# TRECVID 2020 AVS: Solution of ZY_BJLAB Team

Kaixu Cui[1,2], Hui Liu[1,2], Chen Wang[1,2], Changliang XU[1,2], Yudong Jiang
[1]XinHuaZhiYun Inc.
[2]State Key Laboratory of Media Convergence Production Technology and Systems
{cuikaixu, liuhui, wangchen, xuchangliang}@shuwen.com
nebuladream@gmail.com(Work at in XinHuaZhiYun Inc.)

## Abstract

*In this report, we describe the datasets and methods we used in TRECVID2020's AVS task. In the training stage, we used the fusion of MSRVTT, MSVD and ActivityNet datasets as the training set in part, and the pre-training model of HowTo100M in the other part. In the inference phase, we use a query ensemble and a penalty ensemble approach to get the final result.*

## 1. Introduction

TRECVID[1] is a video track for the TREC conference series, which is sponsored by the National Institute of Standards and Technology (NIST) with additional support from other U.S. government agencies. TRECVID devoted to research in automatic segmentation, indexing, and content-based retrieval of digital video. Ad-hoc Video Search (AVS) is a sub-task of TRECVID, with a goal to build a video clips search system using text queries. The test collection(V3C1)[2] and a set of Ad-hoc queries are provided in AVS, and the model need return for each query a list of at most 1000 shot IDs from the test collection ranked according to their likelihood of containing the target query.

Since V3C1 only has test set, we need to build our own training dataset. We fused MSRVTT[3], MSVD[4] and ActivityNet[5] as our training dataset. In the task, the methods we used mainly include two types: one is based on the pre-training model of HowTo100M[6], and the other is based on CE[7].

## 2. Datasets

**V3C1**, Vimeo Creative Commons Collection, is 7475 videos with mean video duration of 8 min and total 1,082,659 video segments.

**HowTo100M**, is a large-scale dataset of narrated videos with an emphasis on instructional videos where content creators teach complex tasks with an explicit intention of explaining the visual content on screen. HowTo100M contains 136M video clips with captions sourced from 1.2M Youtube videos and 23k activities from domains such as cooking, hand crafting, personal care, gardening or fitness.

**MSRVTT**, provides 10K web video clips with 41.2 hours and 200K clip-sentence pairs in total, covering the most comprehensive categories and diverse visual content, and representing the largest dataset in terms of sentence and vocabulary.

**MSVD**, consists of 1970 YouTube clips (ranging in length from 10 to 25s), each tagged with about 40 English sentences.

**ActivityNet**, connects videos to a series of temporally annotated sentence descriptions. On average, each of the 20k videos contains 3.65 temporally localized sentences, resulting in a total of 100k sentences.

## 3. Method

### 3.1. The model based on HowTo100M

HowTo100M is a large-scale dataset of 136 million video clips sourced from 1.22M narrated instructional web videos depicting humans performing and describing over 23k different visual tasks. In this method, the video search model pre-trained in Howto100M is used as the basic model, and we modified the text encoding method, replacing the original text encoding method with Bert[8].

### 3.2. The model based on CE

CE is a multi-modal video search model that uses multiple modal information to encode video segments. For example, rgb features, action features, audio features, etc. There is a lot of modal information used in the CE model, but because of the amount of data we only use one RGB feature, one action feature, and one audio feature. Feature extraction use pre-training models for the modes mentioned in CE. At the training stage, we used MSRVTT, MSVD and ActivityNet fusion dataset for training and verification on MSRVTT val dataset.

### 3.3. Query Ensemble

We find that in some queries, the subject order has a tie and a sibling relationship, for example, "boy and gril" or "boy or girl". In the inference phase, we split sentences with

subjects like this, we verify them separately and then ensemble with the result of the original sentence.

### 3.4. Penalty Ensemble

We use a penalty approach[] to ensemble the models produced by our two approaches. First step each model is outputs 1000 sorted lists for each query. The second step is to score the videos in the result list, if a video is ranked at the first position, it gets 0 penalty, a video ranked at the second position gets penalty of 1, and so on. The candidate videos are finally ranked according to the sum of penalty points across the models.

## References


[1] Awad G, et al. "TRECVID 2020: comprehensive campaign for evaluating video retrieval tasks across multiple application domains. " Proceedings of TRECVID, 2020.

[2] Rossetto, Luca, et al. "V3c–a research video collection." International Conference on Multimedia Modeling. Springer, Cham, 2019.

[3] Xu, Jun, et al. "Msr-vtt: A large video description dataset for bridging video and language." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[4] Chen, David, and William B. Dolan. "Collecting highly parallel data for paraphrase evaluation." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011.

[5] Krishna, Ranjay, et al. "Dense-captioning events in videos." Proceedings of the IEEE international conference on computer vision. 2017.

[6] Miech, Antoine, et al. "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips." Proceedings of the IEEE international conference on computer vision. 2019.

[7] Liu, Yang, et al. "Use What You Have: Video Retrieval Using Representations From Collaborative Experts." (2019).

[8] a Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).