Video-adverb retrieval with compositional adverb-action embeddings.

BMVC 2023 Submission # 581

Abstract

Retrieving adverbs that describe an action in a video poses a crucial step towards fine-grained video understanding. We propose a framework for video-to-adverb retrieval (and vice versa) that aligns video embeddings with their matching compositional adverb-action text embedding in a joint embedding space. The compositional adverb-action text embedding is learned using a residual gating mechanism, along with a novel training objective consisting of triplet losses and a regression target. Our method achieves state-of-the-art performance on five recent benchmarks for video-to-adverb retrieval. Furthermore, we propose dataset splits to benchmark the adverb-video retrieval for unseen adverb-action compositions on subsets of the MSR-VTT Adverbs and ActivityNet Adverbs datasets. Our proposed framework outperforms all prior works for the generalisation task of retrieving adverbs from videos for unseen adverb-action compositions. Code and the proposed dataset splits will be available upon acceptance.

1 Introduction

Fine-grained video understanding requires not only to recognise actions in videos, e.g. *cutting*, but also to understand details about the execution of an action, e.g. *cutting slowly*. While there has been significant progress in action retrieval and recognition in videos [2, 28, 39, 42], the fine-grained understanding of actions remains challenging. In particular, understanding properties of the actions themselves can require perceiving how they are performed. As a step towards achieving this, we consider the bidirectional video-to-adverb retrieval task where we retrieve adverbs that match an action in a video and vice versa.

In the bidirectional video-to-adverb retrieval task, adverbs and action words can be combined in a compositional manner. The same adverb can describe multiple actions, such as *cutting slowly* or *dancing slowly*. The compositional nature of the adverb-action pairings can also be exploited when learning adverb-action representations.

Our ReGaDa framework for adverb-video retrieval uses a **re**sidual **ga**ting mechanism to compose **ad**verb-**a**ction (ReGaDa) representations for retrieval. At its core, it learns to align adverb representations and video representations in a shared embedding space using a novel training objective which consists of a direct regression loss between the adverb and video representations and triplet losses. To obtain the adverb representation, the adverb and action are jointly embedded using a residual gating mechanism, which we adapted to the video-adverb retrieval task from [45]. It models the composition as a transformation of the adverb embedding based on the action, by using a gate and a residual mechanism. The gate allows the preservation of meaningful information from the adverb embeddings based on the adverb-action composition. Our final composition is learned as a residual combination on top of the gated adverb embeddings. This allows our composed embeddings to be in the same "feature space" as the original adverb embeddings. Crucially, the gated residual mechanism consistutes an inductive bias for this task through the gated mechanism which focuses on extracting the most useful information from the adverb and then adding it into the compositional embedding. Similar to previous works for this task, our model assumes knowledge of the ground-truth action class to perform adverb-video retrieval.

The compositional adverb-action embeddings and our proposed training objective prove to be beneficial for the adverb-retrieval performance, specifically for the retrieval of unseen adverb-action compositions. REGADA obtains state-of-the-art results on the five video-adverb retrieval benchmarks HowTo100M Adverbs [13, 27], VATEX Adverbs [12, 47], ActivityNet Adverbs [7, 12], MSR-VTT Adverbs [12, 53], and Adverbs in Recipes [27, 30]. Furthermore, we propose two additional splits for benchmarking the retrieval of unseen adverb-action compositions on the ActivityNet Adverbs and MSR-VTT Adverbs datasets. In our extensive model ablation studies, we show that our proposed compositional text encoder and our training objective boost the results for adverb-video retrieval and lead to better generalisation to unseen compositions.

To summarise, we make the following contributions: 1) Our proposed method for video-adverb retrieval uses a text encoder based on a gated residual mechanism and a novel training objective. 2) We evaluate REGADA on the challenging unseen video-adverb retrieval task and introduce new benchmark splits, compliant with zero-shot learning principles, for the retrieval of unseen adverb-action compositions based on the ActivityNet Adverbs and MSR-VTT Adverbs datasets. 3) Our framework outperforms prior work for both the seen and the unseen adverb-action composition retrieval tasks.

2 Related work

Fine-grained action understanding in video retrieval. Early works for video understanding extended retrieval approaches for images to videos, by temporally aggregating frames in a video [11, 37, 43, 54]. With the availability of large video-text datasets [3, 5, 20, 27, 36, 47, 53, 55], different methods focused on sentence disambiguation [9, 50], self-supervision [1, 40, 56], weakly supervised learning [27, 28, 39], multiple embedding experts [14, 23, 26], or the use of large pre-trained models [21, 24, 38, 51]. Video-action retrieval specifically aims at retrieving videos based on an action, e.g. using a verb to describe the same [16, 49]. Moreover, [10, 15, 50, 54, 57] use nouns in addition to verbs for video-text retrieval. In a more general setting, [31] recently proposed to use a large language model to generate modified captions to improve verb understanding in video-language models. Different to these methods, we focus on adverbs in the adverb-video retrieval task.

