Appendix of UniVTG

A. CLIP teacher strategy

The concept bank is a class list for open-world detection, sourced from here[1]. This list comprises $19,995$ class names, such as "Sandwich Cookies," "Air conditioning," and "Advertising." After conducting a manual check, we determined that the class list can effectively encompass the majority of common concepts.

In our approach, we begin by capturing frame-level clip image features from the video at a rate of 2 fps. Following this, we calculate their respective similarity scores in relation to the given class list. We then determine top-5 classes with the highest average scores, representing the most significant concepts within the video.

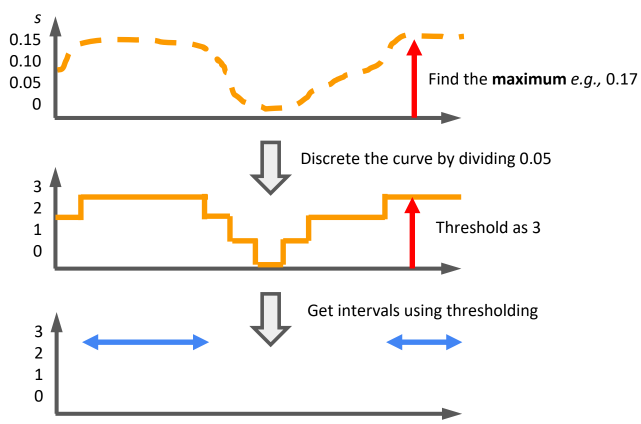


Figure 1: Demonstration of how to threshold each video's curve.

To derive intervals from the curve obtained from the diverse distributions, a fixed threshold is hard to determined and lacks the flexiblity. Thus, we discretize the continuous curve by a small value of 0.05 and pick the maximum discrete value as our threshold. Then, adjacent clips that share the maximum discrete value to form an interval. In this way, we may produce multiple temporal windows from one video. This process is shown in Fig. 1.

B. Datasets

Pretraining corpus. To establish our pretraining corpus, we collect data through three ways: For point labels, we extract the timestamped narrations from Ego4D [2] by *excluding the NLQ val / test splits*. For interval labels, we select a subset of videos (less than 300K) sourced from VideoCC [2], and treat their start and end timestamp as windows and caption as query. For curve labels, we derive them from the above VideoCC subset videos. Below, we describe the benchmarks used for the four settings separately.

(i) Joint Moment Retrieval and Highlight Detection. QVHighlights [4] is the only dataset with available annotations for both moment retrieval and highlight detection, making it an ideal choice for benchmarking multi-task joint optimization. This dataset contains $10,148$ videos with an average length of 150 sec that covers daily vlogs, travel vlogs, and news events scenarios. There are a total of $10,310$ queries associated with $18,367$ moments (on average, 1.8 disjoint moments per query in the video).

(ii) Moment Retrieval. We utilize three benchmarks to further evaluate moment retrieval: Charades-STA [1], Ego4D Natural Language Queries (NLQ) [2] and TACoS [7]. (a) Charades-STA contains $16,128$ indoor videos with an average length of 30.6 sec, which are made up of $12,408$ query-interval pairs for training and $3,720$ query-interval pairs for testing. (b) NLQ focuses on daily egocentric scenarios, where videos are $8-20$ minutes long and queries are question, e.g."What did i pour in the bowl?", making this benchmark challenging. The training set contains 11.3K annotated queries from 1K videos, whereas the validation set contains 3.9K queries from 0.3K videos. (c) TACoS contains 127 videos with an average duration of 4.78 minutes, where 75 videos are used for training, 27 and 25 videos for validation and testing, respectively.

(iii) Highlight Detection. We utilize two benchmarks to further evaluate highlight detection: YouTube Highlights [10] and TVSum [9]. (a) YouTube Highlights has 6 domains with 433 videos, where video titles are not provided, thus we use the domain name of each video as text queries. (b) While TVSum includes 10 domains, each with 5 videos, we use their video titles as text queries. We follow [5] data splits that the ratio of training:testing is 0.8:0.2.

