Use What You Have: Video Retrieval Using Representations From Collaborative Experts

Yang Liu*
yangl@robots.ox.ac.uk
Samuel Albanie*
albanie@robots.ox.ac.uk
Arsha Nagrani*
arsha@robots.ox.ac.uk
Andrew Zisserman
az@robots.ox.ac.uk

Visual Geometry Group
University of Oxford
UK

Abstract

The rapid growth of video on the internet has made searching for video content using natural language queries a significant challenge. Human generated queries for video datasets 'in the wild' vary a lot in terms of degree of specificity, with some queries describing 'specific details' such as the names of famous identities, content from speech, or text available on the screen. Our goal is to condense the multi-modal, extremely high dimensional information from videos into a single, compact video representation for the task of video retrieval using free-form text queries, where the degree of specificity is open-ended.

For this we exploit existing knowledge in the form of pre-trained semantic embeddings which include 'general' features such as motion, appearance, and scene features from visual content, and more 'specific' cues from ASR and OCR which may not always be available, but allow for more fine-grained disambiguation when present. We propose a *collaborative experts* model to aggregate information effectively from these different pre-trained experts. The effectiveness of our approach is demonstrated empirically, setting new state-of-the-art performances on five retrieval benchmarks: MSR-VTT, LSMDC, MSVD, DiDeMo, and ActivityNet, while simultaneously reducing the number of parameters used by prior work. Code and data can be found at www.robots.ox.ac.uk/~vgg/research/collaborative-experts/.

1 Introduction

Videos capture the world in two important ways beyond a simple image: first, video contains temporal information – semantic concepts, actions and interactions evolve over time; Second, video may also contain information from multiple modalities, such as an accompanying audio track. This make videos both richer and more informative, but also more challenging to represent. Our goal in this paper is to embed the information from multiple modalities and multiple time steps of a video segment into a compact fixed-length representation. Such a compact representation can then be used for a number of video understanding tasks, such as video retrieval, clustering and summarisation. In particular, we focus on retrieval; our objective is to be able to retrieve video clips using a free form text query that may contain both general and specific information.

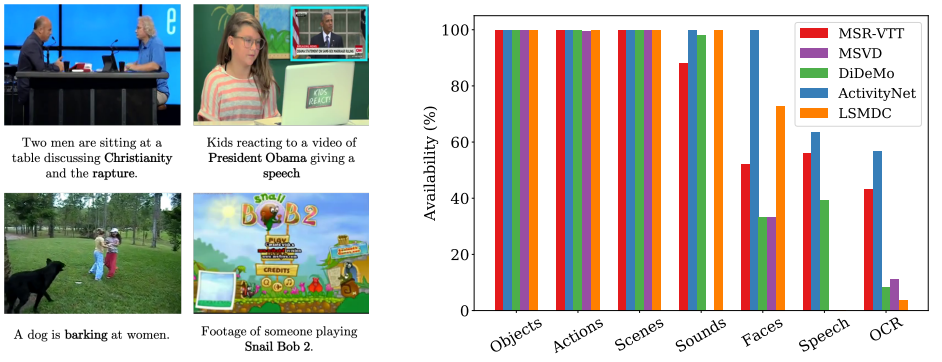

Figure 1: (Left): Unconstrained videos 'in the wild' convey information in various different ways, including (clockwise from top-left), clues from distinctive speech, names of individuals on screen, other text clues embedded in the video and audio. (Right): For the five video datasets considered in this work, the chart portrays the video-level availability of "expert" embeddings from different domains: certain generic embeddings can almost always be extracted via pretrained object/action/scene classification networks while other features such sounds, faces, speech and OCR are less consistently available. However, as we show in Sec. 4.3, these rarer features are often highly discriminative.

Learning a robust and compact representation *tabula rasa* for this task is made extremely challenging by the high dimensionality of the sensory data contained in videos—to do so with discriminative training would require prohibitively expensive textual annotation of a vast number of videos. The primary hypothesis underpinning our approach is the following: *the discriminative content of the multi-modal video embedding can be well approximated by the set of semantic representations of the video data learnt by individual experts (in audio, speech, faces, etc)*. In essence, this approximation enables us to exploit knowledge from existing individual sources where the cost of annotation is significantly reduced (e.g. classification labels for objects in images, identity labels for speech, labels for actions in videos etc.) and where consequently, there exist very large-scale labelled datasets. These large-scale datasets can then be used to train a number of independent "experts" for different perception tasks, which in turn provide a robust, low-dimensional basis for the discriminative query-content approximation described above.

The two key aspects of this idea that we explore in this paper are: (i) *General and specific features:* rather than solely using generic video descriptors (e.g. objects and actions) we also encode quite specific information from the clip, for example text from overlaid captions and text from speech to provide effective coverage of the "queryable content" of the video (Fig. 1, left). While such features may be highly discriminative, they may not always be available (Fig. 1, right) and as we show through experiments (Sec. 4.3), making good use of these cues requires an appropriate model—one that is tolerant to varying levels of noise and conflicts between modalities. We therefore also propose (ii) *Collaborative experts:* a framework that seeks to make effective use of embeddings from different 'experts' (e.g. objects, actions, faces) by learning their combination in order to render them more discriminative. Each expert is filtered via a simple dynamic attention mechanism that considers its relation to all other experts to enable their collaboration. This pairwise approach enables, for instance, the sound of a dog barking to inform the modulation of the RGB features, selecting the features that have encoded the concept of the dog. As we demonstrate in the sequel, both these ideas make a substantial difference to the retrieval performance.

Concretely, we make the following three contributions: (i) We propose the *Collaborative Experts* framework for learning a joint embedding of video and text by combining a collection of

pretrained embeddings into a single, compact video representation. Our joint video embeddings are independent of the retrieval text-query and can be pre-computed offline and indexed for efficient retrieval; (ii) We use both *general* video features such as motion, RGB and audio features, and *specific* video features such as text embedded on screen and speech obtained using OCR and ASR respectively. We show that specific features are highly discriminative but often missing, and our collaborative experts model can deal with this gracefully. (iii) We verify the performance of the resulting representation on a number of retrieval benchmarks, in several cases advancing the state-of-the-art by a considerable margin.

2 Related Work

Cross-Modal Embeddings: A range of prior work has proposed to jointly embed images and text into the same space [17, 18, 19, 29, 39], enabling cross-modal retrieval. More recently, several works have also focused on audio-visual cross-modal embeddings [2, 38], as well as audio-text embeddings [7]. Our goal in this work, however, is to embed videos and natural language sentences (sometimes multiple sentences) into the same semantic space, which is made more challenging by the high dimensional content of videos.