Adverb-video retrieval. The adverb-video retrieval task was introduced by [13] along with the HowTo100M Adverbs dataset. [13] learns a shared representation between videos and adverbs, modeling adverb information as learned linear transformations on action class label word embeddings, similar to [33] for object attributes. Unlike [13], we choose to utilize semantic information from adverb embeddings in addition to action embeddings for modeling adverb-action compositions. [12] extends [13] to the low-data regime with pseudo-labelling. The recently proposed [30] tackles the task either as a classification or regression problem. Its video encoder builds on [13] with an additional projection following the attention while

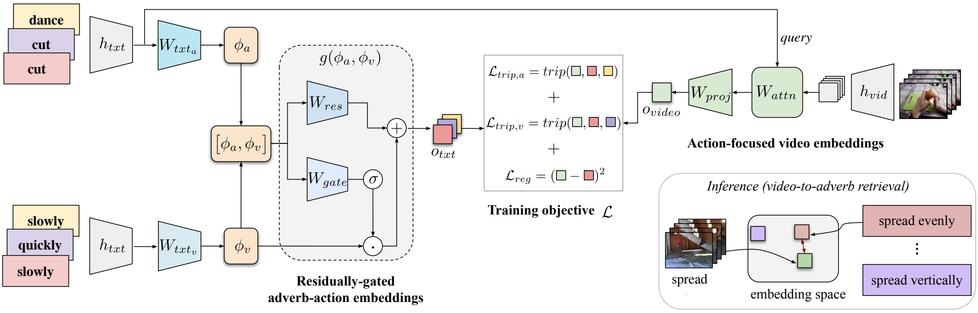

Figure 1: **Overview of our ReGaDa framework for video-to-adverb and adverb-to-video retrieval.** Our model composes adverb-action embeddings with a gated residual between the adverbs ϕ_v and the concatenated action and adverb embeddings $[\phi_a, \phi_v]$. The training objective $\mathcal{L}$ aligns the learned text and video representations in a joint embedding space. For test time inference, outputs are obtained based on similarity in the embedding space.

keeping the text representations frozen. The classification variant is trained with a cross-entropy loss for adverb classification, while the regression variant uses a regression target describing the change an adverb induced in an action embedding. Different to [30], our work aims at learning adverb-action representations and video representation in a shared embedding space. We show that formulating the task as a shared-embedding space alignment problem, combined with a novel text encoder for creating compositional adverb-action representations, significantly boosts the performance for video-adverb retrieval.

Learning with object attributes. Approaches for learning object-attribute pairs from images can be broadly categorized into classification [22, 25, 29, 32, 33] and retrieval approaches [6, 8, 18, 35, 45, 46, 48]. Our adverb-action compositions are most closely related to [45], which proposed a residual gating mechanism for learning compositional image-text embeddings. This mechanism proved particularly useful for retrieving images using both an image and a text query, the text describing a desired modification on the query image. We adapt a similar residual gating mechanism for learning compositional adverb-action embeddings by aligning the composition with action-focused video embeddings.

3 ReGaDa framework for adverb-video retrieval

In this section, we provide details about our proposed ReGaDa framework for adverb-video retrieval which is visualized in Fig. 1. We first describe the adverb-video retrieval task, and then provide details about the video and text encoders in our framework. Finally, we detail our training objective and the inference procedure for adverb-video retrieval.

Task setting and dataset. The adverb-to-video retrieval task aims at retrieving matching videos from a pool of videos for a given adverb. Similarly, for the video-to-adverb retrieval task, given a video, the aim is to retrieve the adverb that best describes the action depicted in the video from a pool of pre-set adverbs. We denote a dataset with N samples, A action classes and V adverb classes by $\mathcal{D} = \{\mathcal{X}_{[i]}, y_{[i]}\}_{i=1}^{N}$, consisting of video data $\mathcal{X}_{[i]}$, and ground-truth action and adverb labels $y_{[i]} = \{a_{[i]}, v_{[i]}\}$ with one-hot encodings for the action $a_{[i]} \in \mathbb{R}^A$ and adverb $v_{[i]} \in \mathbb{R}^V$. We define the sets of possible actions and adverbs as $\mathcal{A}$ and $\mathcal{V}$. The set of all possible adverb-action combinations is $\mathcal{C} = \mathcal{A} \times \mathcal{V}$.

Our proposed REGADA framework learns to align video and adverb-action representations in a joint embedding space. It generates compositional textual representations for adverb-action pairs using a text encoder. Additionally, the visual information is processed in a video encoder to obtain visual representations that contain information about the adverb associated with a given action. In the following, we describe how we obtain class label embeddings for the actions and adverbs, and how the video and text encoders process the video features and class label embeddings.

Residually-gated adverb-action embeddings. We obtain word embeddings for the action a and for the adverb v from a pre-trained language encoder h_{txt}. This gives $\theta_v = h_{txt}(v)$, and $\theta_a = h_{txt}(a)$, where $\theta_a, \theta_v \in \mathbb{R}^{d_\theta}$, for the adverb v and action a respectively. We additionally use two linear maps $W_{txt_a}, W_{txt_v} : \mathbb{R}^{d_\theta} \to \mathbb{R}^{d_{dim}}$, such that $\phi_a = W_{txt_a}(\theta_a)$ and $\phi_v = W_{txt_v}(\theta_v)$. The action and adverb embeddings are then further processed jointly in our text encoder. Additionally, the action word embedding θ_a serves as a query vector in the video encoder's attention for generating an action-focused video embedding.