(iv) Video Summarization. We utilize the QFVS [8] benchmark to evaluate the video summarization. This dataset includes the four videos in UT Egocentric dataset [3]. Each video is recorded in daily life and lasts between $3-5$ hours. Each query in this dataset is represented by two words from a total of 48 pre-defined concepts.

C. Experimental settings

(i) In Tab. 1, we detail the parameters for each setting. Notably, for highlight detection benchmarks YouTube Highlights and TVSum, which contain multiple domains treated as separate splits, we perform parameters tuning for λ_{intra} within each domain. Then we aggregate the results obtained using optimal settings. The optimal settings are listed in Tab. 2-3.

(ii) During training, to maintain the balance between positive and negative samples, we allocate a weight of 0.1 to the negatives ($f_i = 0$) in binary cross-entropy loss Eq. **??**.

(iii) When inferring highlights scores, we observe that

[1] `https://storage.googleapis.com/openimages/v6/oidv6-class-descriptions.csv`

[2] `https://github.com/google-research-datasets/videoCC-data`

Type	**Datasets**	l	BS	Epoch	Warmup	LR	Weight dacay	Gamma	LR drop	$\lambda_{\text{SmoothL1}}$	λ_{iou}	λ_{f}	λ_{intra}	λ_{inter}
Pretraining	4.2M corpus	2	64	10	-	$1e^{-4}$	$1e^{-4}$	-	-	10	1	10	0.1	0.1
Joint MR & HL	QVHighlights	2	32	200	10	$1e^{-4}$	$1e^{-4}$	0.1	80	10	1	10	0.05	0.01
Moment Retrieval	NLQ	2	32	200	10	$1e^{-5}$	$1e^{-5}$	0.1	100	10	1	50	0.1	1.0
	Charades-STA	1	32	100	10	$1e^{-5}$	$1e^{-5}$	0.1	100	10	1	10	1.0	0.5
	TACoS	2	32	100	10	$1e^{-4}$	$1e^{-4}$	0.1	30	10	1	10	0.5	0.1
Highlight Detection	YouTube Highlights	$1^{\dagger}$	4	100	10	$1e^{-4}$	$1e^{-4}$	-	-	0	0	1	Search	0
	TVSum	2	4	200	10	$1e^{-4}$	$1e^{-4}$	-	-	0	0	1	Search	0
Video Summarization	QFVS	5	20^{*}	20	0	$5e^{-5}$	$5e^{-5}$	-	-	0	0	1	0.9	0

Table 1: **Parameter selections for each settings** where l denotes the clip length; BS denotes the batch size; LR denotes the learning rate; LR drop denotes the learning rate drop up epoch; Warmup denotes the warmup epoch. Search denotes to parameter searching individually for each domain. $\dagger$ means YouTube Highlights clips has overlapping frames, which is align with the [5]. $*$ means batchsize in QFVS is based on the segment-level instead of video-level.

Domains	Dog	Gyn	Par.	Ska.	Ski.	Sur.
λ_{intra}	0.6	0.5	0.4	0.5	0	0.7

Table 2: Optimal λ_{intra} under each domain in the Youtube HL.

Domains	BK	BT	DS	FM	GA	MS	PK	PR	VT	VU
λ_{intra}	0.7	0.9	0.6	0.4	0.1	0.1	0	0.6	0.1	0.5

Table 3: Optimal λ_{intra} under each domain in the TVSum.

$\{\tilde{f}_i + \tilde{s}_i\}_{i=1}^{L_v}$ can typically achieves better performance in QVHighlights, while for smaller datasets YouTube Highlights and TVSum, using $\tilde{f}_i$ yield more reliable prediction.

(iv) For video summarization, we adhere to the same preprocessing settings in [11], which extracts video frame features at 1 FPS and take a 5 seconds as a clip and compute the average frame feature within a clip to generate its clip-level feature. By applying the KTS algorithm [6], we split a long video into small segments under the conditions that the number of segments in a video is no more than 20 and each segment contains no more than 200 clips.

During evaluation, we compute the foreground scores $\tilde{f}_i$ for each segment within a video, then aggregate these scores to derive an overall video score which is used to compute the metrics. We calculate the conceptual similarity between each two video clip based on the intersection-over-union (IOU) of their related concepts. This conceptual similarity is then used as edge weights in a bipartite graph between two summaries, which aids in identifying the maximum weight match in the graph. Finally, precision, recall, and F1 scores can be determined based on the matching pairs.

