Multi-Query Video Retrieval

Zeyu Wang, Yu Wu, Karthik Narasimhan, and Olga Russakovsky

Princeton University
{zeyuwang,yuwu,karthikn,olgarus}@cs.princeton.edu

Abstract. Retrieving target videos based on text descriptions is a task of great practical value and has received increasing attention over the past few years. Despite recent progress, imperfect annotations in existing video retrieval datasets have posed significant challenges on model evaluation and development. In this paper, we tackle this issue by focusing on the less-studied setting of multi-query video retrieval, where multiple descriptions are provided to the model for searching over the video archive. We first show that multi-query retrieval task effectively mitigates the dataset noise introduced by imperfect annotations and better correlates with human judgement on evaluating retrieval abilities of current models. We then investigate several methods which leverage multiple queries at training time, and demonstrate that the multi-query inspired training can lead to superior performance and better generalization. We hope further investigation in this direction can bring new insights on building systems that perform better in real-world video retrieval applications.[1]

Keywords: Video retrieval · Multi-query · Evaluation

1 Introduction

With the vast amount of videos available online and new ones being generated and uploaded everyday, an efficient video search engine is of great practical value. The core of a video search engine is the retrieval task, which deals with finding the best matching videos based on the user's text query [39,37,33,16,10]. With collective efforts over the years, current state-of-the-art methods [3,36,17] have achieved reasonable performance on several different video retrieval benchmarks [55,9,4,28,43].

Despite these advances, existing works focus mostly on experimenting under the single-query setting, *i.e.,* retrieving target videos given a *single* text description as input. A major issue with this paradigm is the problem of under-specification. As most existing video retrieval benchmarks are based on datasets collected for other tasks (*e.g.,* video captioning) [55,50,9,28,43], lots of text descriptions are of low-quality and unsuited for video retrieval. Figure 1 top shows some examples of noisy annotations from one of the most widely-used video retrieval datasets MSR-VTT [55]. One can notice that these annotations are very

---

[1] Code is available at `https://github.com/princetonvisualai/MQVR`.
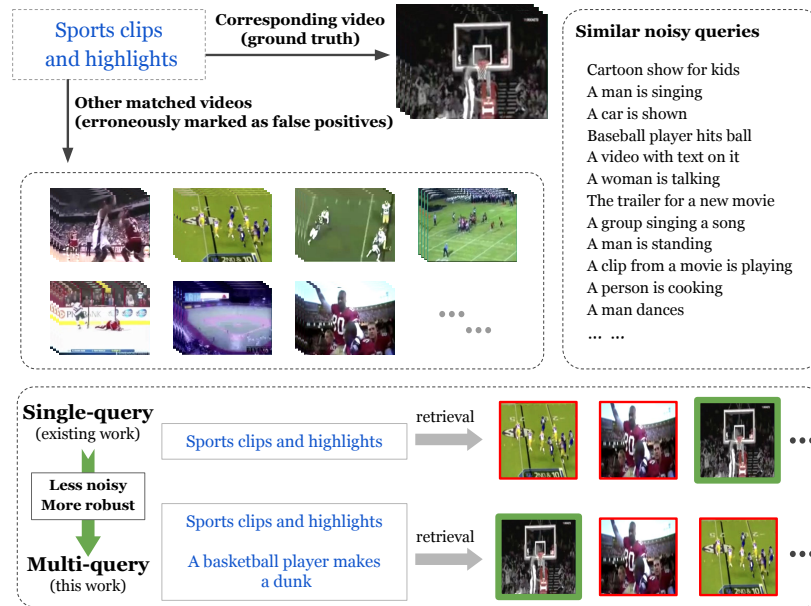
Fig. 1: Existing video retrieval works focus on the single-query setting, where a single text description is input for retrieving target video. However, due to imperfect annotations in the datasets (too general descriptions that match many different videos, shown on top), retrieval model may be unfairly penalized for not finding the single target video. To counteract this under-specification issue, in this work we tackle the previously less-studied *multi-query* setting (bottom), which provides a less noisy benchmarking of retrieval abilities of current models.

general and can be perfectly matched to lots of videos. This can cause trouble during evaluation as all matched videos would be given similar matching scores and essentially ranked randomly, thus resulting in a wrong estimate of the model's true performance. One way to solve such problem potentially is to build better datasets with cleaner annotations. However, it is extremely hard and laborious to build a "perfect" dataset and many of the most widely used ones are all with different imperfections [13,1]. Especially, latest models are more and more often being trained with large-scale web-scraped data [42,24], which inevitably contains various kind of noise.

