MVMR: Evaluating Natural Language Video Localization Performance on Multiple Reliable Videos Pool

Anonymous ACL submission

Abstract

With the explosion of multimedia content in recent years, Video Corpus Moment Retrieval (VCMR), which focuses on detecting a video moment that matches a given natural language query from multiple videos, has become a critical problem. However, existing VCMR studies have regarded all videos not paired with a specific query as negative, neglecting the possibility of including false negatives when constructing the negative video set. In this paper, we propose an **MVMR** (**M**assive **V**ideos **M**oment **R**etrieval) task that aims to localize video frames within a reliable massive video set, including clearly identified positive and negative videos. For this task, we suggest methods for reliably constructing datasets by employing semantic similarity filtering on the existing video moment localization datasets and introduce three MVMR datasets. To solve the MVMR task, we further propose a strong model, Reliable Mutual Matching Network (RMMN), which employs cross-directional contrastive learning that selectively filters the reliable and informative negatives, making the model more robust on the MVMR task. Experimental results on the introduced datasets reveal that existing video moment search models are easily distracted by negative video frames, whereas our model shows significant performance.[1]

1 Introduction

Enabled by the increased accessibility of video-sharing platforms along with the advancement of networking and storage technology, a vast number of new content is available on the web daily. With this huge number of contents, Natural Language Video Localization (NLVL) task, searching for a video moment suitable for a given query, has emerged as an essential problem (Anne Hendricks et al., 2017; Gao et al., 2017). However, this line of research narrows down the task to an impractical setting by targeting the search over a single video; thus, it bears limitations in retrieving relevant information over whole media pools.

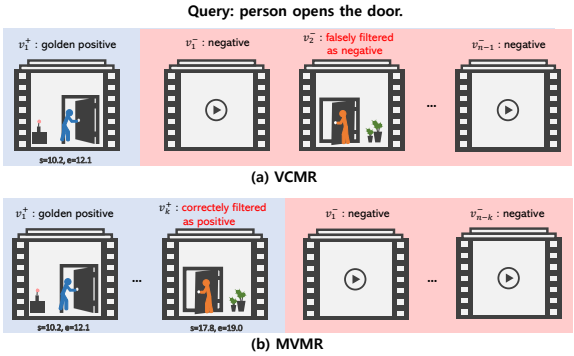


Figure 1: **MVMR vs. VCMR.** In existing VCMR studies, only a single golden positive video moment is designated as positive, while entire videos not paired with a specific query are regarded as negative. Our proposed MVMR filters the positive video moment for a query from the whole video set; thus, it considers more practical and generalized use cases with reliable settings. v_i^+ and v_j^- mean a positive and a negative video, respectively.

Therefore, Video Corpus Moment Retrieval (VCMR) task that retrieves a correct positive moment among numerous negative videos has emerged (Escorcia et al., 2019; Lei et al., 2020; Ma and Ngo, 2022; Jung et al., 2022). Nonetheless, the VCMR task also has limitations as it merely extends the NLVL task by considering only one paired positive video moment to a specific query as a positive, while categorizing all unpaired videos as negative videos. This assumption diverges from the practical setting of video moment search where multiple positive videos exist in whole media collections. Furthermore, the existing VCMR studies have overlooked the potential presence of false-negative videos within the negative video set when constructing video retrieval pools for a specific query. As a result, evaluating the performance of NLVL models using an existing VCMR framework

[1]We have opened an anonymous GitHub page to make our code and dataset publicly available: link

may lead to unreliable results. For example, in the TACoS test dataset, it is noteworthy that the number of videos with queries exactly matching the text *"The person gets out a knife."* amounts to 60% of all the videos. It is a significant problem since more videos will likely be incorrectly classified as negatives when expanding the matching criteria to include broader semantic similarities.

To bridge these gaps, we propose an **MVMR** (**M**assive **V**ideos **M**oment **R**etrieval) task, which expands the search coverage to a massive video set that can include any number of positive videos as shown in Figure 1. Specifically, given a query q with a positive video set $V_q^+ = \{v_1^+, ..., v_k^+\}$ and a negative video set $V_q^- = \{v_1^-, ..., v_{n-k}^-\}$, the MVMR task aims to detect temporal moments in the positive videos V_q^+ that matches the query from a massive video set $V_q^{+,-} = \{v_1^+, ..., v_k^+, v_1^-, ..., v_{n-k}^-\}$.

Furthermore, we propose novel methods to construct a reliable MVMR dataset by leveraging publicly available NLVL datasets. A critical challenge in converting an NLVL dataset into a reliable MVMR dataset is accurately defining positive and negative videos for each query. To address this challenge, we employ a semantic text embedding model and a video captioning evaluation model to define reliable MVMR positive and negative videos. Specifically, we examine the similarity between a target query and all videos in an NLVL dataset using these models to define a massive video set for each target query. To this end, we construct three MVMR datasets, and we confirm that three samples of the introduced datasets contain only 1.5%, 0.2%, and 3.5% falsely defined videos, showing that our approach successfully creates a practical benchmark dataset from human evaluation.

The key aspect of solving the MVMR task is distinguishing positives from negative videos. To tackle the MVMR task, we introduce a novel model architecture, Reliable Mutual Matching Network (RMMN). The RMMN successfully addresses the primary challenge of the MVMR task by utilizing reliable and hard negative sampling methods during the training procedure. Specifically, the RMMN distinguishes and excludes false and easy negatives to derive reliable and informative negative samples for a cross-directional contrastive learning approach.

We demonstrate the performance of strong baseline models, including a state-of-the-art model, to compare with our model in the new scenario (MVMR). The experimental results show that the performance of baseline models significantly degrades on the new task. For example, the state-of-the-art model, MMN, shows average 29.7% degradation on R@1(IOU0.5) for all MVMR datasets, revealing the difficulty of the MVMR task. In contrast, our RMMN outperforms baseline models significantly in the MVMR setting, and this result proves that reliable and hard negative sampling methods positively influence the performance of the MVMR task. Moreover, we solve the MVMR task in a more practical scenario of video moment retrieval by constructing a pipeline with a video retrieval model, showing that the RMMN still outperforms the strong baseline.

2 Backgrounds

2.1 Natural Language Video Localization

Natural language video localization (NLVL), which aims to detect a specific moment in a video that matches a given text query, is one of the most challenging problems in the video-language multimodal domain (Zhang et al., 2020b,a; Nan et al., 2021; Gao and Xu, 2021; Liu et al., 2021; Wang et al., 2022). Therefore, the NLVL task has been explored extensively, with numerous effective methods emerging over time. Some novel approaches have regarded the problem as question-answering (QA) tasks (Seo et al., 2018; Joshi et al., 2020), by leveraging a QA model to encode a multi-modal representation and then predicting the frames that correspond to the start and end of the moment that match a query (Ghosh et al., 2019; Zhang et al., 2020a). However, NLVL studies operate within an impractical framework, focusing on searching within a single video. Consequently, they have limitations in retrieving relevant information across the whole media repositories.

2.2 Video Corpus Moment Retrieval

Existing studies (Escorcia et al., 2019; Lei et al., 2020; Ma and Ngo, 2022; Jung et al., 2022) have tried to extend the NLVL to search on multiple video pools, naming it the Video Corpus Moment Retrieval (VCMR) task, which focuses on retrieving a moment from multiple videos for a given query. However, the VCMR simply extends the NLVL task by designating only one matched positive video moment for a target query as a positive, while labeling all unmatched videos as negative

videos. This approach deviates from the practical scenario of the video moment search since multiple positive videos can be present for a target query. Moreover, existing VCMR studies have failed to consider the possibility of false negative when constructing the negatives video set for a specific query, which poses a significant reliability issue.