Our text encoder uses a residual gating mechanism which is based on [45]. Given ϕ_a and ϕ_{v_j} as inputs, the output of the text encoder is defined as:

$$o_{txt_j} = g(\phi_a, \phi_{v_j}) = \omega_g * g^{gate}(\phi_a, \phi_{v_j}) + \omega_r * W^{res}(\phi_a, \phi_{v_j}), \tag{1}$$

where $j \in \{1, \cdots, V\}$, and ω_g, ω_r are learnable scalar weights for balancing the gating mechanism and the residual. For easier readability, we omit the subscripts $_j$ in the following:

$$g^{gate}(\phi_a, \phi_v) = \sigma(W_{gate}(\phi_a, \phi_v)) \odot \phi_v, \tag{2}$$

where $\odot$ is an element-wise product, σ the sigmoid function, and W_{gate} is an MLP with $W_{gate}(\phi_a, \phi_v) = o_{gate}^{N_g}$. The first $N_g - 1$ layers of W_{gate} consist of a linear layer $W_{gate}^l : \mathbb{R}^{2*d_{dim}} \to \mathbb{R}^{2*d_{dim}}$, a dropout layer [41] g_{gate}^{DL} with probability $drop_{gate}$, and Leaky ReLU [52] g_{gate}^{LReLU}. The input is passed through a concatenation operator $[*]$ and batch normalisation [17] g_{gate}^{bn}, such that $o_{gate}^0 = g_{gate}^{bn}([\phi_a, \phi_v])$. The last layer is a linear layer $W_{gate}^{N_g} : \mathbb{R}^{2*d_{dim}} \to \mathbb{R}^{d_{dim}}$. We can then write

$$o_{gate}^l = \begin{cases} g_{gate}^{LReLU}(g_{gate}^{DL}(W_{gate}^l(o_{gate}^{l-1})), & 0 \leq l \leq N_g - 1 \\ W_{gate}^{N_g}(o_{gate}^{l-1}), & l = N_g. \end{cases} \tag{3}$$

The residual function W^{res} consists of an MLP with N_r layers, such that $W_{res}(\phi_a, \phi_v) = o_{res}^{N_r}$. The first $N_r - 1$ layers are composed of a linear layer $W_{res}^l : \mathbb{R}^{2*d_{dim}} \to \mathbb{R}^{2*d_{dim}}$, dropout g_{res}^{DL} with probability $drop_{gate}$, and Leaky ReLU activation function g_{res}^{LReLU}. The last layer, N_r is a linear layer $W_{res}^{N_r} : \mathbb{R}^{2*d_{dim}} \to \mathbb{R}^{d_{dim}}$. The inputs are first concatenated with a concatenation operator, and batch normalisation g_{res}^{bn} is applied, to give $o_{res}^0 = g_{res}^{bn}([\phi_a, \phi_v])$. We then get:

$$o_{res}^l = \begin{cases} g_{res}^{LReLU}(g_{res}^{DL}(W_{res}^l(o_{res}^{l-1})), & 0 \leq l \leq N_r - 1 \\ W_{res}^{N_r}(o_{res}^{l-1}), & l = N_r. \end{cases} \tag{4}$$

We tackle adverb-video retrieval by aligning text and videos in a learned shared embedding space. Our residual gating mechanism models the composition as a transformation of the adverb embedding based on the action. The gating mechanism g^{gate} thereby allows to retain information from adverbs when actions do not provide useful semantic information.

Action-focused video embeddings. A pre-trained video classification network h_{vid} is used to extract a sequence of visual features $\boldsymbol{x}_{[i]} = \{x_1, ..., x_t, ..., x_T\}_i$, where $\boldsymbol{x}_{[i]} = h_{vid}(\mathcal{X}_{[i]})$ and $x_t \in \mathbb{R}^{d_x}$. We use T to denote the number of temporal segments in a video clip.

Given a sequence of video features $\boldsymbol{x}_{[i]}$ and its associated action word embedding $\theta_{a_{[i]}}$ (for easier readability, we omit the subscripts $_{[i]}$), we obtain action-focused video embeddings using a similar mechanism as the one proposed in [13]. The video embeddings are obtained using weak action-level ground-truth in the scaled multi-headed dot-product attention [44]. The action word embedding θ_a serves as the query for the attention mechanism to focus on parts of the video that are relevant to the given action, and ignore the temporal segments that may be relevant to other actions.

