# Renmin University of China and Zhejiang Gongshang University at TRECVID 2019: Learn to Search and Describe Videos

Xirong Li[†], Jinde Ye[*], Chaoxi Xu[†], Shanjinwen Yun[†],
Leimin Zhang[*], Xun Wang[*], Rui Qian[†], Jianfeng Dong[*]

[†]AI & Media Computing Lab
Renmin University of China
Beijing, China

[*]School of Computer and Information Engineering
Zhejiang Gongshang University
Hangzhou, China

## Abstract

*In this paper we summarize our TRECVID 2019 [1] video retrieval experiments. We participated in two tasks: Ad-hoc Video Search (AVS) and Video-to-Text (VTT). For the AVS task, we develop our solutions based on two deep learning models,* i.e. *the W2VV++ network [11] and the Dual Encoding Network [7]. For the VTT Matching and Ranking subtask, our entry is also based on the W2VV++ and Dual Encoding Networks. For the VTT Description Generation subtask, we enhance the classical encoder-decoder model with multi-level video encoding and attribute prediction. The 2019 edition of the TRECVID benchmark has been a fruitful participation for our joint-team. Our runs are ranked at the second place for AVS and VTT Matching and Ranking tasks and the third place for the VTT Description Generation subtask in terms of the ciderD criterion.*

# 1 Ad-hoc Video Search

## 1.1 Approach

The key to AVS is to compute the semantic relevance between a given natural-language query $s$ and a specific video $v$ from the video collection to be searched. To that end, we aim to learn deep cross-modal embeddings denoted as $f(v)$ and $f(s)$ such that the relevance can be effectively measured by the cosine similarity between the learned embeddings, *i.e.*

$$relevance(s, v) := \frac{< f(s), f(v) >}{||f(s)|| \cdot ||f(v)||}. \tag{1}$$

Accordingly, the AVS task with respect to a specific query is solved by sorting the video collection in descending order by Eq. 1 and returning the top-1,000 ranked results.

Our approach is developed based on two deep learning models, *i.e.* the W2VV++ network [11] and the Dual Encoding Network [7]:

- W2VV++: As a superversion of W2VV [6], the W2VV++ network consists of a sentence encoding subnetwork and a feature transformation subnetwork. Given a natural-language query, the sentence encoding subnetwork uses multi-scale text encoding consisting of of bag-of-words, word2vec, and Gated Recurrent Units (GRU) to encode the query into a real-valued feature vector. The feature transformation subnetwork projects the feature vector into a learned common space for video-query relevance computation. W2VV++ is the winning entry for the TRECVID 2018 AVS task [10].

- Dual Encoding: The dual encoding network employs a multi-level encoding architecture for both modalities. In particular, the multi-level encoding network consists of three encoding blocks that are implemented by mean pooling, bidirectional GRU (biGRU) and biGRU-CNN respectively. The three blocks are stacked to explicitly model global, local and temporal patterns in both videos and sentences. The output of a specific encoding block is not only used as input of a follow-up encoding block, but also re-used via skip connection to contribute to the final output of the entire encoding network. It generates new, higher-level features progressively. Dual Encoding has demonstrated state-of-the-art performance for video-to-text and text-to-video retrieval on the MSR-VTT dataset [7].

Both models, based on fully deep learning, contain three key components, *i.e.* video representation, query representation and common space where cross-modal matching is performed.

**For video representation**, we use deep visual features extracted by pretrained CNNs in a over-sampling manner. Given a video, we uniformly sample frames with an interval of 0.5 second. Each frame is resized to 256 × 256. CNN feature are extracted from its 10 sub images, which are generated by clipping the frame and its horizontal flip with a window of 224 × 224 at their center and their four corners. The ten features are averaged as the frame-level feature. In particular, we adopt a pre-trained ResNet-152 model used in [4] and a pre-trained ResNeXt-101 model used in [14]. Each model extracts a 2,048-dim CNN feature vector from a given frame. By concatenating the two vectors, a 4,096-

dim feature vector is obtained per frame. For W2VV++, a 4096-dim video-level feature is obtained by mean pooling over frames. Dual Encoding uses its multi-scale encoding module, rather than mean pooling, to obtain the video representation.