Video-Text Embeddings: While a large number of works [12, 40, 41, 50, 54] have focused on learning visual semantic embeddings for video and language, many of these existing approaches are based on image-text embedding methods by design and typically focus on single visual frames. Mithun el al. [37] observe that a simple adaptation of a state-of-the-art image-text embedding method [17] by mean-pooling features from video frames provides a better result than many prior video-text retrieval approaches [12, 40]. However, such methods do not take advantage of the rich and varied additional information present in videos, including motion dynamics, speech and other background sounds, which may influence the concepts in human captions to a considerable extent. Consequently, there has been a growing interest in fusing information from other modalities—[35, 37] utilise the audio stream (but do not exploit speech content) and use models pretrained for action recognition to extract motion features. These methods do not make use of speech-to-text or OCR for additional cues, which have nevertheless been used successfully to understand videos in other domains, particularly lecture retrieval [44, 55] (where the videos consist of slide shows) and news broadcast [21] retrieval, where a large fraction of the content is displayed on screen in the form of text. Our approach draws particular inspiration from the powerful joint embedding proposed by [35] (which in turn, builds on the classical Mixtures-of-Experts model [27]) and seeks to extend it to take advantage of additional cues (such as speech and text) and make more effective use of pretrained features via the robust collaborative gating mechanism described in Sec. 3.

Annotation scarcity: A key challenge for video-retrieval is the small size of existing training datasets, due to the high cost of annotating videos with natural language. We therefore propose to use the knowledge from existing embeddings pretrained on a wide variety of other tasks. This idea is not new: semantic projections of visual inputs in the form of 'experts' was used by [15] for the task of image retrieval and has also been central to modern video retrieval methods such as [35, 37]. More recently, alternative approaches to addressing the issue of annotation scarcity have been explored, which include self-supervised [49] and weakly-supervised [60] video-text models.

3 Collaborative Experts

Given a set of videos with corresponding text captions, we would like to create a pair of functions ϕ_v and ϕ_t that map sensory video data and text into a joint embedding space that respects this

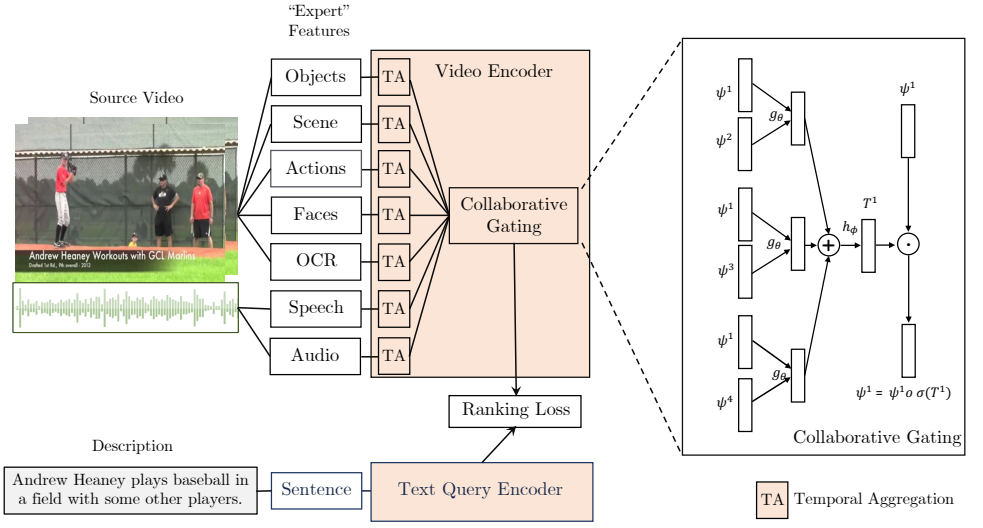

Figure 2: (Left): The proposed Collaborative Experts framework for learning a joint video-text embedding (coloured boxes denote learnable parameters). The information provided by each pretrained "expert" is temporally aggregated as it enters the video encoder and then refined through the use of a collaborative gating mechanism (right) to obtain the video-embedding (for visual clarity, we show the interaction of just a single expert with three others, though in practice all experts are used—see Sec. 3.1 for details). Note that to maintain retrieval efficiency, collaboration occurs only between video experts (the text-query and video embeddings are computed independently).

correspondence—embeddings for paired text and video should lie close together, while embeddings for text and video that do not match should lie far apart. We would also like ϕ_v and ϕ_t to be independent of each other to enable efficient retrieval: the process of querying then reduces to a distance comparison between the embedding of the query and the embeddings of the collection to be searched (which can be pre-computed offline). The proposed Collaborative Experts framework for learning these functions is illustrated in Fig. 2. In this work, we pay particular attention to the design of the video encoder ϕ_v and the process of combining information from different video modalities (Sec. 3.1). To complete the framework, we then discuss how the query text is encoded and the ranking loss used to learn the joint embedding space (Sec. 3.2).

3.1 Video Encoder

To construct the video encoder ϕ_v, we draw on a collection of pretrained, single-modality experts. These operate on the video sensory data $\mathbf{v}$ and project it to a collection of n variable-length task-specific embeddings $\{\Psi^{(1)}_{\text{var}}(\mathbf{v}), \ldots, \Psi^{(n)}_{\text{var}}(\mathbf{v})\}$. Here $\Psi^{(i)}_{\text{var}}$ represents the i^{th} expert (we use the "var" subscript to denote a variable length output when applied to a sequence of frames) whose parameters have been learned on a prior task such as object classification and then frozen. Each element of this collection is then aggregated along its temporal dimension to produce fixed-size, task-specific embeddings per video $\{\Psi^{(1)}(\mathbf{v}),\ldots,\Psi^{(n)}(\mathbf{v})\}$. Any temporal aggregation function may be used here—in this work, we use simple average pooling to aggregate "slow"

visual features such as objects and scenes, and NetVLAD [3] to aggregate more dynamic audio and word features (see Sec. 4.1 for further details). Next, to enable their combination, we apply linear projections to transform these task-specific embeddings to a common dimensionality. Our goal when fusing the resulting representations together into a single condensed video representation is to capture the valuable complementary information between task-specific projections while simultaneously filtering out irrelevant noise and resolving individual expert conflicts on a *per-sample basis*. To do so, this we propose a collaborative gating module, described next.