For the multi-head attention, we map the video features $\{x_t\}_{t\in[1,T]}$ to keys and values using linear maps $W_k : \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_{head_x} H_x}$, $W_v : \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_{head_x} H_x}$ with H_x heads and a dimension of d_{head_x} per head. We also map the action word embeddings θ_a to queries with $W_q : \mathbb{R}^{d_\theta} \rightarrow \mathbb{R}^{d_{head_x} H_x}$. For each attention head j, we have

$$p^j_{attn} = g^{DL}_{attn}\left(softmax\left(\frac{W^j_q(\theta_a)^T W^j_k(\boldsymbol{x})}{\sqrt{d_{head_x}}}\right)\right) W^j_v(\boldsymbol{x}), \tag{5}$$

where g^{DL}_{attn} denotes dropout with probability $drop_{attn}$. The output video embedding is provided by a linear mapping $W_{attn} : \mathbb{R}^{d_{head_x} H_x} \rightarrow \mathbb{R}^{d_{dim}}$ of the aggregation of the per-head attention: $o_{attn} = W_{attn}([p^1_{attn}, \cdots, p^H_{attn}])$. The final output is obtained with an MLP, $W_{proj} : \mathbb{R}^{d_{dim}} \rightarrow \mathbb{R}^{d_{dim}}$, which gives

$$o_{video} = W_{proj}(o_{attn}), \tag{6}$$

where each of the N_{proj} layers of W_{proj} consists of a linear layer $W^l_{proj} : \mathbb{R}^{d_{dim}} \rightarrow \mathbb{R}^{d_{dim}}$, layer normalisation [4] g^{LN}_{proj}, ReLU [34] g^{ReLU}_{proj}, and dropout g^{DL}_{proj} with probability $drop_{proj}$.

Training objectives. Our REGADA framework is trained with triplet losses (based on [13]) and with a direct regression loss between video and text embeddings. We define the triplet loss function as $trip(a, p, n) = max(0, \|a - p\|_2 - \|a - n\|_2 + \mu)$, with a as the anchor embedding, p and n as the embedding of the positive and negative sample, and μ as the margin. The **action triplet loss** encourages the alignment of the video representation o_{video} and text embeddings with the matching action as opposed to a sampled negative action $\phi_{\bar{a}}$. For this, we use the video embedding as the anchor, the text embedding with ground truth action ϕ_a and adverb ϕ_v as the positive sample, and the text embedding of the same adverb but different action as a negative:

$$\mathcal{L}_{trip,a} = \frac{1}{n}\sum_{i=1}^{n} trip(o_{video_i}, g(\phi_{a_i}, \phi_{v_i}), g(\phi_{\bar{a}_i}, \phi_{v_i})) \quad \text{for } \phi_{\bar{a}_i} \neq \phi_{a_i}. \tag{7}$$

We use an **adverb triplet loss** to push text embeddings containing the adverb antonym $\phi_{\bar{v}}$ further away from the ground-truth text embedding:

$$\mathcal{L}_{trip,v} = \frac{1}{n}\sum_{i=1}^{n} trip(o_{video_i}, g(\phi_{a_i}, \phi_{v_i}), g(\phi_{a_i}, \phi_{\bar{v}_i})). \tag{8}$$

By restricting the negatives of adverbs to antonyms, the loss does not punish potential ambiguities of actions in videos (e.g. a drawer being opened slowly can at the same time be opened partially but not quickly). Our **regression loss** directly minimises the distance between the output video and text embeddings:

$$\mathcal{L}_{reg} = \frac{1}{n}\sum_{i=1}^{n} (o_{video_i} - g(\phi_{a_i}, \phi_{v_i}))^2. \tag{9}$$

Dataset	# tr (s)	# t (s)	# tr (p)	# t (p)
VATEX	6603	3293	319	316
MSR-VTT	987	454	225	225
ActivityNet	1490	848	635	543

Table 1: Statistics of the proposed dataset splits for the retrieval of unseen adverb-action compositions on the MSR-VTT and ActivityNet datasets. (tr: train, t: test, s: samples, p: adverb-action pairs)

Model	VATEX	ActivityNet	MSR-VTT
Act. Mod. [12]	53.8	57.0	56.0
AC$_{\text{CLS}}$ [30]	54.3	55.1	53.7
AC$_{\text{REG}}$ [30]	54.9	53.9	59.0
REGADA	**60.4**	**58.9**	**60.9**

Table 2: Retrieval of unseen adverb-action compositions on the VATEX, ActivityNet and MSR-VTT benchmarks. [12] uses pseudo-labelling.

The final loss is computed as the weighted sum of the above losses according to

$$\mathcal{L} = \lambda_a * \mathcal{L}_{trip,a} + \lambda_v * \mathcal{L}_{trip,v} + \lambda_{reg} * \mathcal{L}_{reg}, \tag{10}$$

with λ_a, $\lambda_v, \lambda_{reg} \in \mathbb{R}$.

Retrieving adverbs and videos (inference). Similar to [13], we evaluate our method on adverb-to-video and video-to-adverb retrieval given the ground-truth action a. For video-to-adverb retrieval, given a video $\boldsymbol{x}$ and action query a, we embed the video to obtain o_{video}, and we obtain embeddings for j adverb-action combinations o_{txt_j} for $j \in \{1, \cdots, V\}$. Using the cosine similarity metric we rank all the text embeddings o_{txt_j} by their similarity to the query video embedding o_{video} and we consider the highest-ranked pair as the retrieved adverb.