3 Problem Definition

We evaluate video moment retrieval models under an MVMR dataset constructed with the k positive sample videos and the $n-k$ negative sample videos for each query. Given a positive video set $V_q^+ = \{v_1^+, ..., v_k^+\}$ and a negative video set $V_q^- = \{v_1^-, ..., v_{n-k}^-\}$, the MVMR task aims to localize a temporal moment (x_s^v, x_e^v) of a specific video v that matches the query from massive video set $V_q^{+,-} = \{v_1^+, ..., v_k^+, v_1^-, ..., v_{n-k}^-\}$, where x_s^v and x_e^v mean start and end points of the moment, respectively. In the MVMR task, we aim to retrieve an optimal positive moment $(x_s^{v^+}, x_e^{v^+})^*$ that matches semantically with the text query q. We derive confidence scores for moments of the whole n massive videos using an NLVL model and select the moment with the highest score as a prediction.

4 MVMR Dataset Construction

In this section, we define the MVMR task and describe how to build an MVMR dataset using a public NLVL dataset. The process of building an MVMR dataset is as follows: (1) We calculate similarities between a target query and all videos using semantic text matching and visual text matching methods. (2) We add videos with a similarity higher than a positive threshold as positive candidates for the target text query. Similarly, we include videos with a similarity lower than a negative threshold as negative candidates. (3) We sample MVMR positive and MVMR negative videos for a target query from the pre-defined positive and negative candidates pool.

4.1 Query Similarity Filtering

This section describes how to calculate the similarity between a target query q and a video v using a semantic text embedding model, SimCSE (Gao et al., 2021), to decide which videos to filter as positives and negatives. NLVL datasets consist of videos set $V = \{v_1, ..., v_{l_v}\}$ and paired queries set $Q_v = \{q_1^v, ..., q_{l_q}^v\}$ for each video v, where l_v and l_q is the number of videos and text queries included in a dataset, respectively. Since text queries in a video describe the video, we use the queries to obtain semantic information about the video. Specifically, we calculate the similarity between a target query and each video's queries. We first obtain an embedding e_q of the target text query q using SimCSE. We also derive query embeddings set $E_v = \{e_{q_1^v}, ..., e_{q_{l_q}^v}\}$ by embedding all video text queries Q_v. We calculate cosine similarity between e_q and $e_{q_i^v}$ to obtain the similarity $s_{te}(q, q_i^v)$. Then, we investigate the most similar one among the video's queries and define the similarity of it as the similarity between q and v as follows:

$$s_{te}(q, v) = \max(\{s_{te}(q, q_1^v), ..., s_{te}(q, q_{l_q}^v)\}) \quad (1)$$

We select *max aggregation* to obtain $s_{te}(q, v)$ since the video should be regarded as a positive video for the target query, even if only one similar query to a target query exists in a video.

4.2 Query-video Similarity Filtering

Queries assigned to a video may include limited information about the video since they may not cover all the context of the video. Therefore, we additionally adopt a visual semantic matching between a query and a video to filter positive and negative videos. We use EMscore (Shi et al., 2022), a video captioning evaluation model, to calculate the query-video similarity s_{tv}, which is used as a complementary filtering method on videos primarily filtered through the query similarity s_{te}.

4.3 False-negative Video Identification

This section describes how to construct MVMR positive and MVMR negative video sets using the calculated query-video similarities, s_{te} and s_{tv}.

MVMR positive candidates are constructed by (1) primarily filtering out videos with $s_{te}(q, v) < t_{te}^+$ and (2) removing videos that have lower similarity than the mean s_{tv} of all golden positive query-video pairs (q, v^*) for all queries as follows:

$$V^{c,+}(q) = \{v | s_{te}(q, v) \geq t_{te}^+ \cap s_{tv}(q, v) \geq \mu_{s_{tv}}^*\} \quad (2)$$

where $\mu_{s_{tv}}^*$ is the mean similarity of $s_{tv}(q, v^*)$ for all queries. We construct final MVMR positives by randomly sampling k videos from this set of positive candidates.

MVMR negative candidates are defined by (1) filtering out videos with $s_{te}(q, v) > t_{te}^{-}$ and (2) excluding videos that have higher similarity than the mean s_{tv} of query-video pairs (q, v^{-}) that is primarily filtered as negatives using s_{te} as follows:

$$V_{te}^{c,-}(q) = \{v | s_{te}(q, v) \leq t_{te}^{-}\}, \quad \forall v \in V$$
$$\overline{\mu}_{s_{tv}} = \frac{1}{N} \sum_{i=1}^{N} s_{tv}(q, v_i), \quad \forall (q, v_i) \in V_{te}^{c,-} \tag{3}$$
$$V^{c,-}(q) = \{v | s_{tv}(q, v) \leq \overline{\mu}_{s_{tv}}\}, \ \forall v \in V_{te}^{c,-}(q)$$

where N is the number of video samples in $V_{te}^{c,-}$. Similar to MVMR positives, we construct final MVMR negatives by randomly sampling $(n - k)$ videos from this set of negative candidates. In this process, a total of n videos retrieval pool $V_q^{+,-} = \{v_1^{+}, ..., v_k^{+}, v_1^{-}, ..., v_{n-k}^{-}\}$ is obtained for each query.

5 Reliable Negatives Selection in Training

In this section, we introduce a novel model, Reliable Mutual Matching Network (RMMN), to solve the MVMR task. A key challenge in solving the MVMR task is distinguishing positives from multiple negative distractors. MMN (Wang et al., 2022) is the optimal baseline model for the MVMR task as it uses contrastive learning method to distinguish negative samples from positives effectively. Therefore, we select the MMN as the backbone of our model to solve the MVMR task.

However, MMN also has some limitations, in that it neglects out-video false-negative video moments since it regards all randomly selected in-batch samples as negatives in the training procedure. Furthermore, it overlooks that the importance gap can exist among negative samples. RMMN solves these two limitations of the MMN by adopting two reliable negative selection methods: (1) Reliable true-negative filtering and (2) Reliable hard-negative filtering, to tackle the MVMR task.

5.1 Reliable True-negatives Filtering

The reliable true-negative filtering in the training process is similar to the query similarity filtering of the MVMR dataset construction described in Section 4.1. We pre-construct a query-video similarity matrix using the SimCSE score, and define samples with a similarity lower than t_{tr} as reliable true-negative samples:

$$V_{tr}^{-}(q) = \{v | s_{te}(q, v) < t_{tr}\}, \quad \forall v \in V \tag{4}$$

We use only the filtered true-negative samples as negative samples for contrastive learning.

5.2 Reliable Hard-negatives Filtering

Easy or false-negatives may distract a model from learning proper knowledge during the training process. Inspired by recent progress in using hard-negative sampling for the document retrieval task (Zhou et al., 2022), in this work, we propose a strong hard-negative sampling method, Cross-directional Query-video Matching Ambiguous Negatives Sampling (CM-ANS), to improve the MVMR performance. CM-ANS is a hard negative sampling method performing the cross-directional query-to-video and video-to-query hard-negatives. We categorize negatives using the relevance score between a target query and negative moments as follows: (1) Negatives that are clearly irrelevant and have low relevance scores should be sampled less frequently (Easy-negative); (2) Negatives that are highly relevant and have high relevance scores should also be sampled less frequently (False-negative); (3) Negatives that are uncertain and have relevance scores similar to true-positives should be sampled more frequently since they provide useful information (Hard-negative). Therefore, we define the negative moments sampling distribution as follows:

$$p'(m|q) \propto \exp\left(-a(r(q, m) - \overline{r}(q, m^{+}) - b)^2\right), \ \forall m \in M_q^{-} \tag{5}$$

where $r(q, m)$ is a relevance score between a query and a moment calculated using the already trained video moment retrieval model, and $\overline{r}(q, m^{+})$ is a mean relevance score for all positive moments. M_q^{-} is the negative moment candidates for a query q. a is a hyper-parameter to control the density of the distribution. Likewise, we can define the negative query sampling probability distribution as follows:

$$p'(q|m) \propto \exp\left(-a(r(q, m) - \overline{r}(q^{+}, m) - b)^2\right), \ \forall q \in Q_m^{-} \tag{6}$$

where Q_m^{-} means the negative query candidates for each video moment. Finally, we reformulate the formula to reduce the computational cost by converting the moment m to the video v as follows:

$$\begin{aligned} r(q, v) &= \max\left(\{r(q, m_1), ..., r(q, m_l)\}\right), \quad \forall m_i \in v \\ p'(v|q) &\propto \exp\left(-a(r(q, v) - \overline{r}(q, v^{+}) - b)^2\right), \quad \forall v \in V_q^{-} \end{aligned} \tag{7}$$

$p'(q|v)$ can be formulated similarly by calculating $\overline{r}$ about positive queries for each video. We define the distributions for negatives, and train our model using contrastive learning by sampling hard-negatives from them. These distributions are used to train pre-trained models further to focus on learning representations of informative negatives.