**For query representation**, Dual Encoding uses its text-side multi-level encoding module to generate a dense representation. As for W2VV++, we additionally include a pre-trained BERT [3] model as another text encoder. That is, in parallel with three sentence encoders, *i.e.* BoW, word2vec and GRU previously used in [11], the BERT encoder is used to encode an input sentence as a 1,024-dim feature vector.

**For common space learning**, we train both W2VV++ and Dual Encoding with the improved marginal ranking loss [8]. Our training data is a joint collection of MSR-VTT [16] and TGIF [12], with hyper-parameters tuned on the training set of the TRECVID 2016 VTT task.

## 1.2 Submissions

We submit the following runs.

- *Run 4* is W2VV++.
- *Run 3* is W2VV++ with the BERT encoder.
- *Run 2* is Dual Encoding.
- *Run 1* equally combines models from all the other runs and trained with different setups.

An overview of the AVS task benchmark is shown in Fig. 1. *Run 4* servers as our baseline. *Run 3*, by adding a BERT encoder, is slightly worse than *Run 4*. Dual Encoding as *Run 2* outperforms the W2VV++ models. The ensemble, *i.e.* *Run 1*, performs the best, and with infAP of 0.160, it is comparable to the best result of this year (infAP of 0.163).

A retrospective experiment of our four runs on the AVS tasks of the previous years is reported in Table 1. There are a number of interesting observations. While adding BERT did not help for the AVS 2019 task, it shows improvement on the AVS 2017 task. How to reliably exploit the state-of-the-art text encoding requires further investigation.

While Dual Encoding (*Run 2*) and W2VV++ perform close in the previous AVS tasks, there is a noticeable performance gap (0.152 *versus* 0.127) in the 2019 task. Interestingly, we found that when re-evaluating *Run 1* with only the Dual Encoding models combined, we obtain the best infAP of 0.170.

**Table 1: Retrospective experiment of this year's runs on the previous AVS tasks**. Dual Encoding* indicates Run 1 but with only Dual Encoding models combined.

| | **TRECVID edition** | | | |
|---|---|---|---|---|
| | *2016* | *2017* | *2018* | *2019* |
| *Previous best run* | 0.054 [9] | 0.206 [14] | 0.121 [10] | 0.163 |
| ***Ours:*** | | | | |
| *Run 4* | 0.163 | 0.196 | 0.115 | 0.127 |
| *Run 3* | 0.161 | 0.217 | 0.115 | 0.124 |
| *Run 2* | 0.165 | 0.228 | 0.117 | 0.152 |
| *Run 1* | **0.169** | 0.235 | 0.129 | 0.160 |
| *Dual Encoding** | 0.162 | **0.239** | **0.132** | **0.170** |

# 2 Video to Text Description

## 2.1 Matching and Ranking

Given a video, participants were asked to rank a list of predefined candidate sentences in terms of their relevance with respect to the given video. In the 2019 edition, the test video set consists of 2,054 videos where 1044 videos are collected from Twitter Vine and 1010 videos are from Flickr. Five sentence sets are provided by the task organizers, denoted as setA, setB, setC, setD and setE. Each sentence set has 2,054 sentences.

### 2.1.1 Approach

Dual Encoding [7] is used. Additionally, we improve the dual encoding network by including the BERT encoder as used in our AVS runs. The BERT feature is concatenated with the global, local and temporal features extracted by multi-level encoding, see Fig. 2.

We train models on a combined set of MSR-VTT [16], MSVD [2] and TGIF [12], with hyper-parameters tuned on the TRECVID 2016 VTT training set.

### 2.1.2 Submissions

We submit the following four runs:

- *Run 1* is the original dual encoding model using the ResNeXt-101 feature.
- *Run 2* is the improved dual encoding model with the BERT features as the extra input. The ResNeXt-101 feature is used.
- *Run 3* equally combines six models, among with three models are based on *Run 1* with their video feature varies. That is, ResNeXt-101 feature, ResNet-152 feature, and the concatenate of ResNeXT-152 and ResNeXt-101 features. The other three models are based on *Run 2*, using the same three video features.
- *Run 4* combines *Run 3* and other two W2VV++ variants. One is original W2VV++ model (*Run 4* in AVS task), the other is W2VV++ with BERT encoder (*Run 3* in AVS task).