For adverb-to-video retrieval, given an adverb v and action a that are embedded to o_{txt}, we define the set of test videos containing action a as Γ. We rank all video embeddings o_{video_j} for videos in Γ using the similarity computed between each o_{video_j} and o_{txt} and select the video which is closest to o_{txt}.

4 Adverb-video retrieval benchmarks

In this section, we provide details about the datasets used in our experiments. In particular, we use five datasets for adverb-video retrieval. Furthermore, we propose two new dataset splits for the task of retrieving adverbs from videos for unseen adverb-action compositions.

Adverb-video retrieval datasets. HowTo100M Adverbs [13] consists of 5,824 video clips with annotations for 6 adverbs and 72 actions. In the following, we refer to HowTo100M Adverbs as **HowTo100M**. The recently proposed **Adverbs in Recipes** dataset has 10 adverbs, 48 actions and 7,003 videos. VATEX Adverbs [12] dataset has, with 34 adverbs and 135 actions, the largest variety of annotated adverbs and actions, consisting of 14,617 videos. We refer to VATEX Adverbs as **VATEX**. ActivityNet Adverbs [12] consists of 3,099 videos with 20 adverbs and 114 actions, and MSR-VTT Adverbs [12] of 1,824 videos with 18 adverbs and 106 actions. We refer to those as **ActivityNet** and **MSR-VTT** respectively.

Unseen adverb-action compositions splits. We explore the ability of REGADA to recognise adverbs for unseen adverb-action combinations during testing. [12] proposed a dataset split for unseen compositions on the VATEX dataset. Using the available videos in VATEX from [30], we replicate this split for the S3D video and text features used in this work, by omitting the unavailable videos. We additionally propose new splits for ActivityNet and MSR-VTT. We exclude HowTo100M Adverbs and Adverbs in Recipes, as both are subsets from HowTo100M which was used to pre-train the text and S3D video model, and hence this would not comply with zero-shot learning principles. To create splits for ActivityNet and MSR-VTT, we follow the protocol in [12]: We first split the set of possible compositions into two non-overlapping sets, so that all adverbs and all actions are present in both sets, but individual compositions are only contained in one of the sets. We additionally constrain the compositions for each set so that for a given adverb-action composition, its antonym-action composition is assigned to the same set. We assign the videos from one of the sets to the training set and split the videos of the other half into two different sets, assigning half of the instances in each composition to the test set and the other to an unlabelled set (which is used to train [12] with pseudo-labelling). Table 1 shows the details on the replicated split for VATEX, as well as our newly proposed splits for ActivityNet and MSR-VTT (the full details are provided in the supplementary material).

5 Experiments

In this section, we provide details about the baselines, implementation details, and evaluation metrics used in this work. Adverb-video retrieval results on five benchmarks are presented in Section 5.1, and we present model ablation studies in Section 5.2. In Section 5.3, we investigate the transfer to unseen adverb-action compositions during inference.

Baselines. Here, we briefly describe the baselines which we compare to. We report the **Prior** and **S3D pre-trained** baselines from [30]. **Prior** does not use any training but uses the data distribution and adverb frequency for scoring. **S3D pre-trained** is also training-free and uses the similarity between frozen video and text representations from the S3D backbone jointly trained on video and text. We also compare our framework to **Action Modifier** [13] and to the recently proposed **AC** frameworks [30]. AC tackles the task either as a classification (AC_{CLS}) or regression (AC_{REG}) problem.

Implementation details. We use the video and text features provided by [30], extracted using a frozen S3D model jointly pre-trained on video-text pairs from HowTo100M [27]. Here, $d_x = 1024$ and T is the length of the video in seconds, and $d_\theta = 512$. REGADA uses an internal embedding dimension $d_{dim} = 400$. We use $N_g = 2$, except for HowTo100M and Adverbs in Recipes where $N_g = 3$ and $N_g = 4$ respectively. $N_r = 2$ except for Adverbs in Recipes where $N_r = 3$. The residual-gated dropout probability is $drop_{gate} = 0.6$ and for Adverbs in Recipes and HowTo100M $drop_{gate} = 0.7$. $\lambda_a = 1$, $\lambda_v = 2.0$ for all datasets and $\lambda_v = 1.5$ for Adverbs in Recipes and $\lambda_{reg} = 1.0$ for all dataset except HowTo100M where $\lambda_{reg} = 1.5$. We use a batch size of 512, and the Adam [19] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay 10^{-5}, to train all models. Our method is trained for 2000 epochs using a $lr = 10^{-5}$ for all datasets, except HowTo100M where $lr = 3 * 10^{-5}$. We follow [30], and train all baselines for 1000 epochs using a learning rate of 10^{-4}. We conduct all experiments on a single Nvidia 2080-Ti GPU.