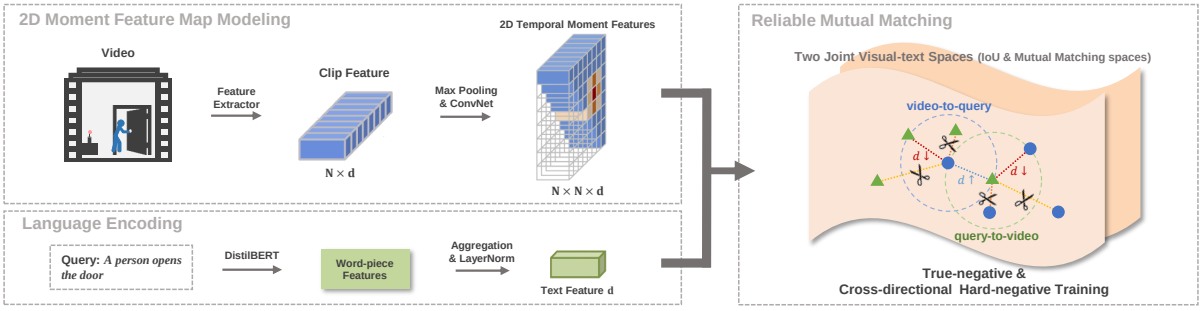


Figure 2: **Reliable Mutual Matching Network.** We apply true and cross-directional hard negatives filtering for the MVMR task. The dots and triangles are the feature of moments and texts. The blue dash lines are matched moment-text pairs to be pulled in, while the red dash lines are negative samples of intra/inter-video to be pushed away. The yellow and orange dash lines are unmatched moment-text pairs, but not to train by filtering out since they are easy and false negatives, respectively.

5.3 Reliable Mutual Matching Network

This section describes a robust model, Reliable Mutual Matching Network (RMMN), that uses our reliable negative sampling methods to solve the MVMR task. RMMN adopts a bi-encoder architecture with a late modality fusion by an inner product in the joint visual-text representational space following the baseline MMN. By adopting a bi-encoder architecture, we can efficiently search for moments with pre-calculated video moment representations, given a user query. The whole model architecture of the RMMN is shown in Figure 2. RMMN represents each query and video moment using query and video encoders to two joint visual-text spaces (IoU and mutual matching spaces). From the joint visual-text space, RMMN computes the similarity between the query and video moment representations and selects the best-matched video moment. The details of the model architecture are described in the appendix C.1.

We select the binary cross entropy and contrastive learning losses and train our RMMN following two steps: (1) the first training step using the two loss functions (binary cross entropy and contrastive learning losses) with only true-negatives by filtering out false-negatives described in Section 5.1; (2) the second training step using two loss functions with only hard-negatives by removing false-negatives and easy-negatives as Section 5.2.

To calculate the similarity between the text and video moment features projected in the IoU space, we compute the cosine similarity s^{iou}. To adjust the range of the final prediction, we multiply the cosine similarity by a factor of 10, following the MMN. This amplification results in the final prediction $p_i^{iou} = \sigma(10 \cdot s_i^{iou})$, where σ represents the sigmoid function. The binary cross-entropy loss is then calculated as follows:

$$L_{bce} = -\frac{1}{C}\sum_{i=1}^{C}\left(y_i \log p_i^{iou} + (1-y_i)\log\left(1-p_i^{iou}\right)\right) \tag{8}$$

where p_i^{iou} is the confidence score of each moment, and C is the total number of valid candidates. We calculate L_{bce} not only for the positive samples but for the reliable negatives.

We use the cross-directional contrastive learning loss to effectively train the model using the informative negatives derived from the methods described in Section 5.1 and 5.2. The mutual matching contrastive learning loss considering the reliable negative samples selection process is as follows:

$$\begin{aligned} p^*(q_i|m) &= \frac{\exp((f_i^{q\top} f^m - \psi)/\tau_m)}{\exp((f_i^{q\top} f^m - \psi)/\tau_m) + \sum_{j\in\mathcal{I}_m^-}\exp(f_j^{q\top} f^m/\tau_m)} \\ p^*(m_i|q) &= \frac{\exp((f_i^{m\top} f^q - \psi)/\tau_q)}{\exp((f_i^{m\top} f^q - \psi)/\tau_q) + \sum_{j\in\mathcal{I}_q^-}\exp(f_j^{m\top} f^q/\tau_q)} \\ L_{rmm} &= -\left(\sum_{i=1}^{N}(\log p^*(i_m|q_i)) + \sum_{i=1}^{N}(\log p^*(i_q|m_i))\right) \end{aligned} \tag{9}$$

where q_i and m_i are corresponding positive query and moment for each moment and query, respectively. f^m and f^q are the moment and query features in joint visual-text space. τ_q and τ_m are temperatures. $\mathcal{I}^-$ means the indices of informative negatives derived using our sampling methods.

The final loss, denoted as L, is formulated as a linear combination of the binary cross-entropy loss and the reliable mutual matching loss. The matching score s, for a specific moment given the text query, is obtained by multiplying the iou score s^{iou}, with the reliable mutual matching score s^{rmm}. The reliable mutual matching score is calculated for the features in the mutual matching space the same as the iou score.

Dataset	#Queries/#Videos	Avg Len (sec) Moment/Video	Avg #Moments per Query	Avg Query Len	#Retrieval Pool	#Max Positive Moments per Query
$\text{MVMR}_{\text{Charades-STA}}$	3716 / 1334	7.83 / 29.48	3.07	6.23	50	5
$\text{MVMR}_{\text{ActivityNet}}$	16941 / 4885	40.32 / 118.20	1.11	12.03	50	5
$\text{MVMR}_{\text{TACoS}}$	2055 / 25	28.55 / 367.15	2.24	7.31	5	5

Table 1: **Summary of datasets.** Remarkably, the queries of $\text{MVMR}_{\text{Charades-STA}}$ and $\text{MVMR}_{\text{TACoS}}$ generally include multiple positive moments, and it proves that existing VCMR studies are exposed to the risk of false negatives.

$$L = L_{bce} + \lambda L_{rmm}, \quad s = s^{iou} \cdot s^{rmm} \tag{10}$$

6 Experiments

6.1 NLVL Datasets

We construct the MVMR evaluation datasets (§4) using three widely-used NLVL datasets (Charades-STA (Gao et al., 2017), ActivityNet (Krishna et al., 2017), and TACoS (Regneri et al., 2013)) and name them $\text{MVMR}_{\{\$\text{DATASOURCE}\}}$. Each NLVL dataset consists of multiple videos with query-moment pairs. The details of each NLVL dataset are shown in Appendix A.

6.2 MVMR Settings.

This section describes how to derive MVMR positive and negative candidates using SimCSE and EMScore in details.