In this work, we propose the task of *multi-query video retrieval* (MQVR) where multiple text queries are provided to the model simultaneously (Figure 1 bottom). It addresses the noise effects introduced by imperfect annotations in a simple and effective way. Intuitively, it is less likely that there are many videos in the dataset *all* matching *all* the provided queries. As a result, MQVR can produce a more accurate evaluation of the model's performance compared to a single-query retrieval benchmark. We verify this intuition empirically by inspect-
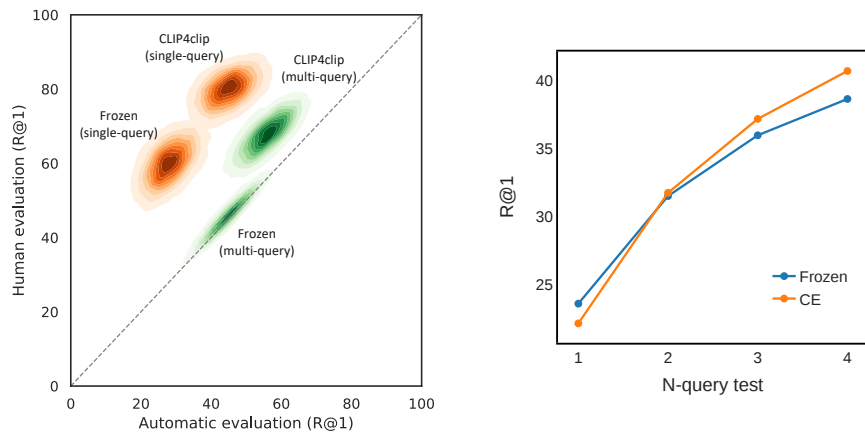
Fig. 2: **Left**: Density plot of human evaluation vs. automatic evaluation for two video retrieval models [30,8] on MSR-VTT [55]. Single-query retrieval is done by selecting the video with the highest similarity with the input query; for multi-query retrieval, the average similarity with all queries is used. Human evaluation (obtained by manual inspection of the top retrieved video) is compared to automatic evaluation (a binary indicator of whether the input queries are the corresponding annotations in the dataset for the top retrieved video). The results are computed on 100 retrieval samples, and the the plot is generated with Gaussian kernel density estimation on 10,000 samples generated with bootstrapping. Multi-query automatic evaluation shows significantly better correlation with human judgement compared to the single-query counterpart. **Right**: Relative performance of models [33,8] could change when evaluated with a different number of queries, suggesting that multi-query setting may reveal important insights about the retrieval potential of different models. The two models depicted are collaborative experts (CE) [33] and Frozen [8]; both models are trained from scratch on MSR-VTT [55] for a fair comparison.

ing 100 retrieval results, and manually labeling whether the top retrieved video matches the provided query or queries. As shown in Figure 2 left, single-query evaluation has a large gap with human judgement: automated methods unfairly penalize models for retrieving videos which actually match the provided query (as per human assessment) but are not the single video intended to be retrieved by the benchmark. Multi-query evaluation significantly reduces the gap by making the retrieval task less under-specified, so the model can better retrieve the target video (higher automatic evaluation score) and by making it less likely for other videos to match input queries. Going further, in Figure 2 right we show that this has practical consequences as the relative ordering of two retrieval models changes under the single-query vs. multi-query evaluation settings.

Equipped with these insights, we re-purpose existing video retrieval datasets and perform an extensive investigation of state-of-the-art models adapted to-

wards handling multiple queries. We first experiment with different ways of using multiple queries only during inference, and then explore ways of learning retrieval models explicitly with multiple queries provided at training time. Our experiments over three different datasets demonstrate that MQVR more optimally utilizes the capabilities of modern retrieval systems, and can provide new insights for building better retrieval models. Especially, we find that a multi-query inspired training can lead to representations with superior performance and better generalization.

To summarize, we make the following contributions:

- We extensively investigate multi-query video retrieval (MQVR) – where multiple text queries are provided to the model – over multiple video retrieval datasets (MSR-VTT [55], MSVD [9], VATEX [50]).
- We find that dedicated multi-query training methods can provide large gains (up to 21% in R@1) over simply treating each query independently and combining their similarity scores. We also investigate several architectural changes such as localized and contextualized weighting mechanisms which further improve MQVR performance.
- To facilitate future research in this area, we further propose a new metric based on the area under the curve for MQVR with varying number of queries, which is complementary to standard metrics used in single-query retrieval.
- Finally, we also demonstrate that multi-query training methods can be utilized to benefit standard single-query retrieval, and that the multi-query trained representations have better generalization than the single-query trained counterpart.

2   Related work

Video-text representation learning. The multi-modal learning for videos [45,7,40,54,56,5,14,53,19,2] has been receiving increasing attention in recent years. Among them, learning to generate better video and language representation [57,19,59,48] is one of the most attractive topics. Recent works leverage large-scale vision-language datasets for model pretraining [37,8,31,27]. Howto100M [38] is a large-scale video-text pretraining dataset, which takes texts obtained from narrated videos using automatic speech recognition (ASR) as the supervision for video-text representation learning. Miech *et al.* [37] then proposed a multiple instance learning approach derived from noise contrastive estimation (MIL-NCE) to learn from these noisy instructional video-text pairs. Lei *et al.* [29] proposed ClipBERT which leverages sparse sampling to train the model in an end-to-end approach. Recently, CLIP (Contrastive Language-Image Pre-training) [42] has shown great success by pre-training on large-scale webly-supervised image and text pairs. The learned image and language representations are proved to be helpful in most downstream vision-language tasks.

Text-to-video retrieval. To search videos by unconstrained text input, both video and text modalities should be embedded into a shared representation space for further similarity matching. Early works [39,37,33,20,49] used

pre-trained models to extract representations from multi-modal data (RGB, motion, audio). Mithun *et al.* [39] proposed to use hard negative mining to improve the training for text and video joint modeling. Liu *et al.* [33] leverage seven modalities (*e.g.,* speech content, scene texts, faces) to build the video representation. Miech *et al.* [37] further proposed a strong joint embedding framework based on mixture-of-expert features. Gabeur *et al.* [20] introduce multi-modal transformer to jointly encode those different modalities with attentions. Recently, there are some works [36,17,30,21] based on the large-scale pretrained vision-language models (*e.g.,* CLIP [42]), which achieve superior performance compared to previous training-from-scratch models. However, these works are evaluated using single-query benchmarks, suffering from the query quality issue. Differently, we study the multi-query video retrieval in this paper, where the input is a set of multiple queries.