Table 3 summaries our results in the TRECVID 2019 VTT matching and ranking subtask. *Run 2*, by including the BERT features, outperforms *Run 2*. *Run 3* combing 6 models gives the best performance.

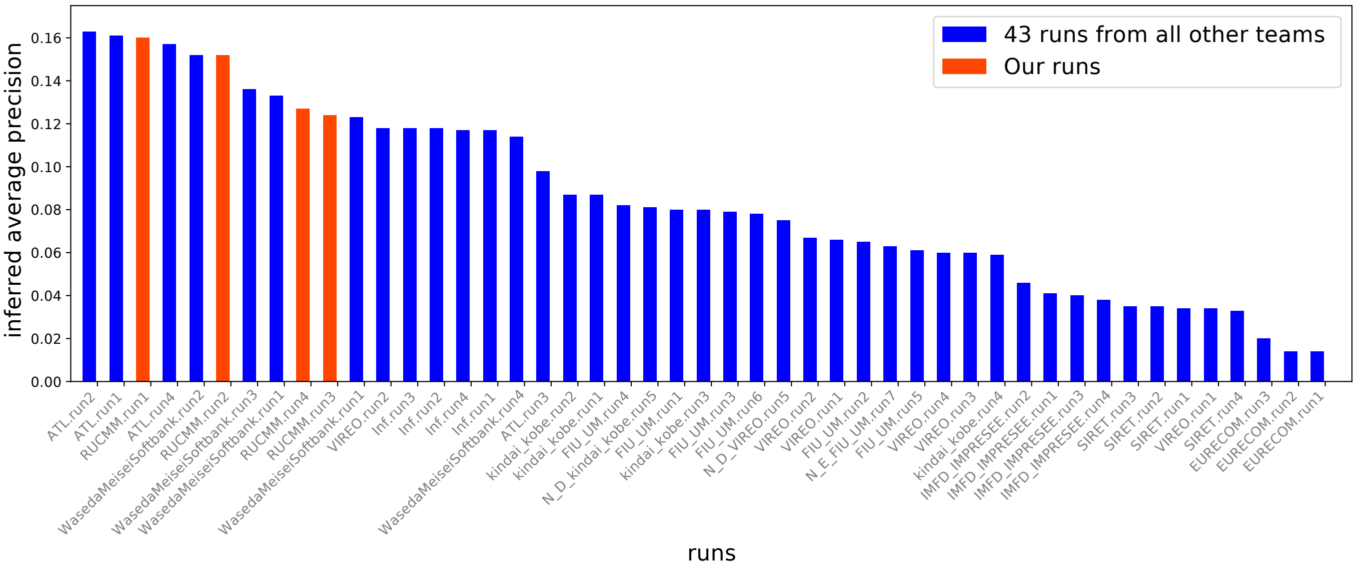


**Figure 1: Overview of the TRECVID 2019 ad-hoc video search task benchmark**, all runs ranked according to mean infAP.

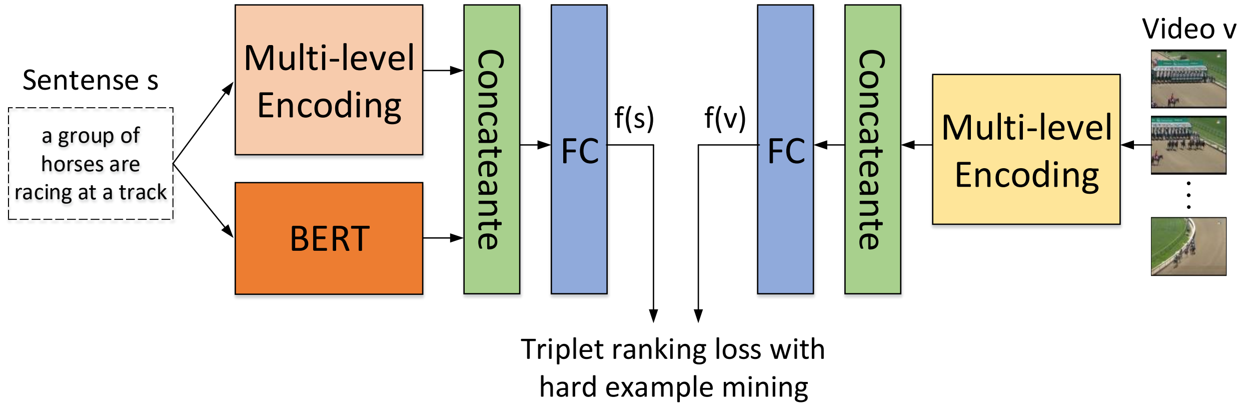