SimCSE Filtering. We empirically validate varying thresholds and choose the best one by comparing the filtered similar texts for each threshold. For example, if a threshold about 0.7 is chosen, query instances with lexical overlaps are derived, even if the meaning is completely shifted since the subject is different (*e.g. "A man eats some food." vs "A girl eats some food."*). Consequently, we select $t_{te}^{+} = 0.9$ to derive MVMR positive candidates as a conservative threshold for the query similarity filtering (§4.1 and §4.3). Since we should be more careful to detect positives than negatives, we select a conservative threshold for $t_{te}^{+} = 0.9$. To filter MVMR negative candidates, we select $t_{te}^{-} = 0.5$ to construct more reliable datasets. The detailed qualitative analysis for the SimCSE filtering is shown in Appendix B.

EMScore Filtering. We set $\overline{\mu}_{s_{tv}}$ as the EMScore threshold to filter out false-negatives to construct MVMR candidates (§4.2 and §4.3). We calculate $\overline{\mu}_{s_{tv}}$ approximately by sampling one negative video about each query to have computational efficiency.

6.3 MVMR Analysis.

We set a video retrieval pool as a total of n, containing at most k positive videos for each query. If the number of MVMR positive candidates derived for each query is less than k, we include all positive candidates as positive samples. Also, if the total number of MVMR positive and MVMR negative candidates for a query is less than n, the query is excluded from the final MVMR dataset.

Datasets Statistic. We set the number of videos in the $\text{MVMR}_{\text{Charades-STA}}$ and the $\text{MVMR}_{\text{ActivityNet}}$ retrieval pools as $n = 50$, and the maximum number of positive moments per query k is set to 5. The former consists of 3,716 queries for 1,334 videos, and each query contains an average of 3.07 positive videos. The latter contains 16,941 queries for 4,885 videos, each including an average of 1.11 positive videos. We set the number of videos for the $\text{MVMR}_{\text{TACoS}}$ and the maximum number of positive moments per query as 5 and 5, respectively, since TACoS test set contains only 25 videos. It includes 2,055 queries for 25 videos, each containing an average of 2.24 positive videos. The summary of the MVMR datasets is shown in Table 1.

Analysis on Dataset Filtering Methods. We visualize the scores derived using SimCSE and EMScore to qualitatively analyze our MVMR datasets. Figure 3 displays SimCSE embeddings of all queries in each dataset, illustrating the similarity between all queries and a specific target query, *"person turn a light on."* (left) and *"He is using a push mower to mow the grass."* (right). We use T-SNE to reduce the dimensionality of each query for displaying each query (dot).

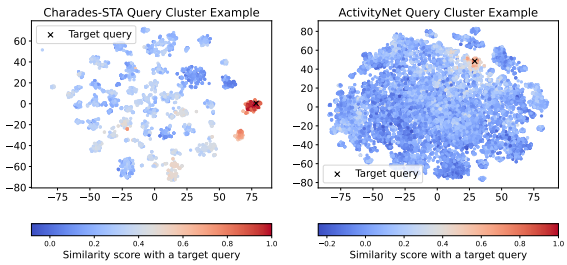


Figure 3: **SimCSE Similarity Distribution.**

Dataset	Model	R@1 IOU0.3	R@1 IOU0.5	R@1 IOU0.7	R@5 IOU0.3	R@5 IOU0.5	R@5 IOU0.7	R@20 IOU0.3	R@20 IOU0.5	R@20 IOU0.7	R@50 IOU0.3	R@50 IOU0.5	R@50 IOU0.7
$\text{MVMR}_{\text{Charades-STA}}$	CET	3.96	2.96 (41.77)	2.02 (21.80)	15.34	11.79 (61.10)	6.97 (40.91)	39.85	32.21	19.91	59.58	50.54	33.88
	BET	4.57	2.99 (38.25)	1.94 (20.73)	14.26	10.98 (61.29)	7.00 (39.27)	31.05	25.11	16.55	45.88	37.86	26.08
	MMN	13.99	12.78 (47.18)	8.40 (27.47)	37.86	33.29 (83.71)	22.36 (56.96)	64.80	57.83	41.71	80.11	73.09	56.75
	RMMN†	18.38	16.31 (47.55)	10.76 (27.80)	**43.86**	**39.05** (83.63)	26.99 (58.28)	**69.48**	**62.89**	46.39	82.88	76.08	60.28
	RMMN	**19.27**	**17.38** (48.06)	**11.44** (27.53)	43.54	39.02 (83.82)	**27.91** (57.66)	68.08	60.98	**46.39**	**82.99**	**76.29**	**60.76**
$\text{MVMR}_{\text{ActivityNet}}$	CET	1.26 (63.68)	0.90 (46.52)	0.56 (27.81)	2.81 (83.23)	2.32 (72.84)	1.75 (58.34)	5.41	4.73	3.87	8.97	8.02	6.56
	BET	1.56 (62.82)	1.16 (44.84)	0.76 (26.73)	3.02 (84.97)	2.55 (74.74)	2.03 (60.27)	5.30	4.64	3.80	8.13	7.10	5.88
	MMN	13.82 (64.72)	10.71 (47.93)	6.93 (29.16)	31.54 (87.10)	25.04 (79.39)	16.55 (64.68)	54.79	44.50	30.62	73.08	62.09	45.76
	RMMN†	16.46 (63.44)	12.68 (47.17)	7.87 (28.64)	36.03 (86.07)	28.47 (77.43)	18.16 (62.06)	57.91	46.52	31.12	73.82	62.12	44.89
	RMMN	**20.63** (64.25)	**15.58** (47.82)	**9.51** (28.52)	**44.13** (86.34)	**34.70** (78.40)	**22.40** (63.45)	**67.01**	**55.58**	**38.55**	**79.91**	**70.02**	**53.20**
$\text{MVMR}_{\text{TACoS}}$	CET	8.95 (37.25)	5.79 (25.0)	3.89	26.18 (60.25)	18.88 (46.50)	12.17	54.84	42.53	23.70	-	-	-
	BET	8.22 (30.37)	5.30 (18.0)	2.97	29.46 (60.96)	14.84 (44.61)	8.32	55.67	39.76	21.51	-	-	-
	MMN	10.56 (39.20)	8.61 (26.27)	5.60	36.16 (62.11)	25.84 (47.42)	15.47	58.20	43.89	23.80	-	-	-
	RMMN†	**13.43** (38.97)	10.75 (27.62)	6.76	36.45 (61.51)	27.01 (47.46)	15.28	58.30	44.38	**25.16**	-	-	-
	RMMN	13.24 (38.44)	**10.75** (27.39)	**7.30**	**39.22** (61.98)	**29.68** (47.31)	**16.50**	**60.05**	**44.77**	24.62	-	-	-

Table 2: **Results on MVMR datasets for RMMN and the baseline models.** Bolded and under-lined results indicate the 1st and 2nd best performance, respectively. RMMN† is the model using only reliable true-negatives filtering (§5.1). RMMN uses both of two reliable negatives filtering method (§5.1 and §5.2). The values in parentheses are the results from the NLVL task. We train all the baselines as three trials and report the averaged NLVL and the MVMR scores.

The figure reveals that queries exhibit well-defined clusters when setting 0.9 threshold for the Charades-STA and TACoS. The figure for the TACoS can also be found in Appendix B.

Figure 4 presents histograms of the SimCSE similarity. The histograms indicate that Charades-STA and TACoS contain numerous text queries with high similarity, supporting the effectiveness of the query similarity filtering.

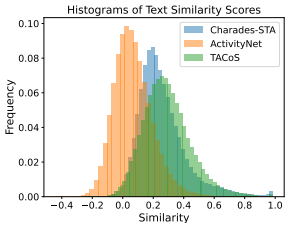


Figure 4: **Histogram of SimCSE Similarity.**

Figure 5 illustrates histograms representing the EMScore similarity between queries and videos that primarily filtered into positives and negatives using SimCSE. The red and blue histograms correspond to the EMScore distribution of positive and negative samples, respectively. For the ActivityNet, EMScore effectively distinguishes between positives and negatives. This success is attributed to that ActivityNet includes detailed queries and videos with diverse features.