Multi-query retrieval. Many previous studies on multi-query retrieval are in the uni-modal domain, such as content-based image retrieval [47,18,25], landmark retrieval [52,51], or person retrieval using multiple images or video frames [58,32]. These methods take multiple images as input queries where each image is obtained from different angles, distances, conditions of the same objects or scenes [47,52]. Instead of getting multiple queries from the user, works on query expansion [12,11,26,6,22,41,23] take a single query as input and further extend it to multiple queries by pre-processing (*e.g.,* synonym replacement) or using relevant candidates produced during an initial ranking. In this work, we utilize multi-query in the task of text-to-video retrieval to address challenges brought by imperfect annotations in the datasets.

3   Multi-query video retrieval

In this section, we first formally define the setting of MQVR, then delineate the various methods experimented to handle multiple queries, and finally introduce a new evaluation metric for MQVR.

3.1   Setting

Video retrieval[2] is a task of searching videos given text descriptions [39,37,33]. Formally, given a video database $\mathcal{V}$ composed of n videos, $\mathcal{V} = \{v_1, v_2, ..., v_n\}$, and a text description q_i of video v_i, the goal is to successfully retrieve v_i from $\mathcal{V}$ based on q_i. This is the setting most existing video retrieval benchmarks adopt [55,4,28,43]. We refer to it as single-query setting, since one caption is used for retrieval during evaluation. Specifically, the goal is to learn a model $\mathcal{M}^{single}$ that can evaluate the similarity between any query-video pair, such that $\mathcal{M}^{single}(q, v) \in \mathbb{R}$ reflects how the query matches with the video. And a

---

[2] Throughout this paper, we use the term 'video retrieval' to refer the specific task of text-to-video retrieval.

perfect retrieval would score the matching pair higher than non-matching pairs,

$$\forall j, j \neq i, \quad \mathcal{M}^{single}(q_i, v_i) > \mathcal{M}^{single}(q_i, v_j). \tag{1}$$

However, this single-query setting might be problematic when the input description has low quality, for example, being too general or abstract that can perfectly describe different videos in the database. This can pose serious trouble during evaluation. Due to the fact that existing video retrieval benchmarks are mostly based on datasets collected for other tasks (*e.g.,* video captioning) [55,50,9,28,43], such low-quality queries are prevalent during model evaluation (as shown in Figure 1).

To tackle this issue, we study the multi-query retrieval setting where more than one query is available during retrieval (Figure 1 bottom). Under similar notation, the task of multi-query retrieval is to retrieve a target video v_i from the video database $\mathcal{V}$ based on multiple descriptions $Q_i = \{q_i^1, q_i^2, ..., q_i^k\}$ of it. And similarly we would like to learn a model $\mathcal{M}^{multi}$ that correctly retrieves the target video,

$$\forall j, j \neq i, \quad \mathcal{M}^{multi}(Q_i, v_i) > \mathcal{M}^{multi}(Q_i, v_j). \tag{2}$$

3.2   Methods

Broadly, we divide the methods into two categories: the first category of methods extends the models trained with single-query to multi-query evaluation in a post-hoc fashion without any retraining, and the second one has dedicated modifications for multi-query retrieval during training.

Post-hoc inference methods. The models trained using single-query can be easily adapted when multiple queries are available, just by considering multiple queries separately and then aggregating the results.

Similarity aggregation (SA). A simple form is to take the mean of the similarity scores evaluated between each query and the video as the final multi-query similarity, which is then used to rank videos.

$$\mathcal{M}_{SA}^{multi}(Q, v) = \frac{1}{k} \sum_{i=1}^{k} \mathcal{M}^{single}(q_i, v). \tag{3}$$

Rank aggregation (RA). Unlike *similarity aggregation* which aggregates the raw similarity score of each query, *rank aggregation* aggregates the retrieval results instead. Denote $\mathcal{R}(v|q, \mathcal{M}^{single}) \in \{1, 2, ..., n\}$ as the rank of v among all videos in the candidate pool based on the evaluation of similarity to q according to $\mathcal{M}^{single}$ (smaller rank means more compatible with the query), then the multi-query similarity can be calculated as:

$$\mathcal{M}_{RA}^{multi}(Q, v) = -\frac{1}{k} \sum_{i=1}^{k} \mathcal{R}(v|q, \mathcal{M}^{single}). \tag{4}$$

Note that the overall similarity score here doesn't have a well-defined quantitative meaning as in the **SA** case, and just serves to order different videos.

Multi-query training methods. Unlike post-hoc methods that essentially deal with the multi-query problem in a "single-query way", dedicated training modifications might be helpful if multiple queries are available during the training phase. This is already provided by many standard benchmarks [55,9,50], but to the best of the authors' knowledge, most existing works only treat the descriptions independently and adopt the single-query training, where a training text-video pair is composed of a video and one description, *i.e.,* (q_i, v_i) for a positive sample, (q_i, v_j) for a negative sample. However, as we will show later, this is not the best choice and that multi-query training, *i.e.,* using (Q_i, v_i) as a positive pair and (Q_i, v_j) as a negative pair, where $Q_i = \{q_i^1, q_i^2, ..., q_i^k\}$ contains more than one descriptions, can provide a large gain.