Evaluation metrics. We follow [30], and report mean Average Precision (mAP) scores for adverb-to-video-retrieval, in particular **mAP M** ("adverb-to-video (all)" in [13]) and **mAP W**. mAP M is computed by ranking videos that contain the same ground-truth action according to their similarity to the adverb-action text embedding. For mAP W, the class scores are reweighed according to their support size in the test set. For video-to-adverb retrieval, we report binary antonym accuracy **Acc-A**. This is equivalent to ranking adverb-action embeddings according to their similarity to the embedded video and calculating the mAP by restricting the set of adverbs to the target adverb and its antonym ("video-to-adverb (antonym)" in [13]). Similar to [30], we report each best metric independently.

	HowTo100M [illegible]			Adverbs in Recipes [30]			ActivityNet [illegible]			MSR-VTT [illegible]			VATEX [illegible]		
	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A
Priors	0.446	0.354	0.786	0.491	0.263	0.854	0.217	0.159	0.745	0.308	0.152	0.723	0.216	0.086	0.752
S3D pre-tr.	0.339	0.238	0.560	0.389	0.173	0.735	0.118	0.070	0.560	0.194	0.075	0.603	0.122	0.038	0.586
Act. M. [illegible]	0.406	0.372	0.796	0.509	0.251	0.857	0.184	0.125	0.753	0.233	0.127	0.731	0.139	0.059	0.751
$AC_{CLS}^{\dagger}$ [30]	0.562	0.420	0.786	0.606	0.289	0.841	0.130	0.096	0.741	0.305	0.131	0.751	0.283	0.108	0.754
$AC_{REG}^{\dagger}$ [30]	0.555	0.423	0.799	0.613	0.244	0.847	0.119	0.079	0.714	0.282	0.114	0.774	0.261	0.086	0.755
ReGaDa	**0.566**	**0.528**	**0.817**	**0.704**	**0.417**	**0.875**	**0.239**	**0.175**	**0.770**	**0.375**	**0.229**	**0.780**	**0.290**	**0.115**	**0.816**

Table 3: Results for adverb-to-video (mAP W/M) and video-to-adverb retrieval (Acc-A). Higher is better for all metrics. † refers to updated results provided by the authors.

5.1 Comparison with the state of the art

In Table 3, we present retrieval results with our ReGaDa framework on five benchmark datasets. It can be observed that ReGaDa outperforms all the other baselines in every metric and dataset. On VATEX, ReGaDa drastically outperforms AC_{CLS} on the adverb-to-video retrieval metrics mAP W and mAP M with scores of 0.290 and 0.115 compared to 0.283 and 0.108. For the video-to-adverb retrieval measure Acc-A, ReGaDa outperforms AC_{REG} with a score of 0.816 compared to 0.755. The same can be observed across all datasets with ReGaDa significantly outperforming all other methods. The most recent and strongest competitor [30], optimises its system using two different losses and reports the best results obtained from these two models for each dataset and metric. [30] does not consistently achieve a high performance for both its model variants. It can be seen that our ReGaDa model outperforms [30] across all metrics and datasets, showing that ReGaDa is more robust than the previous state of the art.

5.2 Model ablations

This section analyses the impact of using different input text information, losses, and components in the text encoder on the overall adverb-retrieval performance of ReGaDa.

Input to the text encoder. The residual gate g^{gate} allows to directly transmit adverb information ϕ_v if the action ϕ_a is not informative. We refer to the adverb as the *main* and action as *auxiliary* modality in ReGaDa. In Table 4, we show the impact of using different main and auxiliary modalities. We investigate if a compositional adverb-action word embedding ϕ_{comp} can be used as the main modality instead, which directly embeds an adverb-action label pair using h_{text} (e.g. "*cut quickly*"). ReGaDa obtains scores of 0.290 and 0.115 for mAP W and mAP M on VATEX compared to 0.245 and 0.080 when using ϕ_a as main modality and ϕ_v as auxiliary. Acc-A is less affected by the type of input information, ReGaDa obtains 0.816 compared to 0.806 when using ϕ_{comp} as main and ϕ_a as auxiliary modality. Overall, using ϕ_v as main and ϕ_a as auxiliary is most effective across datasets.

Losses. In Table 5, we show the impact of our three loss functions, $\mathcal{L}_{trip,a}$, $\mathcal{L}_{trip,v}$ and $\mathcal{L}_{reg}$. On VATEX, ReGaDa obtains a mAP W and mAP M of 0.290 and 0.115 compared to 0.184 and 0.75 when using only L_{reg}. For Acc-A, ReGaDa obtains a score of 0.816 compared to 0.753 for $L_{trip,a} + L_{trip,v}$. The regression loss L_{reg} boosts the performance on all datasets significantly. Our novel loss combination consistently gives the best adverb-video retrieval performance by better aligning adverb-action compositions and video representations. Previous work either only used triplet losses [12, 13] or used a fixed textual regression target [30].