According to the findings presented in this section, the filtering method's efficacy varies depending on the dataset's characteristics. Consequently, it is evident that a combination of two filtering methods (§4.1 and §4.2) should be employed in a complementary manner to ensure the construction of a reliable MVMR dataset.

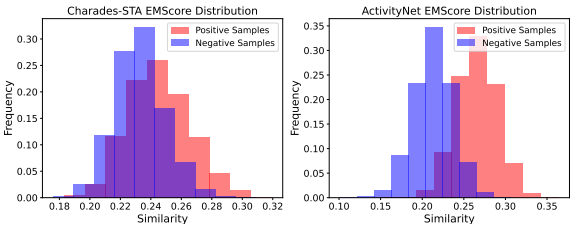


Figure 5: **EMScore Similarity Distribution.**

Dataset Quality. This section verifies the quality by analyzing the three MVMR datasets. Appendix D shows examples of false-negative instances included in the constructed datasets as positive. In the case of Charades-STA dataset, it is evident that the appropriate videos for a target query *"person opens a fridge door"* are identified as positives. We also conduct a human evaluation on our MVMR datasets to reveal that our method build MVMR datasets effectively. We recruit crowd workers and let them examine 100 queries by classifying whether the query and paired video samples are relevant for each dataset. Overall, we analyze 4,900, 4,900, and 400 videos for Charades-STA, ActivityNet, and TACoS, respectively, since each MVMR dataset consists of 49, 49, and 4 newly added positive and negative samples, except for the original golden positive video. From human evaluation, we confirm that the newly introduced dataset contains only 1.5%, 0.2%, and 3.5% falsely categorized videos for the three datasets, respectively. TACoS exhibits a relatively higher query-relevant video ratio than the other two datasets since its test set includes only 25 videos and consists of consid-

erably similar cooking activities within the kitchen.

6.4 Experimental Settings

This section describes the experimental settings of the RMMN and other baselines. More detailed settings of the RMMN and other baselines are shown in Appendix C.

Implementation Details. We use standard off-the-shelf video feature extractors without any fine-tuning: VGG (Simonyan and Zisserman, 2015) feature for Charades-STA; C3D (Tran et al., 2015) feature for ActivityNet and TACoS following (Wang et al., 2022). The reliable true-negatives filtering threshold is set as $t_{tr} = 0.9$. We choose the number of the CM-ANS samples to calculate $p'(q|v)$ as 100, 200, and 100 and $p'(v|q)$ as 50, 100, and 5 for Charades-STA, ActivityNet, and TACoS, respectively. In the hard-negative fine-tuning stage, the learning rates are degraded by multiplying 0.1.

Baselines. We first select the state-of-the-art model, MMN (Wang et al., 2022), as a strong baseline. As the MVMR task can be formulated as an open-domain question answering (QA) task, we further choose two types of QA models that employ a cross-encoder architecture (i.e., Cross-Encoder Transformer) and a dual-encoder architecture (i.e., Bi-Encoder Transformer), respectively. These architectures are widely adopted in existing NLVL tasks (Zhang et al., 2020b; Gao and Xu, 2021) due to the closeness of the two tasks.

Evaluation Metrics. Following the previous setting (Gao et al., 2017) of the NLVL, we evaluate baselines by calculating Rank n@m. It is defined as the percentage of queries having at least one correctly retrieved moment, i.e., IoU $\geq m$, in the top-n derived moments. The existing NLVL studies have reported the results of $m \in \{0.3, 0.5, 0.7\}$ with $n \in \{1, 5\}$ (Wang et al., 2022). But, since our MVMR setting aims to search for massive videos, it is insufficient to analyze the ability of models with previously designed metrics. Hence, we report the extended results as $m \in \{0.3, 0.5, 0.7\}$ with $n \in \{1, 5, 20, 50\}$ for Charades-STA and AcitivityNet and $m \in \{0.3, 0.5, 0.7\}$ with $n \in \{1, 5, 20\}$ for TACoS. We use the same metrics in the previous study (Wang et al., 2022) to report the NLVL scores (parentheses).

6.5 MVMR Results

MVMR Performance. We evaluate the performance of RMMN and the baseline models on the introduced MVMR datasets. Table 2 shows that all the baseline models undergo significant performance degradation in the MVMR task. Surprisingly, although CET and BET show quite strong performance on the NLVL task, their performance on the MVMR task notably drops. These outcomes indicate that the current NLVL task performance alone is not enough to verify that a model is well-trained and that negative sampling methods significantly contribute to the performance of the MVMR. We reveal that RMMN outperforms all other baselines on MVMR task, and this suggests that our model correctly discriminates features of negative and positive videos. Experiments show that the models on Charades-STA and TACoS significantly improve performance from the true-negative filter since similar query overlap frequently exists in these datasets. However, the performance of ActivityNet is attributed to the hard-negative filter since it includes various and complex text queries.

MVMR Settings with Video Retrieval. The practical usage scenario for the MVMR should include video retrieval. Therefore, We also conduct MVMR experiments with a video retrieval model, PRMR model (Dong et al., 2022), shown as Table 3. We construct a pipeline to filter top-5 logit videos from our MVMR retrieval pool using the PRMR model and solve MVMR task. We use the publicly deployed PRMR model on Charades-STA and ActivityNet datasets. These experiments prove that video retrieval models are helpful to increase MVMR performance, and RMMN still shows superior performance than MMN in this pipeline.

Model	R@1 IOU0.3	R@1 IOU0.5	R@5 IOU0.3	R@5 IOU0.5
$\text{MMN}_{\text{Charades-STA}}$	32.15	27.10	65.97	57.66
$\text{RMMN}_{\text{Charades-STA}}$	**36.64**	**29.27**	**68.06**	**59.78**
$\text{MMN}_{\text{ActivityNet}}$	32.39	25.04	66.84	55.03
$\text{RMMN}_{\text{ActivityNet}}$	**39.25**	**29.59**	**71.54**	**60.02**

Table 3: **MVMR Experiments with Video Retrieval.**

7 Conclusion

In this paper, we propose the Massive Videos Moment Retrieval (MVMR) task, which aims to detect a moment for a natural language query from a massive video set. To stimulate the research, we propose a novel method to construct reliable MVMR datasets and build three MVMR datasets using existing NLVL datasets. We further introduce a robust training method called Reliable Mutual Matching to effectively distinguish positives from negative distractors when solving MVMR task.

Limitations

In this work, we introduce a novel model called RMMN, which effectively filters reliable and informative negatives and incorporates them into the contrastive learning framework. Our experimental results primarily focused on enhancing MVMR performance and demonstrated comparable results on the NLVL task without significantly outperforming existing baselines. However, these outcomes indicate that the current NLVL task performance alone is inadequate to verify that a model is well-trained.

References

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2017. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812.

Jianfeng Dong, Xianke Chen, Minsong Zhang, Xun Yang, Shujie Chen, Xirong Li, and Xun Wang. 2022. Partially relevant video retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 246–257.

Victor Escorcia, Mattia Soldan, Josef Sivic, Bernard Ghanem, and Bryan Russell. 2019. Temporal localization of moments in video collections with natural language.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. TALL: temporal activity localization via language query. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5277–5285. IEEE Computer Society.

Junyu Gao and Changsheng Xu. 2021. Fast video moment retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1523–1532.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann. 2019. Excl: Extractive clip localization using natural language descriptions.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 961–970. IEEE Computer Society.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans.

Minjoon Jung, Seongho Choi, Joochan Kim, Jin-Hwa Kim, and Byoung-Tak Zhang. 2022. Modal-specific pseudo query generation for video corpus moment retrieval. *arXiv preprint arXiv:2210.12617*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 706–715. IEEE Computer Society.

Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. 2020. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 447–463. Springer.