D. Ablation studies of training objective

Since we use identical training objectives during the stages of pretraining and downstream transferring. To gain a more thorough understanding of the impact each component has, we have constructed ablation studies as seen in Tab. 4, where the top half, we study the **effect of downstream training** objectives (without introduce any pretraining), while in the bottom half, we investigate the **effect of pretraining training** objectives (the downstream tuning use the same optimal parameter settings).

Pretraining					**Downstream**					**MR@QVHL**		**HL@QVHL**		**MR@NLQ**		**MR@TaCoS**	
$\mathcal{L}_{\text{f}}$	$\mathcal{L}_{\text{SmoothL1}}$	$\mathcal{L}_{\text{iou}}$	$\mathcal{L}_{\text{s}}^{\text{inter}}$	$\mathcal{L}_{\text{s}}^{\text{intra}}$	$\mathcal{L}_{\text{f}}$	$\mathcal{L}_{\text{SmoothL1}}$	$\mathcal{L}_{\text{iou}}$	$\mathcal{L}_{\text{s}}^{\text{inter}}$	$\mathcal{L}_{\text{s}}^{\text{intra}}$	R1@0.5	mAP	mAP	HIT@1	R1@0.3	mIoU	R1@0.3	mIoU
					✓	✓				54.71	29.64	33.12	46.13	5.96	3.97	48.46	30.20
					✓	✓	✓			58.71	35.89	33.21	45.03	6.50	4.43	50.09	32.42
					✓	✓	✓	✓		59.16	36.24	38.59	61.81	6.97	4.88	51.14	33.05
					✓	✓	✓	✓	✓	59.74	36.13	38.83	61.81	7.28	4.91	51.44	33.60
✓					✓	✓	✓	✓	✓	62.00	39.45	39.59	64.00	8.83	5.82	52.04	32.72
✓	✓				✓	✓	✓	✓	✓	63.29	40.43	39.82	64.19	8.49	5.73	51.71	34.76
✓	✓	✓			✓	✓	✓	✓	✓	64.52	41.65	39.93	63.68	8.49	5.74	53.11	34.48
✓	✓	✓	✓		✓	✓	✓	✓	✓	64.45	41.84	40.07	64.32	9.86	6.52	53.89	36.76
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	68.39	45.99	41.25	67.42	11.74	7.88	56.11	38.63

Table 4: **Ablation studies of downstream (top) and pretraining objective (bottom)** on QVHighlights val split, NLQ val split and TACoS val split.

