Kuaishou at TRECVID 2021: Two-stage Ranking Strategy for Ad-hoc Video Search

Fangming Zhou[1*], Yihui Shi[3], Changqiao Wu[2], Xiaofeng Guo[2], Haofan Wang[2], Jincan Deng[2], Debing Zhang[2]
[1]Renmin University of China
[2]MMU, Kuaishou Technology
[3]Beijing University of Posts and Telecommunications

Abstract

We propose a two-stage ranking strategy for the TV21 AVS task. In general, we use the common-used video retrieval model [7, 11] to calculate the cross-modal similarity firstly and then utilize an image-text matching model to re-rank the previous retrieval result in the frame level. Specifically, in the first stage, an advanced SEA++ model inherited from Sentence Encoder Assembly (SEA) [11] is proposed. We improve the SEA in the video end, as SEA++ provides a specific common space for each combination of sentence encoder and video encoder. That is to say, for m video encoders and n sentence encoder, a total of $m \times n$ common spaces are built to calculate the final cross-modal similarity. We consider MSR-VTT [17], TGIF [12], and VATEX [16] as our training data. In the second stage, we only consider the top k videos in descending order by their similarity generated in the previous stage. Then we calculate the similarities of query and all frames extracted from the k videos, employing an image-text matching model [15]. Each video is represented by its frame with the highest similarity. After that, the reordering results of k videos are obtained. The single SEA++ model achieves an infAP of 0.332 on the TV21 task. Late fusion of models, trained by different configurations, gains a higher infAP of 0.340. Our best run, which scored infAP of 0.349, is obtained by re-ranking strategy based on the previous result, ranked second among all submitted runs.

1 Our Approach

In the recent years, most of the methods adopted by top performers, in Ad-hoc Video Task(AVS), follow this general framework: encoding video, encoding text and then project them into a comparable common space for metric learning and similarity ranking of retrieval results.

Because of the uninterpretability of the deep learning method and the strong dependence on the training data, it is difficult for us to get a good retrieval result through the model trained by a single training strategy. The previous work can significantly improve the retrieval result by fusing the models obtained by different training strategies. Similar to model fusion, this paper proposes a two-stage ranking strategy. We use 1) an advanced video retrieval model, SEA++ and 2) a fine-grained image text matching model, contributes to the final retrieval results.

1.1 First Stage

In the first stage, we mainly use the SEA++ model, improved by SEA [11]. It is featured by its simple-design and effectiveness.

Like SEA [11], SEA++ can can easily improve the complementarity between different text encoders and video features, but with higher flexibility and accuracy.

1.1.1 The SEA++ Model

Previous SEA model sets individual common space for different text encoders to match with video feature, which is better than the traditional single space matching model [7, 9]. However, it only considers the diversity and complementarity between text encoders and ignores the impact of video features on the results [3, 18] As illustrated in Fig. 1, we provide individual common space for each combination of each text feature and each video feature.

Specifically, we denote text query as q and unlabeled video as v, for n video feature $\{ f_1(v), f_2(v), \ldots, f_n(v) \}$, and m text feature $\{ e_1(q), e_2(q), \ldots, e_m(q)\}$, a total of $n \times m$ common spaces are built.

By averaging the similarities of $ntimesm$ common spaces, we have the overall similaryty in the first stage as:

$$S_{first}(q, v) = \frac{1}{n \cdot m} \sum_{i=1}^{m} \sum_{j=1}^{n} cosine(FC_{t,i}(e_i(q)), FC_{v,j}(f(v))) \quad (1)$$

, where $FC_{t,i}$ and $FC_{v,j}$ indicate the two FC layers

For the other training strategy, we just follow the setting of SEA [11], like the dimension of fully-connected projection layer, the choice of loss function, and so on.