Daizong Liu, Xiaoye Qu, Jianfeng Dong, Pan Zhou, Yu Cheng, Wei Wei, Zichuan Xu, and Yulai Xie. 2021. Context-aware biaffine localizing network for temporal sentence grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11235–11244.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Zhixin Ma and Chong Wah Ngo. 2022. Interactive video corpus moment retrieval using reinforcement learning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 296–306.

Guoshun Nan, Rui Qiao, Yao Xiao, Jun Liu, Sicong Leng, Hao Zhang, and Wei Lu. 2021. Interventional video grounding with dual contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2765–2775.

Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2018. Bidirectional attention flow for machine comprehension.

Yaya Shi, Xu Yang, Haiyang Xu, Chunfeng Yuan, Bing Li, Weiming Hu, and Zheng-Jun Zha. 2022. Emscore: Evaluating video captioning via coarse-grained and fine-grained embedding matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17929–17938.

Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 4489–4497. IEEE Computer Society.

Zhenzhi Wang, Limin Wang, Tao Wu, Tianhao Li, and Gangshan Wu. 2022. Negative sample matters: A renaissance of metric learning for temporal grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2613–2623.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

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

Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. 2020b. Learning 2d temporal adjacent networks for moment localization with natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 12870–12877. AAAI Press.

Kun Zhou, Yeyun Gong, Xiao Liu, Wayne Xin Zhao, Yelong Shen, Anlei Dong, Jingwen Lu, Rangan Majumder, Ji-Rong Wen, Nan Duan, and Weizhu Chen. 2022. Simans: Simple ambiguous negatives sampling for dense text retrieval.

A Datasets Details

Charades-STA is an extended dataset of action recognition and localization dataset Charades (Sigurdsson et al., 2016) by (Gao et al., 2017) for video moment retrieval. It is comprised of 5,338 videos and 12,408 query-moment pairs in the training set, and 1,334 videos and 3,720 query-moment pairs in the test set. We report evaluation results on test set in our experiments.

ActivityNet-Captions is constructed on ActivityNet v1.3 dataset (Heilbron et al., 2015), where included videos cover various complex human actions. It is originally designed for video captioning, and recently used into video moment retrieval. It contains 37,417, 17,505, and 17,031 query-moment pairs for training, validation, and testing respectively. We report the evaluation result on val_2 set following the setting of the MMN paper (Wang et al., 2022).

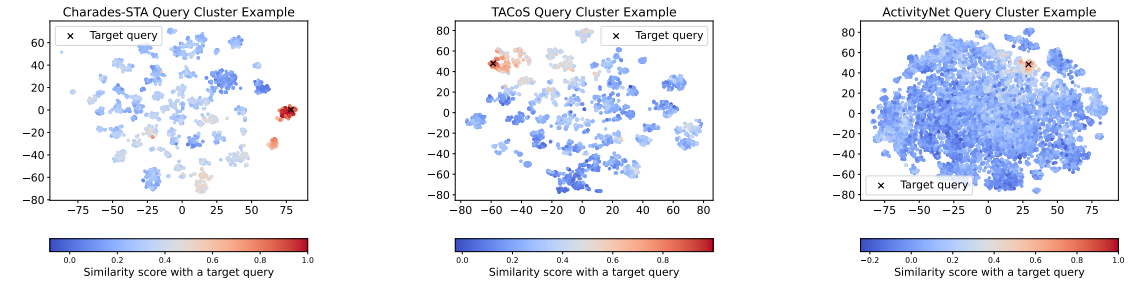
TACoS includes 127 videos selected from the MPII-Cooking dataset. It consists of 18,818 query-moment pairs of various cooking activities in the kitchen annotated by (Regneri et al., 2013). It (Gao et al., 2017) consists of 10,146, 4,589, and 4,083 query-moment pairs for training, validation and testing, respectively. We report evaluation results on test set in our experiments.

B Analysis on Filter Methods

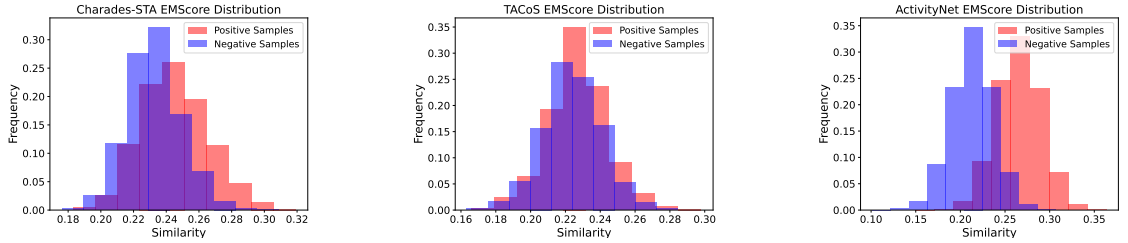
We construct reliable MVMR datasets using SimCSE and EMScore. The visualizations of the similarity scores derived by SimCSE and EMScore is shown as Figure 6. For the SimCSE distribution, we select target queries of *"person turn a light on"*, *"The person peels a kiwi"*, and *"He is using a push mower to mow the grass"* for the Charades-STA, TACoS, and ActivityNet, respectively. The detailed qualitative analysis of SimCSE filtering is shown in Table 4.

C Implementation Details

We use standard off-the-shelf video feature extractors without any fine-tuning. We use VGG (Simonyan and Zisserman, 2015) feature for Charades-STA and C3D (Tran et al., 2015) feature for ActivityNet and TACoS following the previous work (Wang et al., 2022). We train RMMN and MMN on 1 NVIDIA RTX A6000 GPU and early stop by averaging scores of all evaluation metrics.



(a) SimCSE Distribution



(b) EMScore Distribution

Figure 6: **The Visulization of the SimCSE and EMScore Similarity Distribution.**

C.1 RMMN

We implement RMMN by following MMN's model architecture. RMMN represents each query and video moment using query and video encoders to a joint visual-text space. From the joint visual-text space, RMMN computes the similarity between the query and video moment representations and selects the best-matched video moment.

Query and Video Encoders. We choose DistilBERT (Sanh et al., 2019) for the text query encoder following MMN since it is a light-weighted efficient model showing significant performance. We calculate the representation of each query $f^q \in \mathbb{R}^d$ using the global average pooling over all token. We adopt the approach of encoding the input video as a 2D temporal moment feature map, inspired by 2D-TAN (Zhang et al., 2020b) and MMN. To achieve this, we segment the input video into video clips denoted as $\{c_i\}_{i=1}^{l_c/k}$, where each clip c_i consists of k frames. l_c means the total number of frames in the video. The clip-level representations are extracted using a pre-trained visual model (e.g., C3D). By sampling a fixed-length with a stride of $\frac{l_c}{k \cdot N}$, we obtain N clip-level features. These features are then passed through an FC layer to reduce the dimensionality, resulting in the features $\{f_i^v\}_{i=1}^{N}$, where $f_i^v \in \mathbb{R}^d$. Utilizing these features, we construct a 2D temporal moment feature map $F \in \mathbb{R}^{N \times N \times d}$ by employing max-pooling as the moment-level feature aggregation method following the baseline (Wang et al., 2022). Additionally, we generate a 2D feature map F' of the same size by passing F to 2D Convolution, allowing the representation of moment relations as employed in MMN.

Joint Visual-Text Space. Initially, we apply layer normalization to both the video moments and text features. Subsequently, we utilize a linear projection layer and a 1×1 convolution layer to project the text and video in the same embedding space, respectively. The projected features are then employed in two distinct representational spaces: the IoU space and the mutual matching space. These spaces serve as the basis for computing the binary cross-entropy loss and the contrastive learning loss, respectively.