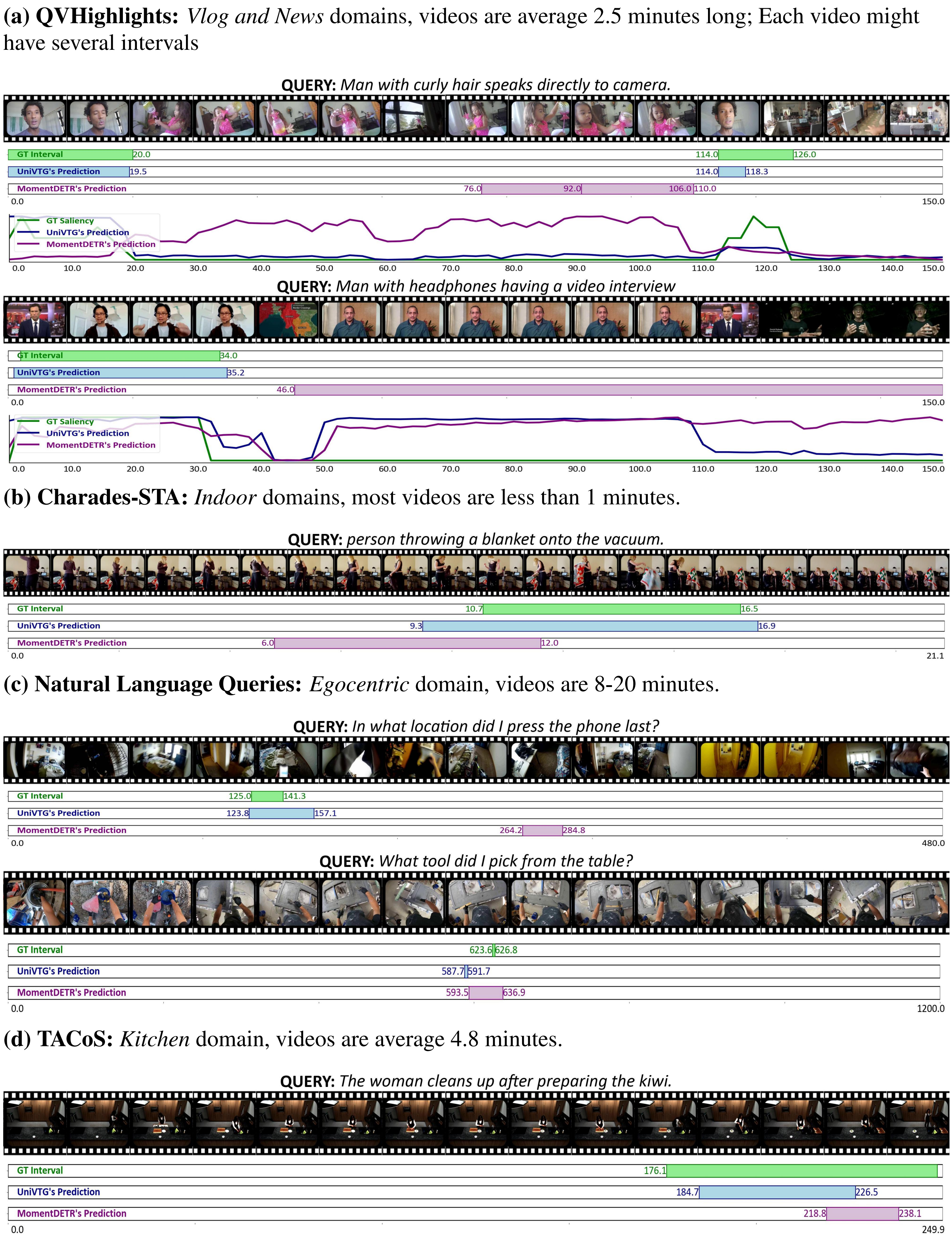
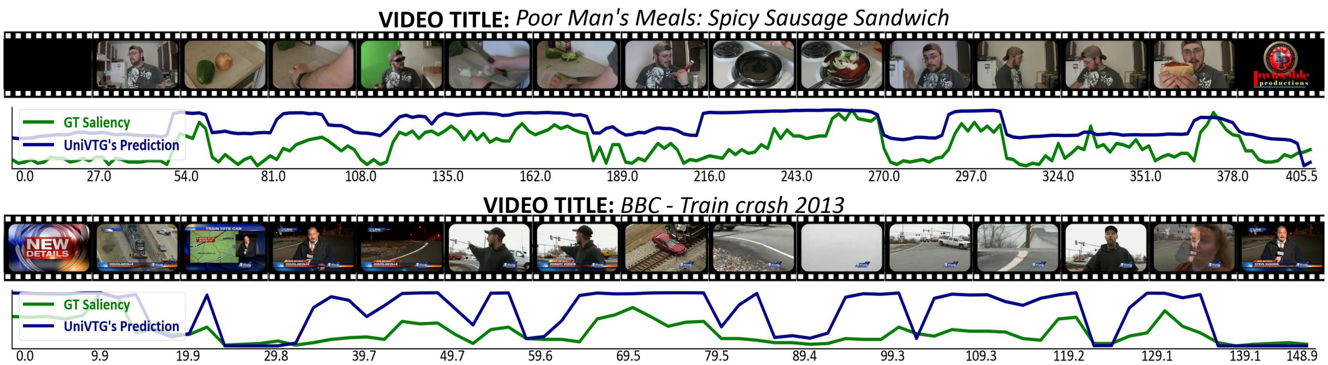