Residual gating mechanism in the text encoder. Table 6 analyses the contributions of the components of the residual gating mechanism, such as the residual branch, the sigmoid, and potential weight sharing between the gated and residual branches. On VATEX, ReGaDa achieves the best results. However, ReGaDa obtains 0.290 for mAP W compared to 0.288

Text Input		HowTo100M [■]			Adverbs in Recipes [■]			ActivityNet [■]			MSR-VTT [■]			VATEX [■]		
main	*auxiliary*	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A
ϕ_a	ϕ_v	0.480	0.414	0.820	0.436	0.217	0.874	0.224	0.146	0.763	0.330	0.130	0.766	0.245	0.080	0.808
ϕ_{comp}	ϕ_v	0.498	0.467	0.820	0.516	0.332	0.879	0.220	0.150	0.750	0.340	0.159	0.783	0.256	0.084	0.809
ϕ_{comp}	ϕ_a	0.508	0.474	**0.830**	0.523	0.365	**0.882**	0.221	0.149	0.758	0.337	0.155	**0.791**	0.256	0.090	0.806
ϕ_v	ϕ_a	**0.566**	**0.528**	0.817	**0.704**	**0.417**	0.875	**0.239**	**0.175**	**0.770**	**0.375**	**0.229**	0.780	**0.290**	**0.115**	**0.816**

Table 4: Effect of using different types of input information for the text encoder in REGADA.

Loss			HowTo100M [■]			Adverbs in Recipes [■]			ActivityNet [■]			MSR-VTT [■]			VATEX [■]		
$\mathcal{L}_{trip,a}$	$\mathcal{L}_{trip,v}$	$\mathcal{L}_{reg}$	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A
✓	✗	✗	0.345	0.239	0.743	0.422	0.209	0.839	0.128	0.079	0.666	0.259	0.123	0.734	0.166	0.058	0.741
✗	✓	✗	0.336	0.223	0.678	0.430	0.213	0.836	0.163	0.106	0.585	0.258	0.138	0.714	0.133	0.047	0.680
✗	✗	✓	0.469	0.378	0.743	0.468	0.233	0.838	0.204	0.143	0.732	0.289	0.186	0.734	0.184	0.075	0.699
✓	✓	✗	0.362	0.243	0.755	0.469	0.239	0.851	0.156	0.096	0.666	0.278	0.120	0.734	0.174	0.060	0.753
✓	✓	✓	**0.566**	**0.528**	**0.817**	**0.704**	**0.417**	**0.875**	**0.239**	**0.175**	**0.770**	**0.375**	**0.229**	**0.780**	**0.290**	**0.115**	**0.816**

Table 5: Impact of using different losses to train REGADA. For losses that are not used, the corresponding scalar weight in $\mathcal{L}$ is set to zero.

Components			HowTo100M [■]			Adverbs in Recipes [■]			ActivityNet [■]			MSR-VTT [■]			VATEX [■]		
R	σ	SW	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A	mAP W	mAP M	Acc-A
✓	✓	✓	0.535	0.433	0.811	0.689	0.404	0.875	**0.256**	**0.190**	**0.771**	0.374	0.182	0.766	0.288	0.109	0.808
✓	✗	✗	0.512	0.496	0.811	0.501	0.269	0.862	0.234	0.171	0.770	0.360	0.194	0.780	0.260	0.098	0.804
✗	✓	✗	0.516	0.477	**0.817**	0.562	0.296	**0.877**	0.228	0.169	0.765	0.367	0.161	**0.783**	0.283	0.111	0.815
✓	✓	✗	**0.566**	**0.528**	**0.817**	**0.704**	**0.417**	0.875	0.239	0.175	0.770	**0.375**	**0.229**	0.780	**0.290**	**0.115**	**0.816**

Table 6: Impact of different components in the residually-gated text encoder. R: With residual branch W_{res}; σ: With sigmoid; SW: Sharing weights between W_{res} and W_{gate}.

when using shared weights. For mAP M and Acc-A, REGADA obtains 0.115 and 0.816 compared to 0.111 and 0.815 when not using the residual. While other combinations can achieve better results in chosen metrics, there is no consistent combination that yields state-of-the-art results in every metric, except ours. This confirms our model design choices.

5.3 Generalisation to unseen adverb-action compositions

We additionally evaluate our REGADA framework on adverb-retrieval for unseen adverb-action compositions on three benchmarks VATEX, MSR-VTT, ActivityNet (see Section 4). Following [12], we report binary antonym classification accuracy for video-to-adverb retrieval. In Table 2, we observe that REGADA significantly outperforms AC_{REG} on VATEX with a score of 60.4 compared to 54.9. On ActivityNet, REGADA obtains a score of 58.9, outperforming [12] with a score of 57.0. This is impressive given that [12] was additionally trained on pseudo-labelled data. We provide a further analysis of using different word embeddings in the supplementary material. Overall, our model performs better than any of the previous baselines for both seen (c.f. Table 3) and unseen compositions.

6 Conclusion

In this work, we proposed a framework for adverb-video retrieval that uses a residual gating mechanism to generate compositional adverb-action representations from adverb and action word embeddings. Along with a novel training objective, our model achieves state-of-the-art results on five adverb-video retrieval benchmarks. Moreover, we introduce two additional dataset splits to benchmark the retrieval of unseen adverb-action compositions. Our proposed framework outperforms all prior works on this task, confirming that our text encoder results in better generalisation abilities.