Collaborative Gating: The collaborative gating module comprises two operations: (1) Prediction of attention vectors for every expert projection $T = \{T^{(1)}(\mathbf{v}),...,T^{(n)}(\mathbf{v})\}$; and (2) modulation of expert responses. Inspired by the relational reasoning module proposed by [46] for visual question answering, we define the attention vector of the i^{th} expert projection T_i as follows:

$$T^{(i)}(\mathbf{v}) = h_\phi(\sum_{j \neq i} g_\theta(\Psi^{(i)}(\mathbf{v}), \Psi^{(j)}(\mathbf{v}))), \tag{1}$$

where functions h_ϕ and g_θ are used to model the pairwise relationship between projection $\Psi^{(i)}$ and projection $\Psi^{(j)}$. Of these, g_θ is used to infer pairwise task relationships, while h_ϕ maps the sum of all pairwise relationships into a single attention vector. In this work, we instantiate both h_ϕ and g_θ as multi-layer perceptrons (MLPs). Note that the functional form of Equation (1) dictates that the attention vector of any expert projection should consider the potential relationships between all pairs associated with this expert. That is to say, the quality of each expert $\Psi^{(j)}$ should contribute in determining and selecting the information content from $\Psi^{(i)}$ in the final decision. It is also worth noting that the collaborative gating module uses the same functions g_θ and h_ϕ (shared weights) to compute all pairwise relationships. This mode of operation encourages greater generalisation, since g_θ and h_ϕ are encouraged not to over-fit to features of any particular pair of tasks. After the attention vectors $T = \{T^{(1)}(\mathbf{v}),...,T^{(n)}(\mathbf{v})\}$ have been computed, each expert projection is modulated follows:

$$\Psi^{(i)}(\mathbf{v}) = \Psi^{(i)}(\mathbf{v}) \circ \sigma(T^{(i)}(\mathbf{v})), \tag{2}$$

where σ is an element-wise sigmoid activation and $\circ$ is the element-wise multiplication (Hadamard product). This gating function re-calibrates the strength of different activations of $\Psi^{(i)}(\mathbf{v})$ and selects which information is highlighted or suppressed, providing the model with a powerful mechanism for dynamically filtering content from different experts. A diagram of the mechanism is shown in Fig. 2 (right).

The final video embedding is then obtained by passing the modulated responses of each expert through a Gated Embedding Module [35] before concatenating the outputs together into a single fixed-length vector and L2-normalising the result.

3.2 Text Query Encoder and Training Loss

To construct the text embeddings, query sentences are first mapped to a sequence of feature vectors with pretrained contextual word-level embeddings (see Sec. 4.1 for details)—as with the video experts, the parameters of this first stage are frozen. These are then aggregated, again using NetVLAD [3] (we also experimented with GRUs [9] for text aggregation, and found performance to be similar). Following aggregation, we follow the text encoding architecture proposed by [35], which projects the aggregated features to separate subspaces for each expert using Gated

Embedding Modules [34]. Each projection is then scaled by a mixture weight (one scalar weight per expert projection), which is computed by applying a single linear layer to the aggregated text-features, and passing the result through a softmax to ensure that the mixture weights sum to one (see [35] for further details). Finally, the scaled outputs are concatenated, producing a vector of dimensionality that matches that of the video embedding (and is likewise L2-normalised).

With the video encoder ϕ_v and text encoder ϕ_t as described, the similarity s_i^j of the i^{th} video, $\mathbf{v}_i$, and the and j^{th} caption, $\mathbf{t}_j$, can then be directly computed as the cosine of the angle between their respective embeddings $\phi_v(\mathbf{v}_i)^T \phi_t(\mathbf{t}_j)$. During optimisation, the parameters of the video encoder (including the collaborative gating module) and text query encoder (the coloured regions of Fig. 2) are learned jointly. Training proceeds by sampling a sequence of minibatches of corresponding video-text pairs $\{\mathbf{v}_i, \mathbf{t}_i\}_{i=1}^{N_B}$ and minimising a *Bidirectional Max-margin Ranking Loss* [48]:

$$\mathcal{L}_r = \frac{1}{N_B} \sum_{i=1, j\neq i}^{N_B} \max(0, m + s_i^j - s_i^i) + \max(0, m + s_j^i - s_i^i) \tag{3}$$

where N_B is the batch size, and m is a fixed constant which is set as a hyperparameter. When assessing retrieval performance, at test time the embedding distances are simply computed via their inner product, as described above.

3.2.1 Missing Experts

When a set of expert features are missing, such as when there is no speech in the audio track, we simply zero-pad the missing experts when estimating the similarity score. To compensate for the implicit scaling introduced by missing experts (the similarity is effectively computed between shorter embeddings), we follow the elegant approach proposed by [35] and simply remove the mixture weights for missing experts, then renormalise the remaining weights such that they sum to one.

4 Experiments

In this section, we evaluate our model on five benchmarks for video retrieval tasks. The description of datasets, implementation details and evaluation metric are provided in Sec. 4.1. A comprehensive comparison on general video retrieval benchmarks is reported in Sec. 4.2. We present an ablation study in Sec. 4.3 to explore how the performance of the proposed method is affected by different model configurations, including the aggregation methods, importance of different experts and number of captions in training.

4.1 Datasets, Implementation Details and Metrics

Datasets: We perform experiments on five video datasets: MSR-VTT [53], LSMDC [45], MSVD [8], DiDeMo [1] and ActivityNet-captions [31], covering a challenging set of domains which include videos from YouTube, personal collections and movies.

Expert Features: In order to capture the rich content of a video, we draw on existing powerful representations for a number of different semantic tasks. These are first extracted at a frame-level, then aggregated to produce a single feature vector per modality per video. *RGB "object"* frame-level embeddings of the visual data are generated with an SENet-154 model [24] (pretrained on ImageNet for the task of image classification). *Motion* embeddings are generated using the I3D inception model [6]. *Face* embeddings are extracted in two stages: (1) Each frame is passed through

an SSD face detector [4, 32] to extract bounding boxes; (2) The image region of each box is passed through a ResNet50 [22] that has been trained for the task of face classification on the VGGFace2 dataset [5]. *Audio* embeddings are obtained with a VGGish model, trained for audio classification on the YouTube-8m dataset [23]. *Speech-to-Text* features are extracted using the Google Cloud speech API, to extract word tokens from the audio stream, which are then encoded via pretrained word2vec embeddings [36]. *Optical Character Recognition* is done in two stages: (1) Each frame is passed through the Pixel Link [10] text detection model to extract bounding boxes for text; (2) The image region of each box is passed through a model [33] that has been trained for scene text recognition on the Synth90K dataset[26]. The text is then encoded via a pretrained word2vec embedding model [36].

Temporal Aggregation: We adopt a simple approach to aggregating the features described above. For appearance, motion, scene and face embeddings, we average frame level features along the temporal dimension to produce a single feature vector per video (we found max-pooling to perform similarly). For speech, audio and OCR features, we adopt the NetVLAD mechanism proposed by [3], which has proven effective in the retrieval setting [34]. As noted in Sec. 3.1, all aggregated features are projected to a common size (512 dimensions).

Text: Each word is encoded using pretrained word2vec word embeddings [36] and then passed through a pretrained OpenAI-GPT model [43] to extract contextual word embeddings. Finally, the word embeddings in each sentence are aggregated using NetVLAD.

Dataset-specific details: Except where noted otherwise for ablation purposes, we use each of the embeddings described above for the MSR-VTT, ActivityNet and DiDeMo datasets. For MSVD, we extract the subset of features which do not require an audio stream (since no audio is available with the dataset). For LSMDC, we re-use the existing appearance, motion, face, text and audio features made available by [35], and combine them with the scene and OCR features described above.