Hyperparameter Settings. Our convolution network for deriving 2D feauture is exactly same as MMN, including the number of sampled clips N, number of 2D conv layers L, kernel size and channels. We set the dimension of the joint feature visual-text space $d = 256$, and temperatures $\tau_m = \tau_q = 0.1$. We utilize the pre-trained HuggingFace implementation of DistilBERT (Wolf et al., 2019). We set margin ψ as 0.4, 0.3, and 0.1 for Charades-STA, ActivityNet, and TACoS, respectively. We use AdamW (Loshchilov and Hutter, 2017) optimizer with learning rate of 1×10^{-4}, 8×10^{-4}, 1.5×10^{-3} for Charades-STA, Activi-

tyNet, and TACoS, respectively. We set λ as 0.05 for Charades-STA and TACoS and 0.1 for ActivityNet. We set the reliable true-negatives filtering threshold $t_{tr} = 0.9$. We set the number of hard-negative samples to calculate $p'(q|v)$ as 100, 200, and 100 and $p'(v|q)$ as 50, 100, and 5 for Charades-STA, ActivityNet, and TACoS, respectively. The experimental results for various hyper-parameters for the hard-negative sampling are shown in Figure 7.

C.2 Baselines

MMN (Wang et al., 2022) adopts a dual-modal encoding design to get video clip and text representations. We utilized the publicly deployed code[2] to implement our own MMN model for our experiment using exactly same hyper-parameters setting.

Cross-Encoder Transformer (CET) is a cross-modal encoding model implemented based on 6-layered transformer architecture. It is designed to predict the start and end points of a video moment by utilizing a model architecture which is used in the QA task. We concatenate the pre-extracted video clip features and the text features derived from the Distil-BERT to use as an input of the transformer. To concatenate two extracted features, we should equalize the dimensions of two features with a linear layer as follows:

$$\begin{aligned} \bar{H}^v &= H^v W^v \\ \bar{H}^q &= H^q W^q \end{aligned} \tag{11}$$

where $H^v \in \mathbb{R}^{l_v, d^v}$, $H^q \in \mathbb{R}^{l_q, d^q}$, $W^v \in \mathbb{R}^{d^v, d}$, $W^q \in \mathbb{R}^{d^q, d}$. H^v is the pre-extracted video clip features and H^q is the features extracted using Distil-BERT. And then, we derive the representation of each video clip feature from the last layer of the transformer-encoder as follows:

$$H^{v,q} = TransformerEnc(\bar{H}^v \oplus \bar{H}^q) \tag{12}$$

where $H^{v,q} \in \mathbb{R}^{l_{vq}, d}$, $l_{vq} = l_v + l_q$. We utilize only the representation part of video features and they are followed by the 3-layered MLPs and Sigmoid to predict the start and end positions of moment, respectively, as follows:

$$\begin{aligned} p^s &= Sigmoid(MLP_s(H^{v,q}_{1:l_v})) \\ p^e &= Sigmoid(MLP_e(H^{v,q}_{1:l_v})) \end{aligned} \tag{13}$$

[2]https://github.com/MCG-NJU/MMN

where p^s and $p^e \in \mathbb{R}^{l_v}$. We also introduce the interpolation labels y^s_i, y^e_i to utilize abundant information of moment labels as follows:

$$\begin{aligned} y^s_i &= \begin{cases} (\frac{x_e - i}{x^*_e - x^*_s + 1})^k & \text{if } x^*_s \le i \le x^*_e \\ 0 & \text{otherwise} \end{cases} \\ y^e_i &= \begin{cases} (\frac{i - x_s}{x^*_e - x^*_s + 1})^k & \text{if } x^*_s \le i \le x^*_e \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{14}$$

We finally use the binary cross-entropy to calculate a loss for the pair of (p^s_i, y^s_i) and (p^e_i, y^e_i) as follows:

$$\begin{aligned} \mathcal{L}^s &= BCE(p^s, y^s) \\ \mathcal{L}^e &= BCE(p^e, y^e) \end{aligned} \tag{15}$$

where BCE means the binary cross-entropy. The final loss to train the CET model is defined as follows:

$$\mathcal{L} = \frac{\mathcal{L}^s + \mathcal{L}^e}{2} \tag{16}$$

We use $k = 10$ and $d = 512$ as a hyperparameter and AdamW optimizer with learning rate 1e-3.

Bi-Encoder Transformer (BET) is a dual-modal encoding model which encodes text and video clip features independently. The pre-extracted video clip features are encoded using 6-layered transformer and text features are encoded using the Distil-BERT followed by average-pooling function and 3-layered MLPs.

$$\tilde{H}^v = TransformerEnc(\bar{H}^v) \tag{17}$$

$$\begin{aligned} \tilde{H}^s &= MLP_s(Pool(\bar{H}^q)) \\ \tilde{H}^e &= MLP_e(Pool(\bar{H}^q)) \end{aligned} \tag{18}$$

Where $\tilde{H}^v \in \mathbb{R}^d$, $\tilde{H}^s \in \mathbb{R}^d$ and $\tilde{H}^e \in \mathbb{R}^d$. $Pool$ means the average-pooling. We calculate the cosine similarity between the encoded video clip representations and the encoded start and end representations to predict a moment.

$$\begin{aligned} p^s &= Sigmoid(r * cosine(\tilde{H}^v, \tilde{H}^s)) \\ p^e &= Sigmoid(r * cosine(\tilde{H}^v, \tilde{H}^e)) \end{aligned} \tag{19}$$

We use the interpolation labels and the binary cross-entropy loss for the BET, similar to CET. We use $r = 10$ and $d = 512$ as a hyperparameter and AdamW optimizer with learning rate 1e-3.

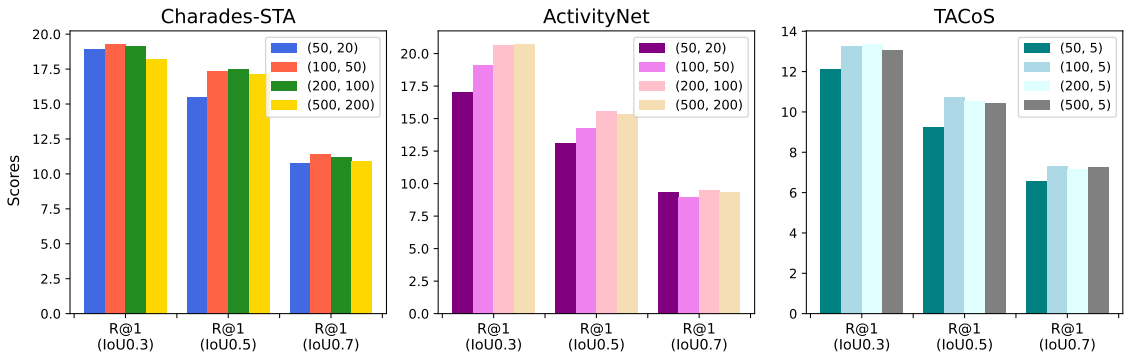


Figure 7: **Evaluation results for various hyper-parameters.** We search various hyper-parameters for the hard negative sampling. We measure R@1 (IoU>m), where m$\in \{0.3, 0.5, 0.7\}$ to find the best hyper-parameters. In this figure, each (x, y) means the number of negative samples. x and y mean a number of negative samples for $p'(v|q)$ (query-to-video) and $p'(q|v)$ (query-to-video), respectively. Consequently, we select the number of the CM-ANS samples to calculate $p'(q|v)$ as 100, 200, and 100 and $p'(v|q)$ as 50, 100, and 5 for Charades-STA, ActivityNet, and TACoS, respectively.