To facilitate the discussion, we first denote the similarity score of a query-video pair as $\mathcal{M}^{single}(q, v) = \mathbf{S}\left(f(q), g(v)\right)$. Here f denotes the feature extraction network of query sentence which maps the input query to a high dimensional vector $f(q) \in \mathbb{R}^m$. Similarly, g, the video feature extractor, maps input video to the same embedding space $g(v) \in \mathbb{R}^m$. $\mathbf{S}$ is the metric that measures the similarity between $f(q)$ and $g(v)$ (*e.g.,* the cosine similarity).

Mean feature (MF). A naive way of combining multiple queries during training is to take the mean of their features to be the final query feature, with corresponding similarity score calculated as:

$$\mathcal{M}_{MF}^{multi}(Q, v) = \mathbf{S}\left(\frac{1}{k}\sum_{i=1}^{k} f(q_i),\ g(v)\right).$$ (5)

We will show in the next section that just by making this simple modification, the trained model can outperform the post-hoc methods by a large margin. However, a potential drawback of *mean feature* is that each query contributes equally to the result regardless of their quality. Thus we make a natural extension and further investigate **weighted feature (WF)** methods,

$$\mathcal{M}_{WF}^{multi}(Q, v) = \mathbf{S}\left(\sum_{i=1}^{k} \alpha_i f(q_i),\ g(v)\right),$$ (6)

where $\sum_{i=1}^{k} \alpha_i = 1$. We experiment with several ways to generate the weights α.

Text-to-text similarity weighting (TS-WF). It is desirable for a query to contain complementary information about the target video that is not captured by other queries. On contrary, a query is not informative if it only contains redundant information already captured by others. This inspires a parameter-free method to evaluate the informativeness of one query by comparing the similarity of it to other queries. Specifically, the informativeness of query q_i is

computed as $\mathbf{I}(q_i) = -\sum_{j \neq i} \mathbf{S}(f(q_i), f(q_j))$, where $q_i, q_j \in Q$. Notice the minus sign here means the informative queries should be different from others. Finally, the weights can be computed by taking softmax among $\mathbf{I}(q)$s.

Localized weight generation (LG-WF). The weights can also be learned using a separate network and trained end-to-end with other parameters. We experiment with a multi-layer perceptron (MLP) that maps extracted text features to a scalar. The MLP is shared across all queries and each query is processed separately. The resulting weights are normalized with softmax.

Contextualized weight generation (CG-WF). Instead of computing weight for every query individually, *CG-WF* attends to other queries when generating each weight. Specifically, we experiment with a transformer-based attention network, where all query features are first input to a transformer to generate contextualized features, and then a MLP head is used to map the contextualized features to scalars. Softmax is used to compute the final normalized weights like in the *localized weight generation* case.

3.3   Evaluation

We adopt the evaluation metrics widely used in the standard single-query retrieval setting and compute the standard R@K (recall at rank K, higher the better), median rank and mean rank (lower the better) in our experiments. Additionally, specific to multi-query evaluation, we would also like to compare models tested under varying number of query inputs (*e.g.,* Figure 5 in the experiments) for a more robust evaluation of the models' generalization ability. To facilitate future research in this subject, we propose an area under the curve (AUC) metric. Specifically, $\mathbf{AUC}_n^{R@K}$ is defined as the normalized area under the curve value of R@K with test query-input varying from 1 to n,

$$\mathbf{AUC}_n^{R@K} = \frac{\mathbf{AUC}([R@K_1, R@K_2, ...R@K_n])}{n - 1}, \tag{7}$$

where $R@K_m$ is the R@K value when evaluating with m queries as input.

4   Experiments

We first describe in section 4.1 the architecture backbones on which we build our multi-query retrieval experiment. Then specific experiment settings and implementation details are described in section 4.2 and finally the experiment results are shown in section 4.3.

4.1   Architecture backbones

CLIP4Clip *[36].* CLIP [42] has been shown to learn transferable features to benefit lots of downstream tasks including video action recognition, OCR, etc.

Recently, such trend has also been demonstrated for video retrieval. Among multiple models proposed in this line of work [3,36,17], we adopt the CLIP4Clip for its simplicity. For our experiment, we use the publicly released "ViT-B/32" checkpoint for initialization of the CLIP model.

***Frozen** [8].* This model proposed by Bain *et al.* utilizes a transformer-based architecture [46,15], which is composed of space-time self-attention blocks. For our experiments, we use the checkpoint provided by the original authors which is pretrained on Conceptual Captions [44] and WebVid-2M [8].

4.2   Experimental setup

Datasets. We conduct our experiments on three datasets, which have multiple descriptions available for each video clip.

- **MSR-VTT** [55] is one of the most widely used datasets for video retrieval. It contains 10K video clips gathered from YouTube and each clip is annotated with 20 natural language descriptions. We follow the previous works to use 9K videos for training and 1K videos for testing.
- **MSVD** [9] is a dataset initially collected for translation and paraphrase evaluation. It contains 1970 videos, each with multiple descriptions in several different languages. We only use the English descriptions in our experiments. Following previous works, we use 1200 videos for training, 100 videos for validation and 670 videos for testing.
- **VATEX** [50] is a large-scale multilingual video description dataset. It contains 34991 videos, each annotated with ten English and ten Chinese captions. We only keep the English annotations and use the standard split with 25991 videos for training, 3000 for validation and 6000 for testing.

Implementation details. All the models are trained with 8-frame inputs and a batch size of 48 for 30 epochs. We adopt the cross entropy loss over similarity scores and a softmax temperature of 0.05 as used in [8]. AdamW [35] optimizer is used with a cosine learning rate scheduler and a linear warm up of 5 epochs [34]. The max learning rate is 3e-5 and 3e-6 for Frozen and CLIP4Clip respectively. We set query number to five for our multi-query experiments. Specifically, all available captions for each video are used during training, but for each training instance during forward pass, a subset of random five descriptions are input to the multi-query model. During test, we evaluate the N-query performance by sampling N query captions per video. However, to avoid selection bias, we repeat the evaluation for a hundred times (each time selects a different subset of N captions as queries for every video) and report the mean as the final results.