Training Details: The CE framework is implemented with PyTorch [42]. Optimisation is performed with the Adam solver [28]. Where possible, we follow learning rate/batch size settings in prior work. Thus, for MSR-VTT, we train with batch size 64 and learning rate 0.0004 and for LSMDC we use a batch size of 512 and a learning rate of 0.0001 (both following [35]). In both cases, a learning rate decay of 0.05 is applied per epoch. We adopt similar settings as for MSR-VTT when training on DiDeMo and ActivityNet. For MSVD (a dataset with many fewer videos), we use a smaller batch size of 16 and a learning rate of 0.002. The value of the ranking loss margin hyperparameter m in Eq. 3 is fixed to 0.2 for all experiments.

Evaluation Metrics: We follow prior work (e.g. [12, 35, 37, 57, 58]) and report standard retrieval metrics (where existing work enables comparison) including median rank (lower is better), mean rank (lower is better) and R@K (recall at rank K—higher is better). When computing video-to-sentence metrics for datasets with multiple independent sentences per video (MSR-VTT and MSVD), we follow the evaluation protocol used in prior work [13, 14, 37] which corresponds to reporting the minimum rank among all valid text descriptions for a given video query. For each benchmark, we report the mean and standard deviation of three randomly seeded runs.

4.2 Comparison to Prior State-of-the-Art

We first compare the proposed method with the existing state-of-the-art on the MSR-VTT benchmark for the tasks of sentence-to-video and video-to-sentence retrieval Tab. 1. We observe that Collaborative Experts (CE) consistently improves retrieval performance for both sentence and video queries—for instance, improving R@1 by almost a factor of three on the full test set. In Tab. 2, we compare with prior work on the ActivityNet paragraph-video retrieval benchmark (note that we compare to methods which use the same level of annotation as our approach i.e. video-level annota-

		Text ⟹ Video					Video ⟹ Text				
Method	Test-set	R@1	R@5	R@10	MdR	MnR	R@1	R@5	R@10	MdR	MnR
JSFusion [57]	1k-A	10.2	31.2	43.2	13	-	-	-	-	-	-
CE	1k-A	**39.4**$_{\pm 0.6}$	**75.2**$_{\pm 0.6}$	**85.1**$_{\pm 0.4}$	**2**$_{\pm 0}$	**7.5**$_{\pm 0.2}$	**40.5**$_{\pm 0.7}$	**74.8**$_{\pm 0.4}$	**85.7**$_{\pm 0.9}$	**2**$_{\pm 0}$	**7.4**$_{\pm 0.2}$
MoEE [35] ‡	1k-B	12.9	36.4	51.8	10	-	-	-	-	-	-
MoEE$_{COCO}$ [35] ‡	1k-B	18.6	44.2	59.0	7	-	-	-	-	-	-
CE	1k-B	**32.2**$_{\pm 1.2}$	**64.7**$_{\pm 0.4}$	**75.8**$_{\pm 0.1}$	**3**$_{\pm 0}$	**12.3**$_{\pm 0.1}$	**31.5**$_{\pm 0.7}$	**66.0**$_{\pm 0.7}$	**77.5**$_{\pm 0.7}$	**3**$_{\pm 0}$	**11.4**$_{\pm 0.2}$
VSE [37]	Full	5.0	16.4	24.6	47	215.1	7.7	20.3	31.2	28	185.8
VSE++ [37]	Full	5.7	17.1	24.8	65	300.8	10.2	25.4	35.1	25	228.1
Mithun et al. [37]	Full	7.0	20.9	29.7	38	213.8	12.5	32.1	42.4	16	134.0
W2VV [13]	Full	6.1	18.7	27.5	45	-	11.8	28.9	39.1	21	-
Dual Encoding [14]	Full	7.7	22.0	31.8	32	-	13.0	30.8	43.3	15	-
CE	Full	**22.5**$_{\pm 0.1}$	**52.1**$_{\pm 0.2}$	**65.5**$_{\pm 0.1}$	**5**$_{\pm 0}$	**22.5**$_{\pm 0.1}$	**34.4**$_{\pm 0.4}$	**64.6**$_{\pm 0.3}$	**77.0**$_{\pm 0.4}$	**3**$_{\pm 0}$	**13.2**$_{\pm 0.6}$

Table 1: Retrieval with sentences and videos on the MSR-VTT dataset. R@k denotes recall@k (higher is better), MdR and MnR denote median rank and mean rank resp. (lower is better). Standard deviations are reported from three randomly seeded runs. 1k-A and 1k-B denote test sets of 1000 randomly sampled text-video pairs used by [57] and [35] resp. ‡ Updated code-base released by Miech et al. [35] which achieves stronger results.

tion). On all metrics, CE outperforms all prior work, in several cases by considerable large margins. We next evaluate the performance of the CE framework on the LSMDC benchmark for sentence-to-video retrieval (Tab. 3, left) and observe that CE matches or outperforms all prior work, including the prior state-of-the-art method [35] which incorporates additional training images and captions from the COCO benchmark during training. We observe similar trends in the results for the MSVD retrieval benchmark (Tab. 3, right). Finally, in Tab. 4 we provide a comparison with previously reported numbers on the DiDeMo benchmark and see that CE again strongly outperforms prior work.

	Text ⟹ Video					Video ⟹ Text				
Method	R@1	R@5	R@50	MdR	MnR	R@1	R@5	R@50	MdR	MnR
LSTM-YT [52] ([58])	0.0	4.0	24.0	102	-	0.0	7.0	38.0	98	-
NOCTXT [51] ([58])	5.0	14.0	32.0	78	-	7.0	18.0	45.0	56	-
FSE [58]	18.2$_{\pm 0.2}$	44.8$_{\pm 0.4}$	89.1$_{\pm 0.3}$	7	-	16.7$_{\pm 0.8}$	43.1$_{\pm 1.1}$	88.4$_{\pm 0.3}$	7	-
DENSE [31]	14.0	32.0	65.0	34	-	18.0	36.0	74.0	32	
HSE(4SEGS) [58]	20.5	49.3	-	-	-	18.7	48.1	-	-	
CE	**27.3**$_{\pm 0.7}$	**61.1**$_{\pm 1.0}$	**94.4**$_{\pm 0.1}$	**4**$_{\pm 0}$	**15.4**$_{\pm 0.9}$	**27.9**$_{\pm 0.6}$	**61.6**$_{\pm 0.4}$	**95.0**$_{\pm 0.2}$	**3.3**$_{\pm 0.6}$	**14.5**$_{\pm 0.8}$

Table 2: Comparison of paragraph-video retrieval methods trained with video-level information on the ActivityNet-captions dataset (val1 test-split).