**Figure 2:** Conceptual diagram of our improved dual encoding network for the Video-to-Text Matching and Ranking task.

**Table 2: Our runs in the TRECVID 2019 VTT matching and ranking subtask.**

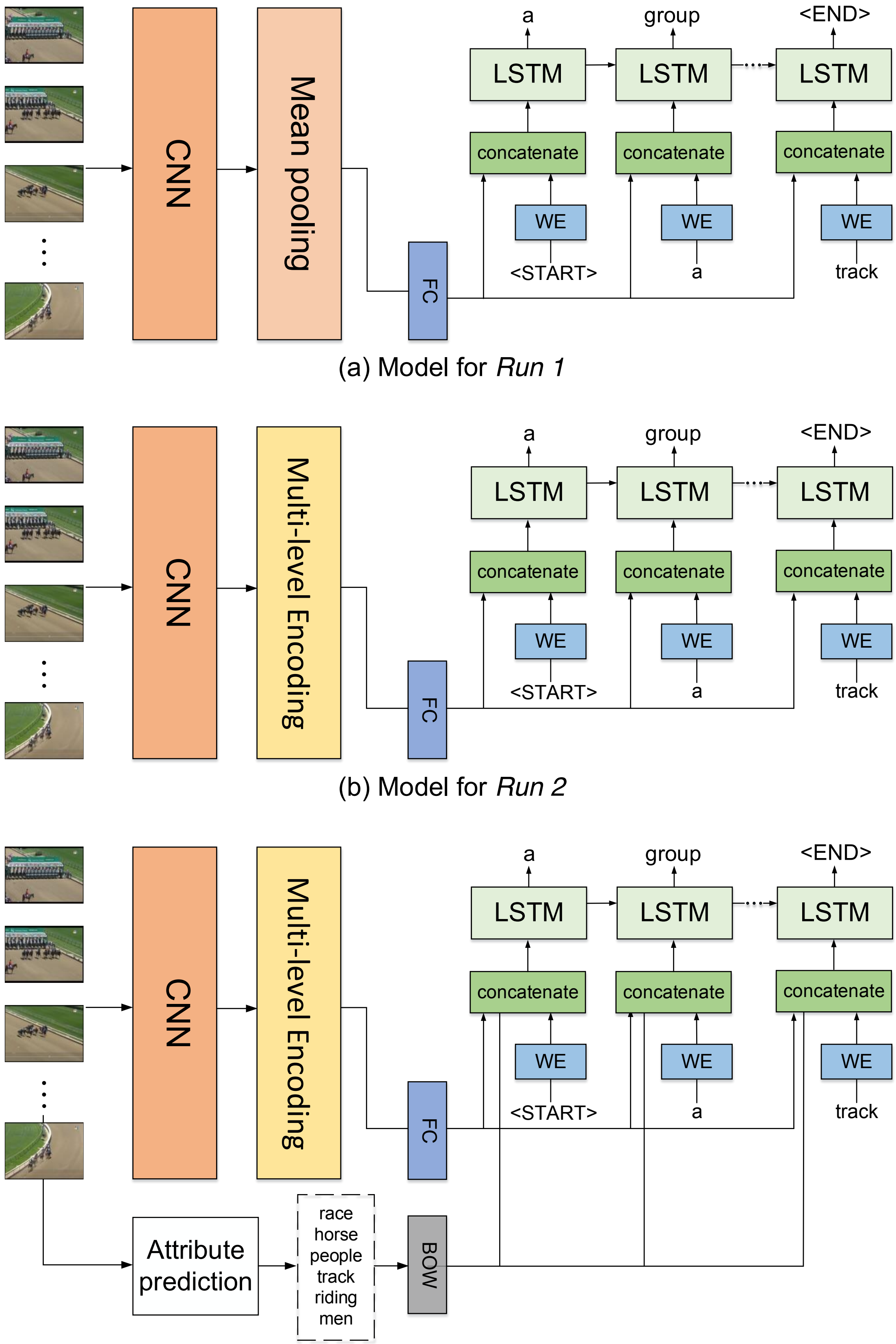
| **Ours** | setA | setB | setC | setD | setE |
|---|---|---|---|---|---|
| *Run 1* | 0.432 | 0.442 | 0.429 | 0.437 | 0.441 |
| *Run 2* | 0.436 | 0.443 | 0.444 | 0.447 | 0.446 |
| *Run 3* | **0.474** | **0.480** | **0.474** | **0.487** | **0.481** |
| *Run 4* | 0.471 | **0.480** | 0.470 | 0.484 | 0.477 |