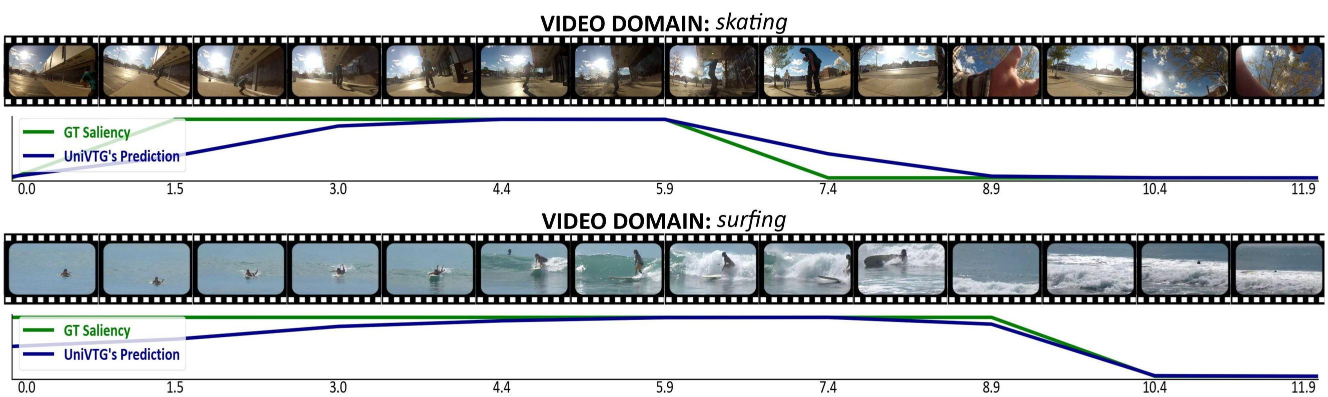