4.3   Results

Table 1 summarizes the evaluation results on the test split of the MSR-VTT, MSVD and VATEX datasets. Due to space limit, we only show the CLIP4Clip

Table 1: Performance of different methods on MSR-VTT, MSVD and VATEX. The baseline is trained and evaluated with one query. Others are evaluated with five-query input. RA, SA are trained with one query. MF, TS-WF, LG-WF, and CG-WF are trained with five-query input. All numbers are the average over 100 evaluations with different query samples. Recall numbers are reported in percent.

	MSR-VTT [55] (CLIP4Clip)					MSR-VTT (Frozen)				
	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓
Baseline	41.5	69.4	79.3	2.0	15.7	31.9	61.1	72.6	3.0	23.2
RA	56.4	84.5	92.1	1.0	4.3	44.6	76.2	85.2	2.0	7.0
SA	68.4	92.1	97.0	1.0	2.4	55.3	85.2	92.5	1.0	4.2
MF	71.3	93.5	97.6	1.0	2.2	59.6	88.0	93.9	1.0	4.0
TS-WF	72.6	**94.2**	**97.8**	1.0	2.1	**60.6**	**88.5**	**94.2**	1.0	3.8
LG-WF	72.6	94.1	97.7	1.0	2.1	59.6	87.6	93.8	1.0	3.9
CG-WF	**73.1**	94.1	**97.8**	1.0	2.1	59.9	88.1	94.0	1.0	3.7

	MSVD [9] (CLIP4Clip)					VATEX [50] (CLIP4Clip)				
	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓	MnR ↓
Baseline	43.8	74.0	83.0	2.0	10.9	33.3	63.5	76.1	3.0	17.9
RA	47.3	79.9	89.8	2.0	4.8	43.6	75.5	86.1	2.0	7.9
SA	57.6	87.1	93.7	1.0	3.4	47.3	77.9	87.9	2.0	6.7
MF	59.6	88.3	94.2	1.0	3.3	57.2	84.7	92.0	1.0	4.9
TS-WF	60.4	88.8	94.5	1.0	3.3	58.4	85.3	92.5	1.0	4.6
LG-WF	60.2	88.0	94.2	1.0	3.2	58.5	85.2	92.4	1.0	4.8
CG-WF	**61.7**	**89.5**	**94.9**	1.0	3.0	**59.0**	**85.6**	**92.7**	1.0	4.7

numbers for MSVD and VATEX datasets. Please see the Appendix for the full results of Frozen model. Since the experiment findings are similar across both models, we will focus on the CLIP4Clip model during the following discussion.

(R1) Multi-query vs. single-query. The first row of Table 1 shows the result for the single-query baseline, where the model is both trained and evaluated with one query. The rest of the rows show the results for five-query evaluation. With five queries available, all models see a big boost in terms of retrieval accuracy compared to single-query. While it's anticipated that multi-query models should perform better than the single-query counterpart, such big improvement is still striking, and shows that single-query evaluation could underestimate the real retrieval abilities of models.

(R2) Post-hoc aggregation methods. Comparing two post-hoc methods which simply extend the pretrained single-query model to the multi-query setting, it's clear that *similarity aggregation* outperforms *rank aggregation* by a

large margin. On CLIP4Clip, the R@1 is improved from 56.4% to 68.4% for MSR-VTT, from 47.3% to 57.6% for MSVD, and from 43.6% to 47.3% for VA-TEX, respectively. This is not surprising as the rank provided by a low-quality query is noisy and would thus drag down the final rank when combining with ranks provided by the other queries. While the similarity generated by a low-quality query is also noisy, its value tends to be small and is typically dominated by the similarity scores provided by the high-quality queries.

(R3) Multi-query training methods. It's clear from Table 1 that multi-query training methods with ad-hoc training modifications improve a lot over the post-hoc methods. Just by simply feeding five captions and taking the mean of the encoded features as the final text features during training (*mean feature*), R@1 of CLIP4Clip model can be improved by 2.9% (from 68.4% to 71.3%), 2.0% (from 57.6% to 59.6%) and 9.9% (from 47.3% to 57.2%) on MSR-VTT, MSVD and VATEX, respectively. We anticipate that such improvement can be mainly attributed to the de-noising effect of multi-query training. As the ranking loss acts on the combined features, the part of loss that tries to push apart the false-negative text-video pairs (due to general descriptions that match with more than one videos) is lessened, thus avoiding potential over-fitting and providing more robust features (additional evidence is also discussed in **R5**).

Even though the *mean feature* training is already a very strong baseline, additional weighting heuristics still manage to introduce further improvements. On the CLIP4Clip model, with the best-performing *weighted feature* training with *contextualized weight generation*, R@1 can be improved over *mean feature* by 1.8% (from 71.3% to 73.1%), 2.1% (from 59.6% to 61.7%) and 1.8% (from 57.2% to 59.0%) points across the datasets. To show that the learned weights can correctly capture the relative quality of the queries, we compute the average weights given to different queries ordered by their quality (we rank the quality of queries by their single-query retrieval result, 1 is the best, and 5 is the worst). Figure 3 shows the result, and it's clear that the generated weights can correctly reflect the quality of the query. The *contextualized weight generation* works better than the *localized weight generation*. This is expected as instead of trying to learn a standalone quality predictor, the former lessens the task by learning to predict the relative quality. Especially when training data is small, *e.g.,* in MSVD, *localized weight generation* would have a hard time learning the quality and give more spread weights across different queries. Figure 4 shows several qualitative examples on MSR-VTT for a model trained with *contextualized weight generation* (more examples shown in the Appendix). While some of the queries are of low-quality, the model can correctly attend to the better ones and achieve good overall retrieval accuracy.