Dataset	Query #1	Query #2	SimCSE Similarity
Charades-STA	person turn a light on.	a person turns on some lights.	0.9152
	person turn a light on.	a person is fixing a light on the room.	0.7817
	person turn a light on.	person opens a window.	0.5028
	person turn a light on.	a person walks into a bathroom holding dishes.	0.3716
	person turn a light on.	person sits in a chair.	0.2211
ActivityNet	The woman throws water with the hose to the female who grabs the hose.	After, the female throws water with the hose to the woman.	0.9646
	A woman brushes and styles her hair.	He washes her hair, then blow dries it.	0.7616
	person turn a light on.	person opens a window.	0.5028
	Two men both dressed in athletic gear are standing and talking in an indoor weight lifting gym filled with other equipment.	One man is holding onto a rope attached to a machine, and the other man instructs him to bend down on his left knee while still holding onto the rope and he showing the man how to have proper form.	0.3651
	Two men both dressed in athletic gear are standing and talking in an indoor weight lifting gym filled with other equipment.	The man in front begins to do karate on the mat.	0.1817
TACoS	She took out knife	She takes out knife	0.9166
	She took out kiwi	She peeled kiwi	0.7483
	She took out kiwi	The person peels the second kiwi	0.5111
	She took out kiwi	She took out plate	0.3391
	She took out kiwi	The person throws away the peels	0.2093

Table 4: **SimCSE similarity scores analysis.**

D Examples of False Negative Videos

Illustrative instances of constructed MVMR datasets can be found in Figure 8. This figure shows video instances that have been identified as similar to a specific target query. The target queries correspond to queries of Charades-STA, ActivityNet, and TACoS datasets, sequentially.

E Human Evaluation for the constructed MVMR Datasets

We conduct a human evaluation on our constructed MVMR datasets to reveal that our method construct MVMR datasets effectively. We recruit crowd workers fluent in English through the university's online community. The recruited annotators were provided with a detailed description of task definitions, instructions. An example of a sheet we use to conduct a human evaluation is shown in Figure 9.

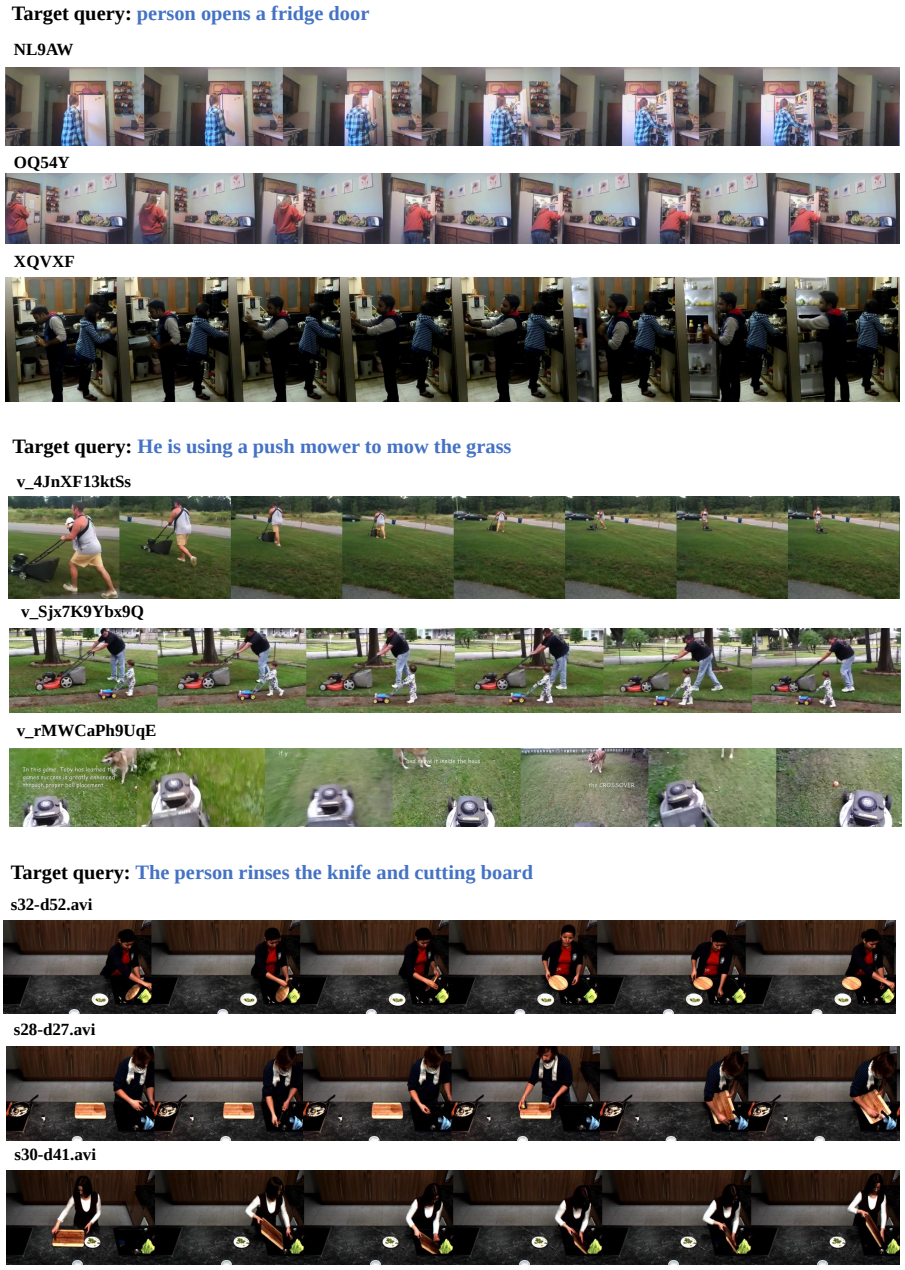


Figure 8: **Examples of false-negative (positive) videos derived using our methods.** Each query corresponds to that of Charades-STA, ActivityNet, and TACoS, sequentially.

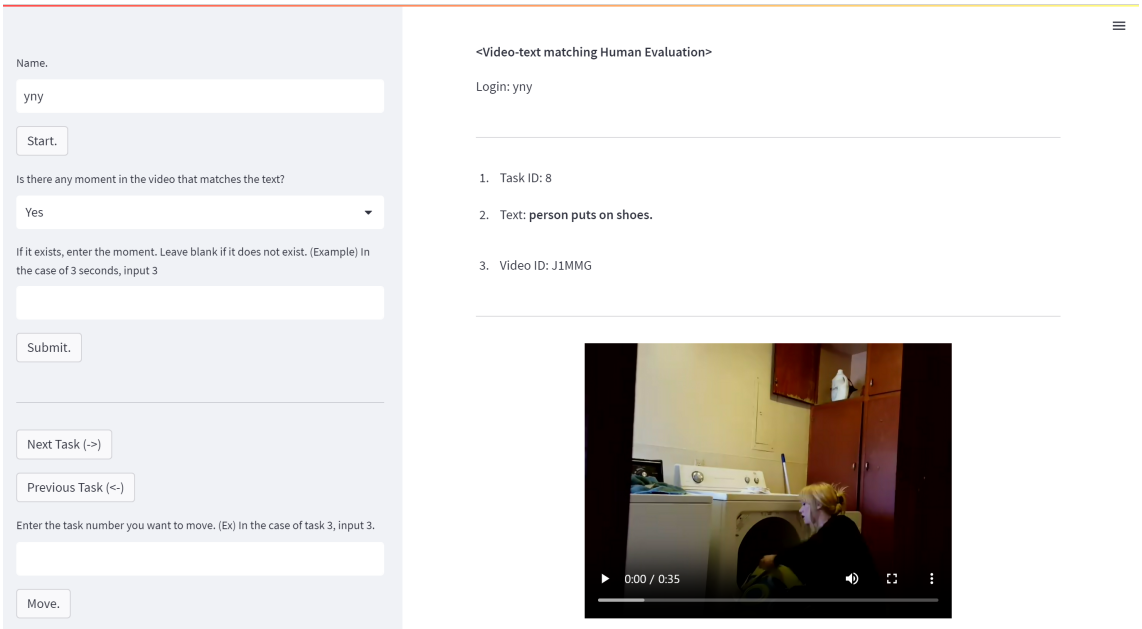


Figure 9: **Human Evaluation Sheet Example.**