	Text ⟹ Video			
Method	R@1	R@5	R@10	MdR
Yu et al. [56]†	3.6	14.7	23.9	50
CCA [30] (rep. by [35])	7.5	21.7	31.0	33
JSFusion [57]‡	9.1	21.2	34.1	36
MoEE [35]	11.6	28.0	37.6	22
MoEE$_{COCO}$ [35]	**12.7**	28.9	39.6	21
CE	12.6$_{\pm 0.8}$	**31.0**$_{\pm 0.6}$	**40.3**$_{\pm 0.5}$	**19**$_{\pm 0}$

	Text ⟹ Video				
Method	R@1	R@5	R@10	MdR	MnR
CCA ([54])	-	-	-	-	245.3
JMDV [54]	-	-	-	-	236.3
VSE [29] ([37])	12.3	30.1	42.3	14	57.7
VSE++ [16] ([37])	15.4	39.6	53.0	9	43.8
Multi. Cues [37]	20.3	47.8	61.1	**6**	28.3
CE	**20.9**$_{\pm 0.5}$	**49.7**$_{\pm 0.4}$	**63.8**$_{\pm 0.4}$	**5.7**$_{\pm 0.6}$	**19.5**$_{\pm 0.9}$

Table 3: Text-to-Video retrieval results on the LSMDC dataset (left) and the MSVD dataset (right). †,‡ denote the winners of the 2016 and 2017 LSMDC challenges, respectively.

	Text ⟹ Video					Video ⟹ Text				
Method	R@1	R@5	R@50	MdR	MnR	R@1	R@5	R@50	MdR	MnR
S2VT [51] ([58])	11.9	33.6	76.5	13	-	13.2	33.6	76.5	15	-
FSE [58]	$13.9_{\pm 0.7}$	$36_{\pm 0.8}$	$78.9_{\pm 1.6}$	11	-	$13.1_{\pm 0.5}$	$33.9_{\pm 0.4}$	$78.0_{\pm 0.8}$	12	-
CE	$\mathbf{22.6}_{\pm 0.5}$	$\mathbf{51.1}_{\pm 1.0}$	$\mathbf{87.3}_{\pm 0.3}$	$\mathbf{5}_{\pm 0}$	$27.4_{\pm 1.1}$	$\mathbf{22.5}_{\pm 1.3}$	$\mathbf{52.3}_{\pm 0.8}$	$\mathbf{89.2}_{\pm 0.8}$	$\mathbf{5}_{\pm 0}$	$23.2_{\pm 1.1}$

Table 4: Comparison of paragraph-video retrieval methods trained with video-level information on the DiDeMo dataset.

4.3 Ablation Studies

In this section, we provide ablation studies to empirically assess: (1) the effectiveness of the proposed collaborative experts framework vs other aggregation strategies; (2) the importance of using of a diverse range of experts with differing levels of specificity; (3) the relative value of using experts in comparison to simply having additional annotated training data.

Aggregation method: We compare the use of collaborative experts with several other baselines (with access to the same experts) for embedding aggregation including: (1) simple expert concatenation; (2) expert concatenation to a single high-dimensional embedding, followed immediately by a large Gated Embedding Module [35]; (3) the state of the art MoEE [35] method and (4) CE without the use of the collaborative gating module described in Sec. 3.1. The results, presented in Tab. 6 (left), demonstrate the significant contribution of collaborative gating which improves R@1 performance by more than 60% and leads to a model with fewer parameters than the prior state of the art.

Importance of different experts: The value of different experts is assessed in Tab. 5 (note that since several experts are not present in all videos, we combine them with features produced by a model pretrained on ImageNet [11] (an RGB "objects" expert) as a baseline to enable a more meaningful comparison). While all experts boost performance, there is considerable variance in their effect. The scene expert brings minimal gains in combination with ImageNet features, which suggests it provides limited additional information to this pairing, and therefore, the set of experts as a whole. In this particular case, it is likely because while ImageNet primarily contains object classes, it also contains a small number of scenes such as "seashore" and "valley". Other experts, such as face features may be often missing from videos (see Fig.1, right), but the results demonstrate that they can nevertheless be highly discriminative and provide a useful complementary signal. We additionally perform a second study in which we incrementally add experts to the CE framework, according to the performance they obtained in the previous experiment. The results demonstrate not only that the different experts contain useful information as a group, but also that the collaborative experts framework is able to make good use of this information to learn a powerful joint embedding.

	Text ⟹ Video				
Experts	R@1	R@5	R@10	MdR	MnR
RGB	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$	$182.0_{\pm 1.3}$
RGB+Scene	$5.0_{\pm 0.0}$	$16.6_{\pm 0.1}$	$25.4_{\pm 0.3}$	$40.0_{\pm 1.0}$	$164.4_{\pm 1.4}$
RGB+Motion	$5.6_{\pm 0.1}$	$18.4_{\pm 0.1}$	$28.1_{\pm 0.1}$	$33.0_{\pm 0.0}$	$144.2_{\pm 0.8}$
RGB+Audio	$7.1_{\pm 0.1}$	$22.1_{\pm 0.2}$	$32.5_{\pm 0.1}$	$25.0_{\pm 0.0}$	$106.4_{\pm 0.0}$
RGB+OCR	$7.1_{\pm 0.1}$	$22.6_{\pm 0.1}$	$33.4_{\pm 0.1}$	$24.3_{\pm 0.6}$	$103.3_{\pm 1.8}$
RGB+Speech	$7.2_{\pm 0.1}$	$22.6_{\pm 0.2}$	$33.4_{\pm 0.3}$	$24.7_{\pm 0.6}$	$101.9_{\pm 2.3}$
RGB+Face	$7.5_{\pm 0.2}$	$23.1_{\pm 0.2}$	$34.0_{\pm 0.1}$	$24.0_{\pm 0.0}$	$101.0_{\pm 0.6}$

	Text ⟹ Video				
Experts	R@1	R@5	R@10	MdR	MnR
RGB	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$	$182.0_{\pm 1.3}$
Previous+Scene	$5.0_{\pm 0.0}$	$16.6_{\pm 0.1}$	$25.4_{\pm 0.3}$	$40.0_{\pm 1.0}$	$164.4_{\pm 1.4}$
Previous+Motion	$5.8_{\pm 0.1}$	$18.6_{\pm 0.1}$	$28.2_{\pm 0.1}$	$33.0_{\pm 0.0}$	$142.9_{\pm 0.5}$
Previous+Audio	$7.8_{\pm 0.2}$	$23.8_{\pm 0.0}$	$38.8_{\pm 0.1}$	$22.0_{\pm 0.0}$	$96.0_{\pm 0.2}$
Previous+OCR	$12.0_{\pm 0.1}$	$33.4_{\pm 0.3}$	$46.2_{\pm 0.1}$	$13.0_{\pm 0.0}$	$56.9_{\pm 0.4}$
Previous+Speech	$16.0_{\pm 0.0}$	$41.4_{\pm 0.3}$	$55.1_{\pm 0.3}$	$8.0_{\pm 0.0}$	$37.7_{\pm 0.7}$
Previous+Face	$22.5_{\pm 0.1}$	$52.1_{\pm 0.2}$	$65.5_{\pm 0.1}$	$5.0_{\pm 0.0}$	$22.5_{\pm 0.1}$

Table 5: **The importance of different experts** (Left): The value of different experts in combination with a baseline set of for text-video retrieval and (right) their cumulative effect on MSR-VTT.