(R4) Evaluation with varying number of queries. Table 1 shows the result where evaluation is performed under the same number of queries as multi-query models are trained (five-query training and five-query test). It is then natural to question whether such result can be generalized across varying number of queries. Figure 5 shows R@1 curves when the same models are tested under

Fig. 3: Averaged weights generated to five queries, which are ranked by their single-query retrieval results. 1 is the best and 5 is the worst. The average is taken over (test split size) * 100 instances.
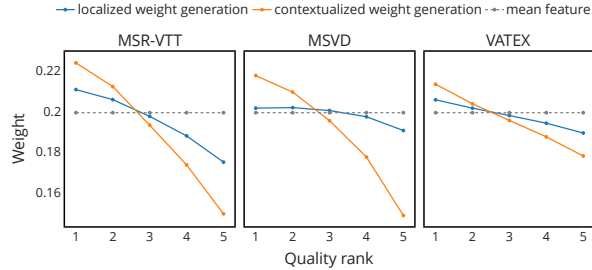
Video	Query	Weight	Single-query rank	Multi-query rank
	A man is busy in cooking in the kitchen	0.36	14	
	Person is mixing the ingredients and preparing the dessert	**0.64**	8	5
	A clip from a video game	0.19	13	
	A girl is talking about having a barn in the sims	**0.81**	1	1
	A man is singing	0.19	30	
	A band is performing on stage	0.32	9	2
	Two men playing guitar on stage	**0.49**	1	
	A man is playing the guitar	0.19	12	
	A band is playing on a stage	0.35	3	1
	A man playing a bass guitar and a man playing on a keyboard	**0.46**	1	

Fig. 4: Qualitative examples of a weighted model (CG-WF) on MSR-VTT. It correctly attends to the disrciminative descriptions.

different number of queries and Table 2 summarizes the results in the form of the proposed $\mathbf{AUC}_n^{R@1}$ metric. First, the performance of all methods improves with more queries available. The curve increases rapidly at beginning and gradually saturates as more queries provides marginally additional information. Second, the five-query trained models outperform the single-query trained models across different number of testing queries except one, with dominating scores for $\mathbf{AUC}_3^{R@1}$, $\mathbf{AUC}_5^{R@1}$ and $\mathbf{AUC}_{10}^{R@1}$. This shows that multi-query training indeed learns better features suited for multi-query evaluation. While single-query trained models maintain the lead at single-query test, the differences are very small. Comparing *similarity aggregation* with *contextualized weight generalization*, the $R@1_1$ (single-query R@1) is 41.5% vs. 41.0% for MSR-TT, 43.8% vs. 43.6% for MSVD, and 33.3% vs. 32.8% for VATEX respectively.

To get the best of both worlds, we further conduct an experiment with a combination of single-query and five-query *mean feature* training, *i.e.,* some training pairs contain one video and one caption while others contain one video and five captions (as with all other experiments, all available captions are used during
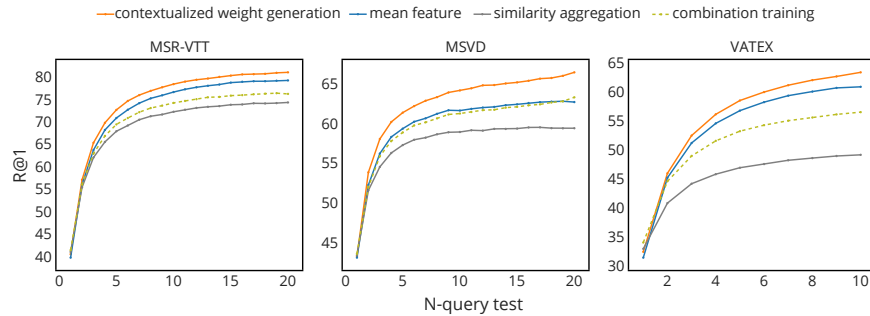
Fig. 5: R@1 performance for SA, MF, CG-WF and *combination training* when evaluated with varying number of queries.
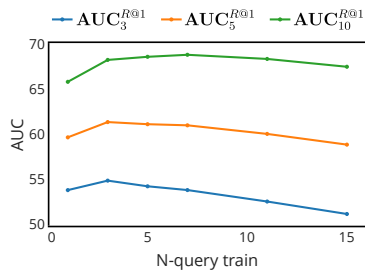
Table 2: Multi-query test results summarized with proposed $\mathbf{AUC}_n^{R@1}$ metric.

