datasets are detailed in Table 1 and Table 2, respectively. Integration of either of our two key sub-modules, Graph Based Alignment (GBA) and Temporal Extension (TEX), consistently enhances the model performance, affirming their critical role and operational efficacy. Furthermore, when these modules are employed together, they exhibit a synergistic effect that surpasses the performance achieved when either is used in isolation, suggesting the modules interact in a way that addresses different facets of the domain shift challenge. Additionally, the substantial performance improvements observed across diverse domain pairings, such as indoor-to-outdoor and real-to-movie, not only underline the versatility of each module but also underscore their robust generalization capabilities.

Table 2. Ablation studies on the R2M-VTR dataset.

GBA	TEX	m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
		33.94	18.05	25.56	11.24
✓		34.78	21.37	33.09	15.94
	✓	34.96	21.97	34.64	16.90
✓	✓	35.91	25.72	42.45	21.78

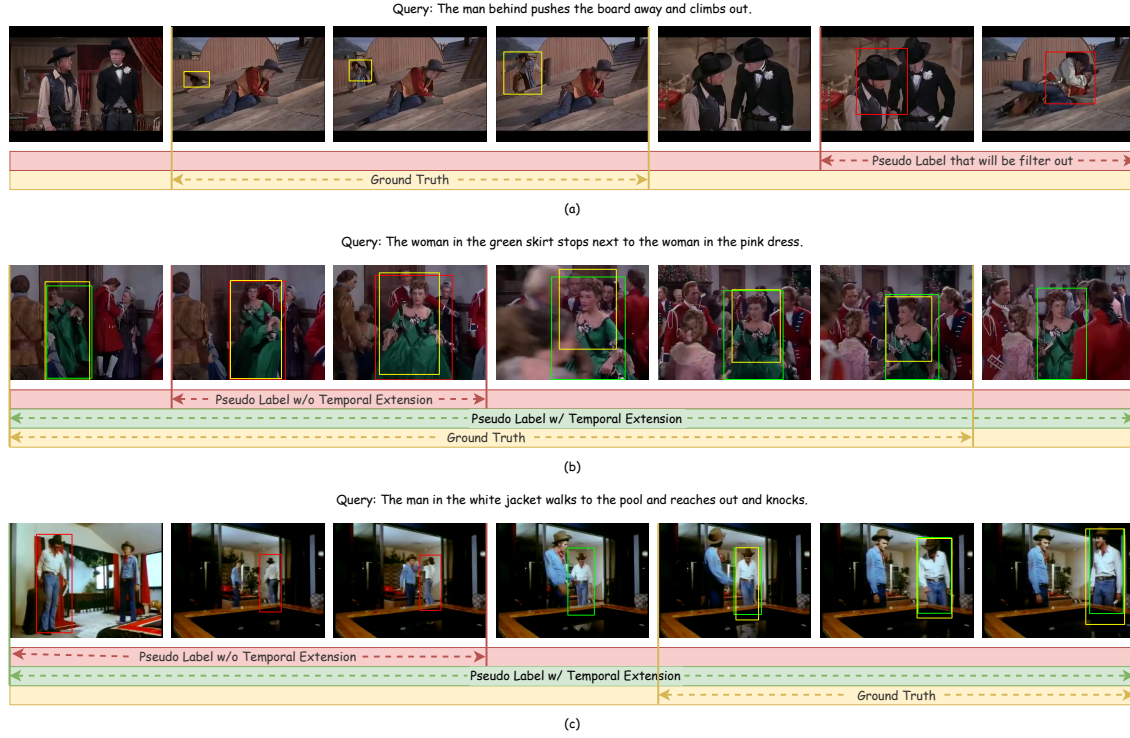


Fig. 1. Examples of pseudo label visualizations from the R2M-VTR dataset, illustrating both the pseudo labels that are discarded (a) and those that are temporally extended post-filtering (b)(c).

## 5 MORE CASE STUDIES

We conducted case studies on randomly selected samples from the target domain dataset, visualizing pseudo label manipulations in the R2M-VTR dataset [9, 14] as depicted in Figure 1. Through these studies, notable observations were made: In Figure 1(a), the model struggles to discern the core semantic content of the video that corresponds to the text query, resulting in bounding boxes that erratically alternate among several individuals and display a diminished average confidence level across the designated time span. Such erroneous predictions are eliminated by our filtering mechanism and are not included in the unsupervised loss computation for that iteration. Conversely, while the model aptly identifies spatial elements in other cases, its comprehension of temporal dynamics remains subpar. This misalignment results in bounding boxes with higher confidence levels but inaccurate temporal intervals, either too brief or markedly offset, as

seen in Figure 1(b), Figure 1(c). In these cases, our method extends the duration of the pseudo labels, enabling a greater number of bounding boxes to engage in the unsupervised loss computation. Such temporal extension typically aligns more closely with the query, facilitating more accurate tube retrieval by the model.

## 6 LIMITATIONS

We aim to elucidate the limitations of our proposed methodology. The incorporation of a teacher-student framework leads to an increase in computational demand, which in turn necessitates a larger allocation of GPU memory. This requirement restricts the attainable spatial and temporal resolution, further limited by the constraints on computational resources. As a result, our experiments were conducted under settings significantly inferior than those specified in the original STCAT paper. Such limitations invariably lead to a decrease in the quality of pseudo labels, thereby increasing the difficulty of our domain adaptation process. Although these constraints do not undermine the validity of the experiments supporting our method’s efficacy, they undeniably prevent us from achieving superior results. On a more positive note, during inference, both the student branch and domain discriminators are no longer required, which ensures that the costs associated with inference time and storage remain unaffected by the aforementioned limitations. Looking forward, we intend to explore strategies to reduce the usage of computational resources and speed up the convergence rate.

## REFERENCES

- [1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [3] Yang Jin, Zehuan Yuan, Yadong Mu, et al. 2022. Embracing consistency: A one-stage approach for spatio-temporal video grounding. *Advances in Neural Information Processing Systems* 35 (2022), 29192–29204.
- [4] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. 2021. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1780–1790.
- [5] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. 2022. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329* (2022).
- [6] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).
- [7] Hamid Rezaatofghi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 658–666.
- [8] Cristian Rodriguez, Edison Marrese-Taylor, Fatemeh Sadat Saleh, Hongdong Li, and Stephen Gould. 2020. Proposal-free temporal moment localization of a natural-language query in video using guided attention. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2464–2473.
- [9] Zongheng Tang, Yue Liao, Si Liu, Guanbin Li, Xiaojie Jin, Hongxu Jiang, Qian Yu, and Dong Xu. 2021. Human-centric spatio-temporal video grounding with visual transformers. *IEEE Transactions on Circuits and Systems for Video Technology* 32, 12 (2021), 8238–8249.
- [10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [11] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. 2022. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 36. 2567–2575.
- [12] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. 2022. Tubedetr: Spatio-temporal video grounding with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16442–16453.
- [13] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. 2019. Semantic conditioned dynamic modulation for temporal sentence grounding in videos. *Advances in Neural Information Processing Systems* 32 (2019).
- [14] Zhu Zhang, Zhou Zhao, Yang Zhao, Qi Wang, Huasheng Liu, and Lianli Gao. 2020. Where does it exist: Spatio-temporal video grounding for multi-form sentences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10668–10677.
- [15] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2020. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159* (2020).