Number of Captions in training: An emerging idea in our community is that many machine perception tasks might be solved through the combination of simple models and large-scale training

Aggreg.	R@1	R@5	R@10	MdR	Params
Concat	$0.5_{\pm 0.1}$	$2.4_{\pm 0.2}$	$4.5_{\pm 0.3}$	$253.3_{\pm 5.7}$	318.2 K
Concat + G	$6.8_{\pm 0.1}$	$20.4_{\pm 0.1}$	$30.1_{\pm 0.3}$	$32.0_{\pm 0.0}$	295.8 M
MoEE [35]	$13.6_{\pm 0.0}$	$36.2_{\pm 0.0}$	$49.5_{\pm 0.1}$	$11.0_{\pm 0.0}$	143.4 M
CE w/o CG	$13.5_{\pm 0.1}$	$36.0_{\pm 0.2}$	$49.0_{\pm 0.2}$	$11.0_{\pm 0.0}$	105.9 M
CE	$22.5_{\pm 0.1}$	$52.1_{\pm 0.2}$	$65.5_{\pm 0.1}$	$5.0_{\pm 0.0}$	107.7 M

Expert	Num. Captions	R@1	R@5	R@10	MdR
RGB	1	$2.4_{\pm 0.1}$	$9.0_{\pm 0.1}$	$14.9_{\pm 0.1}$	$92.7_{\pm 2.5}$
RGB	20	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$
All	1	$11.9_{\pm 0.4}$	$33.9_{\pm 0.6}$	$47.0_{\pm 0.3}$	$12.0_{\pm 0.0}$
All	20	$22.5_{\pm 0.1}$	$52.1_{\pm 0.2}$	$65.5_{\pm 0.1}$	$5.0_{\pm 0.0}$

Table 6: (Left): Aggregation methods for text-video retrieval on MSR-VTT; (Right): The relative value of training with additional captions vs the value of experts.

sets, reminiscent of the "big-data" hypothesis [20]. In this section, we perform an ablation study to assess the relative importance of access to pretrained experts and additional video description annotations. To do so, we measure the performance of the CE model as we vary (1) the number of descriptions available per-video during training and (2) the number of experts it has access to. The results are shown in Tab. 6. We observe that increasing the number of training captions per-video from 1 to 20 brings an improvement in performance, but considerably less so than adding in our full collection of experts, suggesting that indeed, adding experts can help to compensate for a paucity of labelled data. When multiple captions are multiple experts are both available, they naturally lead to the most robust embedding. Some qualitative examples of videos retrieved by the multiple-expert, multiple-caption are provided in Fig. 3.

Figure 3: **Qualitative Results on MSR-VTT:** For each query, we show frames from the top three ranked videos (where present, the ground truth video is indicated by a green box around the similarity score). Top row: (left) Even for imperfect rankings, the model retrieves reasonable videos; Failure case (right) the embeddings can occasionally fail to differentiate between subtle signals (in this case American football vs soccer). Bottom row: (left) the videos retrieved by the proposed model (which correctly assigns the highest similarity to the correct video); (right) removing the proposed CE component produces a nosier ranking.

5 Conclusion

In this work, we introduced collaborative experts, a framework for learning a joint video-text embedding for efficient retrieval. We have shown that using a range of pretrained features and combining them through an appropriate gating mechanism can significantly boost retrieval performance on a range of benchmarks. In future work, we plan to explore the use of collaborative experts for other video understanding tasks such as clustering and summarisation.

Acknowledgements: Funding for this research is provided by the EPSRC Programme Grant Seebibyte EP/M013774/1 and EPSRC grant EP/R03298X/1. A.N. is supported by a Google PhD Fellowship. We would like to thank Antoine Miech, YoungJae Yu and Bowen Zhang for their assistance. We would also like to thank Zak Stone and Susie Lim for their help with cloud computing.

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5803–5812, 2017.

[2] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 609–617, 2017.

[3] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016.

[4] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[5] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *Proc. Int. Conf. Autom. Face and Gesture Recog.*, 2018.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[7] Gal Chechik, Eugene Ie, Martin Rehn, Samy Bengio, and Dick Lyon. Large-scale content-based audio retrieval from text queries. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 105–112. ACM, 2008.

[8] David L Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 190–200. Association for Computational Linguistics, 2011.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[10] Dan Deng, Haifeng Liu, Xuelong Li, and Deng Cai. Pixellink: Detecting scene text via instance segmentation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[12] Jianfeng Dong, Xirong Li, and Cees GM Snoek. Word2visualvec: Image and video to sentence matching by visual feature prediction. *arXiv preprint arXiv:1604.06838*, 2016.

[13] Jianfeng Dong, Xirong Li, and Cees GM Snoek. Predicting visual features from text for image and video caption retrieval. *IEEE Transactions on Multimedia*, 20(12):3377–3388, 2018.

[14] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, and Xun Wang. Dual dense encoding for zero-example video retrieval. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[15] Matthijs Douze, Arnau Ramisa, and Cordelia Schmid. Combining attributes and fisher vectors for efficient image retrieval. In *CVPR 2011*, pages 745–752. IEEE, 2011.

[16] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612*, 2(7):8, 2017.

[17] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612*, 2(7):8, 2017.

[18] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *European conference on computer vision*, pages 15–29. Springer, 2010.

[19] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

[20] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. 2009.

[21] Alexander G Hauptmann, Rong Jin, and Tobun Dorbin Ng. Multi-modal information retrieval from broadcast video using ocr and speech recognition. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 160–161. ACM, 2002.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[23] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017. URL https://arxiv.org/abs/1609.09430.

[24] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[25] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[26] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

[27] Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.

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

[29] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[30] Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. Associating neural word embeddings with deep image representations using fisher vectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4437–4446, 2015.

[31] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 706–715, 2017.

[32] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[33] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Synthetically supervised feature learning for scene text recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 435–451, 2018.

[34] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. *arXiv preprint arXiv:1706.06905*, 2017.

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

[36] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[37] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 19–27. ACM, 2018.

[38] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Learnable pins: Cross-modal embeddings for person identity. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 71–88, 2018.

[39] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 299–307, 2017.

[40] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. Learning joint representations of videos and sentences with web image search. In *European Conference on Computer Vision*, pages 651–667. Springer, 2016.

[41] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4594–4602, 2016.

[42] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[43] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.

[44] N Radha. Video retrieval using speech and text in video. In *2016 International Conference on Inventive Computation Technologies (ICICT)*, volume 2, pages 1–6. IEEE, 2016.

[45] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3202–3212, 2015.

[46] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[47] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2017.

[48] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014.

[49] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. *arXiv preprint arXiv:1904.01766*, 2019.