References

[1] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *NeurIPS*, 2020.

[2] Taha Alhersh, Heiner Stuckenschmidt, Atiq Ur Rehman, and Samir Brahim Belhaouari. Learning human activity from visual data using deep learning. *IEEE Access*, 2021.

[3] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017.

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[5] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021.

[6] Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *ACM MM*, 2013.

[7] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[8] Chao-Yeh Chen and Kristen Grauman. Inferring analogous attributes. In *CVPR*, 2014.

[9] Hui Chen, Guiguang Ding, Zijia Lin, Sicheng Zhao, and Jungong Han. Cross-modal image-text retrieval with semantic consistency. In *ACM MM*, 2019.

[10] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *CVPR*, 2020.

[11] Jianfeng Dong, Xirong Li, and Cees GM Snoek. Predicting visual features from text for image and video caption retrieval. In *IEEE Transactions on Multimedia*, 2018.

[12] Hazel Doughty and Cees GM Snoek. How do you do it? fine-grained action understanding with pseudo-adverbs. In *CVPR*, 2022.

[13] Hazel Doughty, Ivan Laptev, Walterio Mayol-Cuevas, and Dima Damen. Action Modifiers: Learning from Adverbs in Instructional Videos. In *CVPR*, 2020.

[14] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.

[15] Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *CVPR*, 2022.

[16] Meera Hahn, Andrew Silva, and James M Rehg. Action2vec: A crossmodal embedding approach to action learning. *arXiv preprint arXiv:1901.00484*, 2019.

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[18] Phillip Isola, Joseph J Lim, and Edward H Adelson. Discovering states and transformations in image collections. In *CVPR*, 2015.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

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

[21] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021.

[22] Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. Symmetry and group in attribute-object compositions. In *CVPR*, 2020.

[23] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. *BMVC*, 2019.

[24] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 2022.

[25] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *CVPR*, 2021.

[26] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv preprint arXiv:1804.02516*, 2018.

[27] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.

[28] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.

[29] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *CVPR*, 2017.

[30] Davide Moltisanti, Frank Keller, Hakan Bilen, and Laura Sevilla-Lara. Learning action changes by measuring verb-adverb textual relationships. In *CVPR*, 2023.

[31] Liliane Momeni, Mathilde Caron, Arsha Nagrani, Andrew Zisserman, and Cordelia Schmid. Verbs in action: Improving verb understanding in video-language models. *arXiv preprint arXiv:2304.06708*, 2023.

[32] Muhammad Ferjad Naeem, Yongqin Xian, Federico Tombari, and Zeynep Akata. Learning graph embeddings for compositional zero-shot learning. In *CVPR*, 2021.

[33] Tushar Nagarajan and Kristen Grauman. Attributes as operators: factorizing unseen attribute-object compositions. In *ECCV*, 2018.

[34] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[35] Zhixiong Nan, Yang Liu, Nanning Zheng, and Song-Chun Zhu. Recognizing unseen attribute-object pair with generative model. In *AAAI*, 2019.

[36] Andreea-Maria Oncescu, Joao F Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations. In *ICASSP*, 2021.

[37] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. Learning joint representations of videos and sentences with web image search. In *ECCV*, 2016.

[38] Jae Sung Park, Sheng Shen, Ali Farhadi, Trevor Darrell, Yejin Choi, and Anna Rohrbach. Exposing the limits of video-text models through contrast sets. In *ACL*, 2022.

[39] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F. Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *ICLR*, 2021.

[40] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda, Rogerio Feris, et al. AVLnet: Learning audio-visual language representations from instructional videos. In *Interspeech*, 2021.

[41] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[42] Senem Tanberk, Zeynep Hilal Kilimci, Dilek Bilgin Tükel, Mitat Uysal, and Selim Akyokuş. A hybrid deep model using deep learning and dense optical flow approaches for human activity recognition. *IEEE Access*, 2020.

[43] Atousa Torabi, Niket Tandon, and Leonid Sigal. Learning language-visual embedding for movie understanding with natural-language. *arXiv preprint arXiv:1609.08124*, 2016.

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

[45] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval-an empirical odyssey. In *CVPR*, 2019.

[46] Xiaoyang Wang and Qiang Ji. A unified probabilistic approach modeling relationships between attributes and objects. In *ICCV*, 2013.

[47] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019.

[48] Yang Wang and Greg Mori. A discriminative latent model of object classes and attributes. In *ECCV*, 2010.

[49] Michael Wray and Dima Damen. Learning visual actions using multiple verb-only labels. In *BMVC*, 2019.

[50] Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *ICCV*, 2019.

[51] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4video: What can auxiliary captions do for text-video retrieval? In *CVPR*, 2023.

[52] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[53] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[54] Ran Xu, Caiming Xiong, Wei Chen, and Jason Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, 2015.

[55] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018.

[56] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.

[57] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, 2019.