## 2.2 Description Generation

In this subtask, given a video, participants were asked to automatically generate a natural language sentence to describe the content of the given video and without taking into consideration the existence of any annotated descriptions for this videos.

### 2.2.1 Approach

Our approach is based on the classical encoder-decoder framework [15], where an encoder is used to represent videos, and a decoder is employed to generate sentences word by word. We enhance it by re-employing the dual encoding network [7] in the matching and ranking subtask to better represent videos. The overview of our approach is illustrated in Figure 3(c). Specially, instead of employing common mean pooling on the extracted video frame features [5], we utilize the video-side multi-level encoding branch of the dual encoding network. The multi-level encoding branch is derived from the trained dual encoding network for the above matching and ranking subtask, and we add an extra Fully Connected (FC) layer to obtain the video feature vector. Additionally, inspired by [5, 13], we also enriches the current video representation to the decoder by attribute prediction. We implement the attribute prediction based the dual encoding network to automatically predicts relevant attributes for each video. Concretely, we first



(a) Model for *Run 1*

(b) Model for *Run 2*

(c) Model for *Run 3*

**Figure 3:** Conceptual diagrams of models used in our runs for the description generation subtask.

construct the attribute vocabulary from non-stop words in the training sentences, and the most frequently occurring 512 words are kept. Given a video, we then retrieve top 15 sentences from all the training sentences by dual encoding network used for matching and ranking subtask. The attributes appearing at least once in the retrieved sentences are regarded as the final attribute for the given video. For each video, we represent the predicted attributes by a 512-dim bag-of-words (BoW) vector. Finally, the video feature vector, attribute vector and the word embedding vector are concatenated and fed in the LSTM at each time step. Note that we initialize the LSTM by mean pooling vector of video frame features.

The proposed model is trained on a combined set of MSR-VTT [16], MSVD [2] and TGIF [12], with hyper-parameters tuned on the TRECVID 2016 VTT testing set. The dual encoding network is derived from the matching and ranking subtask, without finetuneing. The test video set consists of 2054 videos from Twitter Vine and Flickr. We use the same ResNext-101, ResNet-512 and C3D features as the previous tasks.

### 2.2.2 Submissions

We submit the following four runs:

- *Run 1* is the baseline model, which uses the mean pooling to represent videos and without attributes as the extra input (Figure 3(a)). ResNext-101 feature is used.
- *Run 2* utilizes the multi-level encoding to represent videos and without attributes as the extra input (Figure 3(b)). ResNext-101 feature is used.
- *Run 3* is our proposed approach, which uses the multi-level encoding to represent videos and utilize attribute feature to enrich the input to the LSTM (Figure3(c)). ResNext-101 feature is used.
- *Run 4* is model ensemble. We combine six models: 1) run2 model trained with ResNet-152, ResNext-101, and C3D features respectively. 2) Run3 model trained with ResNet-152, ResNext-101, and C3D features, respectively.