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

[51] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.

[52] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[53] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5288–5296, 2016.

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

[55] Natsuo Yamamoto, Jun Ogata, and Yasuo Ariki. Topic segmentation and retrieval system for lecture videos based on spontaneous speech recognition. In *Eighth European Conference on Speech Communication and Technology*, 2003.

[56] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. Video captioning and retrieval models with semantic attention. *arXiv preprint arXiv:1610.02947*, 6(7), 2016.

[57] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 471–487, 2018.

[58] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 374–390, 2018.

[59] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[60] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. *arXiv preprint arXiv:1903.08225*, 2019.

A Supplementary Material

A.1 Detailed Description of Datasets

MSR-VTT [53]: This large-scale dataset comprises approximately 200K unique video-caption pairs (10K YouTube video clips, each accompanied by 20 different captions). The dataset is

particularly useful because it contains a good degree of video diversity, but we noted a reasonably high degree of label noise (there are a number of duplicate annotations in the provided captions). The dataset allocates 6513, 497 and 2990 videos for training, validation and testing, respectively. To enable a comparison with as many methods as possible, we also report results across other train/test splits used in prior work [35, 57]. In particular, when comparing with [35] (on splits which do not provide a validation set), we follow their evaluation protocol, measuring performance after training has occurred for a fixed number of epochs (100 in total).

MSVD [8]: The MSVD dataset contains 80K English descriptions for 1,970 videos sourced from YouTube with a large number of captions per video (around 40 sentences each). We use the standard split of 1,200, 100, and 670 videos for training, validation, and testing [52, 54][1]. Differently from the other datasets, the MSVD videos do not have audio streams.

LSMDC [45]: This dataset contains 118,081 short video clips extracted from 202 movies. Each video has a caption, either extracted from the movie script or from transcribed DVS (descriptive video services) for the visually impaired. Evaluation is performed on a test set consisting of 1000 videos from movies disjoint from the training set, as outlined by the Large Scale Movie Description Challenge (LSMDC). [2]

ActivityNet-captions [31]: ActivityNet Captions consists of 20K videos from YouTube, coupled with approximately 100K descriptive sentences. We follow the paragraph-video retrieval protocols described in [58], training up to 15 epochs and reporting performance on `val1` (this train/test split allocates 10,009 videos for training and 4,917 videos for testing).

DiDeMo [1]: DiDeMo contains 10,464 unedited, personal videos in diverse visual settings with roughly 3-5 pairs of descriptions and distinct moments per video. The videos are collected in an open-world setting and include diverse content such as pets, concerts, and sports games. The total number of sentences is 40,543. While the moments are localised with time-stamp annotations, we do not use time stamps in this work.

A.2 Ablation Studies - Full Tables

	Text $\Longrightarrow$ Video					Video $\Longrightarrow$ Text				
Experts	R@1	R@5	R@10	MdR	MnR	R@1	R@5	R@10	MdR	MnR
RGB	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$	$182.0_{\pm 1.3}$	$6.3_{\pm 0.5}$	$19.5_{\pm 0.3}$	$29.3_{\pm 0.1}$	$32.2_{\pm 0.3}$	$213.7_{\pm 5.5}$
RGB+Scene	$5.0_{\pm 0.0}$	$16.6_{\pm 0.1}$	$25.4_{\pm 0.3}$	$40.0_{\pm 1.0}$	$164.4_{\pm 1.4}$	$7.4_{\pm 0.4}$	$23.1_{\pm 0.5}$	$33.6_{\pm 0.4}$	$25.0_{\pm 1.0}$	$166.4_{\pm 2.0}$
RGB+Motion	$5.6_{\pm 0.1}$	$18.4_{\pm 0.1}$	$28.1_{\pm 0.1}$	$33.0_{\pm 0.0}$	$144.2_{\pm 0.8}$	$9.1_{\pm 0.2}$	$27.2_{\pm 0.3}$	$38.6_{\pm 0.9}$	$19.0_{\pm 0.0}$	$121.2_{\pm 4.7}$
RGB+Audio	$7.1_{\pm 0.1}$	$22.1_{\pm 0.2}$	$32.5_{\pm 0.1}$	$25.0_{\pm 0.0}$	$106.4_{\pm 0.0}$	$11.7_{\pm 0.2}$	$31.4_{\pm 0.1}$	$43.5_{\pm 0.7}$	$14.7_{\pm 0.6}$	$79.3_{\pm 1.9}$
RGB+OCR	$7.1_{\pm 0.1}$	$22.6_{\pm 0.1}$	$33.4_{\pm 0.1}$	$24.3_{\pm 0.6}$	$103.3_{\pm 1.8}$	$10.4_{\pm 0.3}$	$29.9_{\pm 0.8}$	$42.3_{\pm 0.9}$	$16.0_{\pm 1.0}$	$103.2_{\pm 7.7}$
RGB+Speech	$7.2_{\pm 0.1}$	$22.6_{\pm 0.2}$	$33.4_{\pm 0.3}$	$24.7_{\pm 0.6}$	$101.9_{\pm 2.3}$	$11.1_{\pm 0.2}$	$30.3_{\pm 0.4}$	$42.5_{\pm 0.5}$	$16.0_{\pm 0.0}$	$104.8_{\pm 4.8}$
RGB+Face	$7.5_{\pm 0.2}$	$23.1_{\pm 0.2}$	$34.0_{\pm 0.1}$	$24.0_{\pm 0.0}$	$101.0_{\pm 0.6}$	$11.5_{\pm 0.3}$	$30.6_{\pm 0.4}$	$41.7_{\pm 0.7}$	$15.7_{\pm 0.6}$	$99.9_{\pm 1.4}$

Table 7: Ablation study of importance of each expert when combined with RGB features.

A.3 Implementation Details

Appearance frame-level embeddings of the visual data are generated with a SENet-154 model [24] (pretrained on ImageNet for the task of image classification) from frames extracted at 5fps,