		$R@1_1$	$\mathbf{AUC}_3^{R@1}$	$\mathbf{AUC}_5^{R@1}$	$\mathbf{AUC}_{10}^{R@1}$
MSR-VTT	SA	41.5	54.0	59.9	66.1
	MF	40.3	54.4	61.3	68.8
	CG-WF	41.0	**55.5**	**62.7**	**70.4**
	Cmb (MF)	**41.9**	54.6	60.8	67.5
MSVD	SA	43.8	50.6	53.5	56.4
	MF	43.5	51.3	54.8	58.3
	CG-WF	43.6	**52.6**	**56.4**	**60.2**
	Cmb (MF)	**44.0**	51.2	54.5	57.9
VATEX	SA	33.3	40.1	43.1	46.2
	MF	31.8	43.7	49.2	55.1
	CG-WF	32.8	**44.6**	**50.4**	**56.8**
	Cmb (MF)	**34.4**	43.5	47.6	52.1

training, but for each training instance, a random sample of one or five captions is used as input). The dash lines in Figure 5 show the results. Rather surprisingly, the *combination training* achieves the best single-query performance and outperforms single-query training entirely. This shows that **the de-noising effect of multi-query training can also be utilized to improve standard single-query retrieval**.

(R5) Training with different number of queries. To understand the effect of number of queries (N) during training, we plot the performance of different models in Figure 6a. We observe that the performance for all models increases at first, followed by a decrease (sometimes to a score even lower than N=1) as more queries are added. We hypothesize that this may be due to over-smoothing effect in the training process when too many queries are used, which causes the model to not learn discriminative representations for each individual query.

To further demonstrate the utility of the MQVR framework in learning robust representations, we perform a zero-shot transfer evaluation on VATEX

Fig. 6: Evaluation of *mean feature* model trained on MSR-VTT with different number of queries. (a) In-domain test on MSR-VTT test set. (b) Out of domain zero-shot test on VATEX test set. "nq" means n-query training.

Training	$R@1_1$	$\textbf{AUC}_3^{R@1}$	$\textbf{AUC}_5^{R@1}$	$\textbf{AUC}_{10}^{R@1}$
VATEX 1q	33.3	40.1	43.1	46.2
MSR-VTT 1q	26.5	32.2	34.7	37.3
MSR-VTT 2q	27.0	33.2	36.0	38.9
MSR-VTT 3q	**27.1**	33.8	36.9	40.0
MSR-VTT 5q	**27.1**	34.7	38.2	41.9
MSR-VTT 7q	26.6	**34.8**	**38.7**	**42.8**

dataset using models trained on MSR-VTT (Table 6b). Quite promisingly, we observe that multi-query training (MSR-VTT 5q, MSR-VTT 7q) outperforms their single-query counterparts (MSR-VTT 1q) on both single-query and multi-query evaluations, achieving 0.6% higher single-query R@1 and 2.6-5.5 points higher AUC, respectively. The latter results especially are only a few points below the AUC performance of an in-domain VATEX 1q model, demonstrating that there is a strong transfer and indicating that **multi-query training can lead to better generalization**.

5   Conclusion

In this work, we study the multi-query retrieval problem where multiple descriptions are available for retrieving target videos. We argue that this previously less-studied setting is of practical value both because it can provide significant improvement on retrieval accuracy through incorporating information from multiple queries and that it addresses challenges in model training and evaluation introduced by imperfect descriptions. We then investigate several multi-query training methods and propose a new evaluation metric dedicated for this setting. With extensive experiments, we demonstrate that multi-query inspired training methods can provide superior performance and better generalization. We believe further investigation on the issue will benefit the field and bring new insights to building better retrieval systems for real-world applications.

References

1. Agrawal, A., Batra, D., Parikh, D., Kembhavi, A.: Don't just assume; look and answer: Overcoming priors for visual question answering. In: CVPR (2018)
2. Alayrac, J.B., Recasens, A., Schneider, R., Arandjelovic, R., Ramapuram, J., De Fauw, J., Smaira, L., Dieleman, S., Zisserman, A.: Self-supervised multimodal versatile networks. In: NeurIPS (2020)
3. Andrés Portillo-Quintero, J., Ortiz-Bayliss, J.C., Terashima-Marín, H.: A straightforward framework for video retrieval using CLIP. arXiv:2102.12443 (2021)
4. Anne Hendricks, L., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.: Localizing moments in video with natural language. In: ICCV (2017)
5. Anne Hendricks, L., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.: Localizing moments in video with natural language. In: ICCV (2017)
6. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR (2012)
7. Arandjelovic, R., Zisserman, A.: Look, listen and learn. In: ICCV (2017)
8. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. arXiv:2104.00650 (2021)
9. Chen, D., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: ACL (2011)
10. Cheng, X., Lin, H., Wu, X., Yang, F., Shen, D.: Improving video-text retrieval by multi-stream corpus alignment and dual softmax loss. arXiv:2109.04290 (2021)
11. Chum, O., Mikulik, A., Perdoch, M., Matas, J.: Total recall II: Query expansion revisited. In: CVPR (2011)
12. Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total recall: Automatic query expansion with a generative feature model for object retrieval. In: ICCV (2007)
13. Chung, J., hsin Wuu, C., ru Yang, H., Tai, Y.W., Tang, C.K.: Haa500: Human-centric atomic action dataset with curated videos. In: ICCV (2021)
14. Dong, J., Li, X., Xu, C., Yang, X., Yang, G., Wang, X., Wang, M.: Dual encoding for video retrieval by text. In: PAMI (2021)
15. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929 (2020)
16. Dzabraev, M., Kalashnikov, M., Komkov, S., Petiushko, A.: MDMMT: Multidomain multimodal transformer for video retrieval. In: CVPR (2021)
17. Fang, H., Xiong, P., Xu, L., Chen, Y.: CLIP2Video: Mastering video-text retrieval via image CLIP. arXiv:2106.11097 (2021)
18. Fernando, B., Tuytelaars, T.: Mining multiple queries for image retrieval: On-the-fly learning of an object-specific mid-level representation. In: CVPR (2013)
19. Gabeur, V., Nagrani, A., Sun, C., Alahari, K., Schmid, C.: Masking modalities for cross-modal video retrieval. In: WACV (2022)
20. Gabeur, V., Sun, C., Alahari, K., Schmid, C.: Multi-modal transformer for video retrieval. In: ECCV (2020)
21. Gao, Z., Liu, J., Chen, S., Chang, D., Zhang, H., Yuan, J.: CLIP2TV: An empirical study on transformer-based methods for video-text retrieval. arXiv:2111.05610 (2021)
22. Gordo, A., Almazan, J., Revaud, J., Larlus, D.: End-to-end learning of deep visual representations for image retrieval. IJCV (2017)