The performance of our runs on the TRECVID 2019 test set are summarized in Table 3. *Run 2*, by utilizing the multi-level encoding instead of mean pooling, outperforms *Run 1*. The result shows the effectiveness of multi-level encoding for video representation in the context of video captioning. *Run 3* gives better performance than other two single model without attributes as the extra input, showing the attributes are beneficial. *Run 4*, by model ensemble, achieves the best performance.

**Table 3: Our runs in the TRECVID 2019 VTT description generation subtask.**

| **Ours** | Bleu | Metetor | Cider | Cider-D |
|---|---|---|---|---|
| *Run 1* | 0.020 | 0.213 | 0.333 | 0.130 |
| *Run 2* | 0.027 | 0.232 | 0.414 | 0.168 |
| *Run 3* | **0.029** | 0.233 | 0.424 | 0.173 |
| *Run 4* | **0.029** | **0.241** | **0.426** | **0.181** |

## Acknowledgments

The authors are grateful to the TRECVID coordinators for the benchmark organization effort. The Renmin University of China acknowledges support by NSFC (No. 61672523), the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (No. 18XNLG19). The Zhejiang Gongshang University acknowledges support by NSFC (No. U1609215), the Zhejiang Provincial Natural Science Foundation of China (No. Q19F020002).

## References

[1] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, A. F. Smeaton, Y. Graham, W. Kraaij, and G. Quénot. Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval. In *Proceedings of TRECVID 2019*. NIST, USA, 2019.

[2] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[4] J. Dong, S. Huang, D. Xu, and D. Tao. Dl-61-86 at TRECVID 2017: Video-to-text description. In *TRECVID Workshop*, 2017.

[5] J. Dong, X. Li, W. Lan, Y. Huo, and C. G. Snoek. Early embedding and late reranking for video captioning. In *ACM Multimedia*, pages 1082–1086, 2016.

[6] J. Dong, X. Li, and C. G. M. Snoek. Predicting visual features from text for image and video caption retrieval. *T-MM*, 2018.

[7] J. Dong, X. Li, C. Xu, S. Ji, Y. He, G. Yang, and X. Wang. Dual encoding for zero-example video retrieval. In *CVPR*, 2019.

[8] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler. VSE++: Improving visual-semantic embeddings with hard negatives. In *BMVC*, 2018.

[9] D.-D. Le, S. Phan, V.-T. Nguyen, B. Renoust, T. A. Nguyen, V.-N. Hoang, T. D. Ngo, M.-T. Tran, Y. Watanabe, M. Klinkigt, et al. NII-HITACHI-UIT at TRECVID 2016. In *TRECVID Workshop*, 2016.

[10] X. Li, J. Dong, C. Xu, J. Cao, X. Wang, and G. Yang. Renmin University of China and Zhejiang Gongshang University at TRECVID 2018: Deep cross-modal embeddings for video-text retrieval. In *TRECVID Workshop*, 2018.

[11] X. Li, C. Xu, G. Yang, Z. Chen, and J. Dong. W2VV++: Fully deep learning for ad-hoc video search. In *ACM Multimedia*, 2019.

[12] Y. Li, Y. Song, L. Cao, J. Tetreault, L. Goldberg, A. Jaimes, and J. Luo. TGIF: A new dataset and benchmark on animated GIF description. In *CVPR*, 2016.

[13] C. G. Snoek, J. Dong, X. Li, X. Wang, Q. Wei, W. Lan, E. Gavves, N. Hussein, D. C. Koelma, A. W. Smeulders, et al. University of amsterdam and renmin university at trecvid 2016: Searching video, detecting events and describing video. In *TRECVID 2016 Workshop*, 2016.

[14] C. G. M. Snoek, X. Li, C. Xu, and D. C. Koelma. University of Amsterdam and Renmin university at TRECVID 2017: Searching video, detecting events and describing video. In *TRECVID Workshop*, 2017.

[15] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *T-PAMI*, 39(4):652–663, 2016.

[16] J. Xu, T. Mei, T. Yao, and Y. Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.