[1] Note: referred to by [37] as the JMET-JMDV split

[2] https://sites.google.com/site/describingmovies/lsmdc-2017

	Text ⟹ Video					Video ⟹ Text				
Experts	R@1	R@5	R@10	MdR	MnR	R@1	R@5	R@10	MdR	MnR
RGB	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$	$182.0_{\pm 1.3}$	$6.3_{\pm 0.5}$	$19.5_{\pm 0.3}$	$29.3_{\pm 0.1}$	$32.2_{\pm 0.3}$	$213.7_{\pm 5.5}$
Previous+Scene	$5.0_{\pm 0.0}$	$16.6_{\pm 0.1}$	$25.4_{\pm 0.3}$	$40.0_{\pm 1.0}$	$164.4_{\pm 1.4}$	$7.4_{\pm 0.4}$	$23.1_{\pm 0.5}$	$33.6_{\pm 0.4}$	$25.0_{\pm 1.0}$	$166.4_{\pm 2.0}$
Previous+Motion	$5.8_{\pm 0.1}$	$18.6_{\pm 0.1}$	$28.2_{\pm 0.1}$	$33.0_{\pm 0.0}$	$142.9_{\pm 0.5}$	$9.1_{\pm 0.2}$	$27.0_{\pm 0.3}$	$39.2_{\pm 0.2}$	$18.7_{\pm 0.6}$	$118.4_{\pm 3.1}$
Previous+Audio	$7.8_{\pm 0.2}$	$23.8_{\pm 0.0}$	$38.8_{\pm 0.1}$	$22.0_{\pm 0.0}$	$96.0_{\pm 0.2}$	$12.9_{\pm 0.3}$	$35.1_{\pm 0.4}$	$48.4_{\pm 0.9}$	$11.7_{\pm 0.6}$	$60.1_{\pm 1.4}$
Previous+OCR	$12.0_{\pm 0.1}$	$33.4_{\pm 0.3}$	$46.2_{\pm 0.1}$	$13.0_{\pm 0.0}$	$56.9_{\pm 0.4}$	$17.9_{\pm 0.2}$	$44.8_{\pm 0.9}$	$58.5_{\pm 0.4}$	$7.0_{\pm 0.0}$	$39.1_{\pm 3.8}$
Previous+Speech	$16.0_{\pm 0.0}$	$41.4_{\pm 0.3}$	$55.1_{\pm 0.3}$	$8.0_{\pm 0.0}$	$37.7_{\pm 0.7}$	$23.0_{\pm 0.4}$	$52.8_{\pm 1.0}$	$66.9_{\pm 1.1}$	$5.0_{\pm 0.0}$	$23.0_{\pm 2.0}$
Previous+Face	$22.5_{\pm 0.1}$	$52.1_{\pm 0.2}$	$65.5_{\pm 0.1}$	$5.0_{\pm 0.0}$	$22.5_{\pm 0.1}$	$34.4_{\pm 0.4}$	$64.6_{\pm 0.3}$	$77.0_{\pm 0.4}$	$3.0_{\pm 0.0}$	$13.2_{\pm 0.6}$

Table 8: Ablation study of the importance experts on the MSR-VTT dataset.

		Text ⟹ Video					Video ⟹ Text				
Expert	Num. Caps	R@1	R@5	R@10	MdR	MnR	R@1	R@5	R@10	MdR	MnR
RGB	1	$2.4_{\pm 0.1}$	$9.0_{\pm 0.1}$	$14.9_{\pm 0.1}$	$92.7_{\pm 2.5}$	$314.1_{\pm 8.2}$	$4.2_{\pm 0.3}$	$14.0_{\pm 0.2}$	$21.6_{\pm 0.6}$	$56.3_{\pm 1.0}$	$308.0_{\pm 3.4}$
RGB	20	$4.3_{\pm 0.0}$	$14.8_{\pm 0.1}$	$23.1_{\pm 0.0}$	$46.3_{\pm 0.6}$	$182.0_{\pm 1.3}$	$6.3_{\pm 0.5}$	$19.5_{\pm 0.3}$	$29.3_{\pm 0.1}$	$32.2_{\pm 0.3}$	$213.7_{\pm 5.5}$
All	1	$11.9_{\pm 0.4}$	$33.9_{\pm 0.6}$	$47.0_{\pm 0.3}$	$12.0_{\pm 0.0}$	$47.5_{\pm 2.4}$	$17.8_{\pm 1.7}$	$44.6_{\pm 2.4}$	$58.8_{\pm 2.3}$	$7.3_{\pm 1.2}$	$31.1_{\pm 4.5}$
All	20	$22.5_{\pm 0.1}$	$52.1_{\pm 0.2}$	$65.5_{\pm 0.1}$	$5.0_{\pm 0.0}$	$22.5_{\pm 0.1}$	$34.4_{\pm 0.4}$	$64.6_{\pm 0.3}$	$77.0_{\pm 0.4}$	$3.0_{\pm 0.0}$	$13.2_{\pm 0.6}$

Table 9: Ablation study of the number of captions in training on MSRVTT

where each frame is resized to 224 × 224 pixels. Features are collected from the final global average pooling layer, and have a dimensionality of 2048.

Motion embeddings are generated using the I3D inception model following the procedure described by [6]. Frames extracted at 25fps and processed with a window length of 64 frames and a stride of 25 frames. Each frame is first resized to a height of 256 pixels (preserving aspect ratio), before a 224 × 224 centre crop is passed to the model. Each temporal window produces a (1024x7)-matrix of features.

Face embeddings are extracted in two stages: (1) Each frame (also extracted at 25 fps) is resized to 300 × 300 pixels and passed through an SSD face detector [4, 32] to extract bounding boxes; (2) The image region of each box is resized such that the minimum dimension is 224 pixels and a centre crop is passed through a ResNet50 [22] that has been trained for task of face classification on the VGGFace2 dataset [5], producing a 512-dimensional embedding for each detected face.

Audio embeddings are obtained with a VGGish model, trained for audio classification on the YouTube-8m dataset [23]. To produce the input for this model, the audio stream of each video is re-sampled to a 16kHz mono signal, converted to an STFT with a window size of 25ms and a hop of 10ms with a Hann window, then mapped to a 64 bin log mel-spectrogram. Finally, the features are parsed into non-overlapping 0.96s collections of frames (each collection comprises 96 frames, each of 10ms duration), which is mapped to a 128-dimensional feature vector.

Scene embeddings of 2208 dimensions are extracted from 224×224 pixel centre crops of frames extracted at 1fps using a DenseNet-161 [25] model pretrained on Places365 [59]. **Speech to Text** The audio stream of each video is re-sampled to a 16kHz mono signal. We then obtained transcripts of the spoken speech for MSRVTT, MSVD and ActivityNet using the the Google Cloud Speech to Text API [3] from the resampled signal. The language for the API is specified as English. For reference, of the 10,000 videos contained in MSRVTT, 8,811 are accompanied by audio streams. Of these, we detected speech in 5,626 videos.

Optical Character Recognition are extracted in two stages: (1) Each frame is resized to 800×400 pixels) and passed through Pixel Link [10] text detection model to extract bounding boxes for texts;

[3] https://cloud.google.com/speech-to-text/

(2) The image region of each box is resized to 32×256 and then pass through a model [33, 47] that has been trained for text of scene text recognition on the Synth90K dataset[26], producing a character sequence for each detect box. They are then encoded via a pretrained word2vec embedding model [36].

Text We encode each word using the Google News[4] trained word2vec word embeddings [36]. All the word embeddings are then pass through a pretrained OpenAI-GPT model to extract the context-specific word embeddings (i.e., not only learned based on word concurrency but also the sequential context). Finally, all the word embeddings in each sentence are aggregated using NetVLAD.

[4] GoogleNews-vectors-negative300.bin.gz found at: https://code.google.com/archive/p/word2vec/