23. Gordo, A., Radenovic, F., Berg, T.: Attention-based query expansion learning. In: ECCV (2020)
24. Goyal, P., Caron, M., Lefaudeux, B., Xu, M., Wang, P., Pai, V.M., Singh, M., Liptchinsky, V., Misra, I., Joulin, A., Bojanowski, P.: Self-supervised pretraining of visual features in the wild. arXiv:2103.01988 (2021)
25. Huang, S., Hang, H.M.: Multi-query image retrieval using CNN and SIFT features. In: APSIPA ASC (2017)
26. Imani, A., Vakili, A., Montazer, A., Shakery, A.: Deep neural networks for query expansion using word embeddings. In: ECIR (2019)
27. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: MDETR-modulated detection for end-to-end multi-modal understanding. In: ICCV (2021)
28. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Carlos Niebles, J.: Dense-captioning events in videos. In: ICCV (2017)
29. Lei, J., Li, L., Zhou, L., Gan, Z., Berg, T.L., Bansal, M., Liu, J.: Less is more: ClipBERT for video-and-language learning via sparse sampling. In: CVPR (2021)
30. Li, G., He, F., Feng, Z.: A CLIP-Enhanced method for video-language understanding. arXiv:2110.07137 (2021)
31. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: ECCV (2020)
32. Liu, T., Lin, Y., Du, B.: Unsupervised person re-identification with stochastic training strategy. TIP (2022)
33. Liu, Y., Albanie, S., Nagrani, A., Zisserman, A.: Use what you have: Video retrieval using representations from collaborative experts. arXiv:1907.13487 (2019)
34. Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. arXiv:1608.03983 (2016)
35. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv:1711.05101 (2017)
36. Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., Li, T.: CLIP4clip: An empirical study of CLIP for end to end video clip retrieval. arXiv:2104.08860 (2021)
37. Miech, A., Laptev, I., Sivic, J.: Learning a text-video embedding from incomplete and heterogeneous data. arXiv:1804.02516 (2018)
38. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: Howto100M: Learning a text-video embedding by watching hundred million narrated video clips. In: ICCV (2019)
39. Mithun, N.C., Li, J., Metze, F., Roy-Chowdhury, A.K.: Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In: ICMR (2018)
40. Patrick, M., Huang, P.Y., Asano, Y., Metze, F., Hauptmann, A., Henriques, J., Vedaldi, A.: Support-set bottlenecks for video-text representation learning. arXiv:2010.02824 (2020)
41. Radenović, F., Tolias, G., Chum, O.: Fine-tuning CNN image retrieval with no human annotation. TPAMI (2018)
42. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. arXiv:2103.00020 (2021)
43. Rohrbach, A., Rohrbach, M., Tandon, N., Schiele, B.: A dataset for movie description. In: CVPR (2015)
44. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: ACL (2018)

45. Sun, C., Myers, A., Vondrick, C., Murphy, K., Schmid, C.: VideoBERT: A joint model for video and language representation learning. In: ICCV (2019)
46. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
47. Vural, C., Akbacak, E.: Deep multi query image retrieval. SPIC (2020)
48. Wang, L., Li, Y., Lazebnik, S.: Learning deep structure-preserving image-text embeddings. In: CVPR (2016)
49. Wang, X., Zhu, L., Yang, Y.: T2VLAD: global-local sequence alignment for text-video retrieval. In: CVPR (2021)
50. Wang, X., Wu, J., Chen, J., Li, L., Wang, Y.F., Wang, W.Y.: VATEX: A large-scale, high-quality multilingual dataset for video-and-language research. In: ICCV (2019)
51. Wang, Y., Lin, X., Wu, L., Zhang, W.: Effective multi-query expansions: Robust landmark retrieval. In: ACMMM (2015)
52. Wang, Y., Lin, X., Wu, L., Zhang, W.: Effective multi-query expansions: Collaborative deep networks for robust landmark retrieval. TIP (2017)
53. Wu, Y., Jiang, L., Yang, Y.: Switchable novel object captioner. TPAMI (2022)
54. Wu, Y., Yang, Y.: Exploring heterogeneous clues for weakly-supervised audio-visual video parsing. In: CVPR (2021)
55. Xu, J., Mei, T., Yao, T., Rui, Y.: MSR-VTT: A large video description dataset for bridging video and language. In: CVPR (2016)
56. Yu, Y., Kim, J., Kim, G.: A joint sequence fusion model for video question answering and retrieval. In: ECCV (2018)
57. Zhang, B., Hu, H., Sha, F.: Cross-modal and hierarchical modeling of video and text. In: ECCV (2018)
58. Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., Tian, Q.: Mars: A video benchmark for large-scale person re-identification. In: ECCV (2016)
59. Zhu, L., Yang, Y.: ActBERT: Learning global-local video-text representations. In: CVPR (2020)