Figure 2: Visualization of **Joint moment retrieval and highlight detection** on (a) QVHighlights, and **Moment Retrieval** on (b) Charades-STA, (c) Ego4D, (d) TACoS. Textual queries are mostly *natural sentences*.

(e) TVSum: *Web* diverse domain, videos are average 4.2 minutes long.



(f) YouTube Highlights: *Youtube* diverse domain, videos are average 1.5 minutes long.



(g) Query-Focused Video Summarization: *Egocentric* domain, each video is between 3-5 hrs.

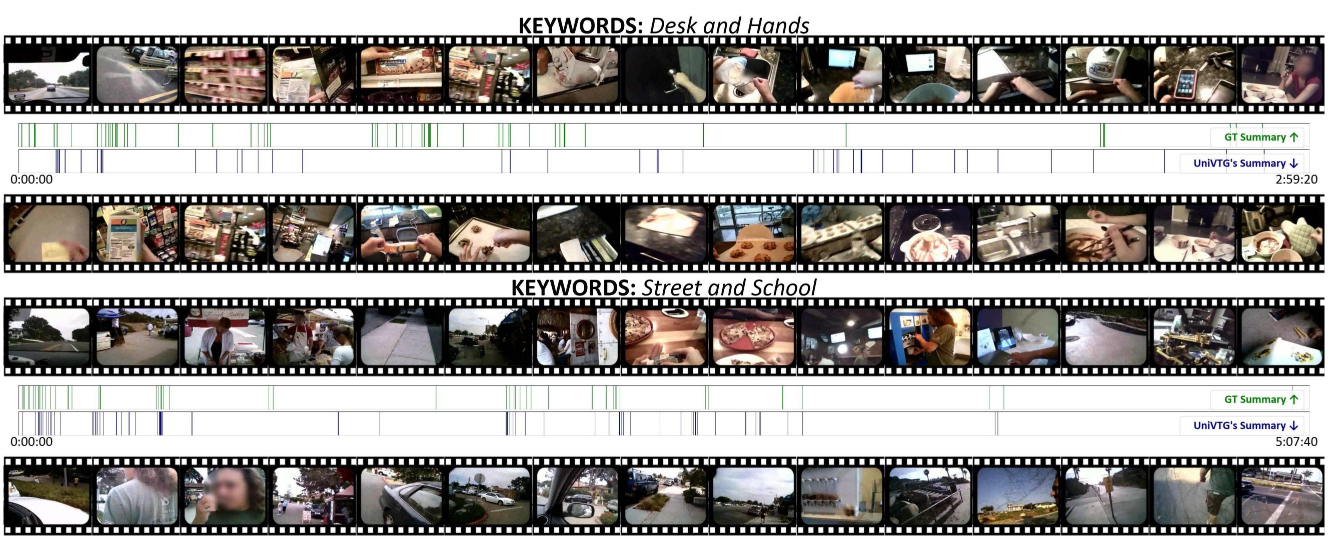


Figure 3: Visualization of **Highlight Detection** on (e) TVSum, (f) YouTube Highlights; and **Video Summarization** on (g) QFVS. Textual queries can be *video title* (e), *video domain* (f), and *keywords* (g).

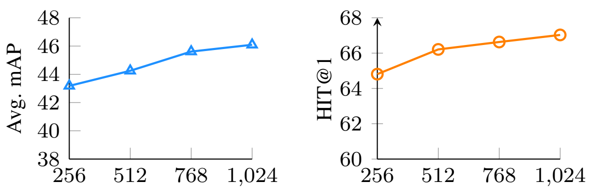
E. Parameters sensitivity

Transformer layers. In Tab. 5, we abalate the transformer layers $L \in [1, 2, 3, 4, 6, 8]$ of multi-modal encoder in our unified model (without pretraining).

# Layers	MR		HD	
	R1@0.5	mAP	mAP	HIT@1
1	47.16	26.62	37.35	60.65
2	55.25	30.70	38.33	60.52
3	59.03	34.06	38.57	62.13
4	59.74	36.13	38.83	61.81
6	61.55	39.88	39.20	63.42
8	60.32	38.24	38.72	60.90

Table 5: **Ablation studies of different transformer layers for multi-modal encoder** on QVHighlights val split.

Projector dimension. In Fig. 4, we study the effect of projector dimension from 256 to 1024 (without pretraining).



(a) Avg. mAP of moment retrieval.

(b) HIT@1 of highlight detection.

Figure 4: **Ablation studies of projector dimension** on QVHighlights val split.

F. Loss weights

In Tab. 6, we study the effect of foreground loss on three moment retrieval benchmarks (with pretraining).

λ_f	QVHighlights		NLQ		TACoS	
	R1@0.5	mAP	R1@0.3	mIoU	R1@0.3	mIoU
0.1	66.97	46.02	9.24	6.64	46.51	33.16
0.5	66.19	46.08	9.50	6.75	50.21	35.06
1	67.74	46.22	9.53	6.80	51.79	35.94
5	67.35	45.63	9.89	6.88	54.01	37.59
10	67.81	45.46	7.26	7.36	54.44	37.55
25	68.00	45.06	11.41	7.77	54.31	37.27
50	66.71	44.32	11.13	7.49	54.21	35.61

Table 6: **Ablation studies of foreground loss weight λ_f** on QVHighlights, NLQ, and TACoS moment retrieval benchmarks.

G. Visualizations

In Fig. 2 and 3, we show quantitative visualizations of UniVTG predictions across different settings and domains.

References

[1] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017.

[2] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *CVPR*, pages 18995–19012, 2022.

[3] Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, pages 1346–1353, 2012.

[4] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *NeurIPS*, pages 11846–11858, 2021.

[5] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *CVPR*, pages 3042–3051, 2022.

[6] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *ECCV*, pages 540–555, 2014.

[7] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Trans. Assoc. Comput. Linguistics*, 1:25–36, 2013.

[8] Aidean Sharghi, Jacob S Laurel, and Boqing Gong. Query-focused video summarization: Dataset, evaluation, and a memory network based approach. In *CVPR*, 2017.

[9] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *CVPR*, pages 5179–5187, 2015.

[10] Min Sun, Ali Farhadi, and Steve Seitz. Ranking domain-specific highlights by analyzing edited videos. In *ECCV*, pages 787–802, 2014.

[11] Shuwen Xiao, Zhou Zhao, Zijian Zhang, Xiaohui Yan, and Min Yang. Convolutional hierarchical attention network for query-focused video summarization. In *AAAI*, volume 34, pages 12426–12433, 2020.