*This work was done during the first author's intership in MMU, Kuaishou Technology

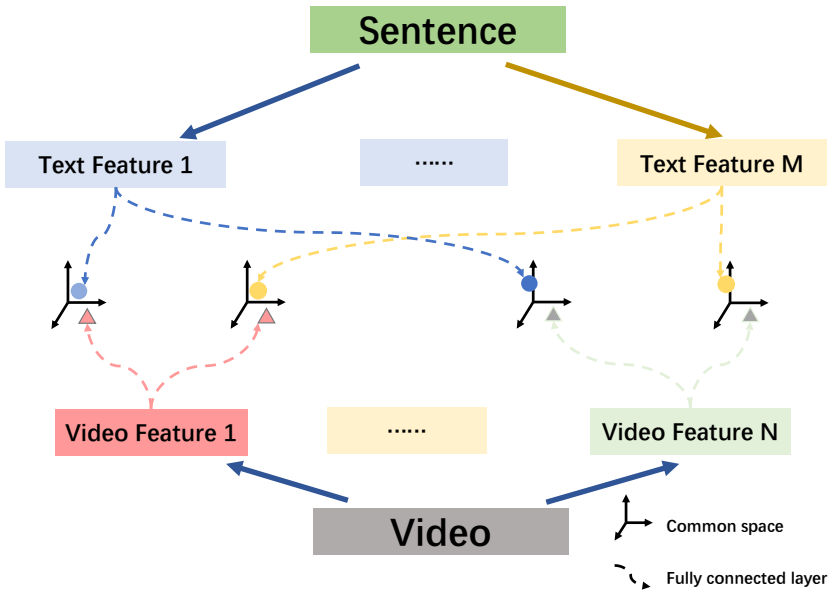

Figure 1: The overall architecture of the proposed SEA++ model. We provide individual common space for each combination of each text feature and each video feature.

1.1.2 Choice of Text Encoders

Referring to the experimental results in sea [10, 11], we choose bag-of-word(BoW) [6] and w2v [14] as our text encoders. We believe that choosing these two concise text encoders instead of other complex text modeling method [4, 5] at this stage can simplify the matching difficulty of the model. That is to say, we are performing a keyword-video matching task in this phase, by ignoring the stop words in text query. Through the blessing of video features, an effective preliminary ranking result is obtained. We leave the task of modeling text temporal semantics to the model in the second stage.

1.1.3 Choice of Video Features

We selected the following typical models in computer vision or cross-modal tasks as our visual feature extractors.

- CLIP [15]
- irCSN [8]
- timesformer [2]
- ResNeXt101 [13]

1.1.4 Choice of Training Data

Following previous works [10, 18], we use the joint set of MSR-VTT [17], TGIF [12] and VATEX [16] as training data and TV2016-vtt-dev [1] as the validation set.

1.2 Second Stage

At the second stage, we directly use the CLIP [15] released by OpenAI as the text-image matching model. CLIP has shown its great "zero-shot" capabilities in many tasks, such as image classification, text to image generation, and our experiments proved that CLIP can also boost AVS performance by post-processing of re-ranking.

For a given video v, we uniformly sample frames with an interval of 0.5 second. After that, the frame set $\{frame_1, \ldots, frame_t,\}$ of video v is obtained, t indicates the number of frames. The similarity of v and q in this stage is defined as:

$$S_{second}(q, v) = \max\left(S_{clip,1}, \ldots, S_{clip,t}\right) \tag{2}$$

$$S_{clip,i} = cosine(CLIP(frame_i), CLIP(q)) \tag{3}$$

, where the $frame_i$ indicates the ith frame of v.

The final similarity of q and v are built by weighted summing of the two similarities:

$$S_(q, v) = w \cdot S_{first}(q, v) + (1 - w) \cdot S_{second}(q, v) \tag{4}$$

, we set the $w = 0.2$ in our experiments.

2 Submissions

As shown in Table 1, we submitted runs 3, corresponding to our best single SEA++ model, which scores infAP of 0.332. By leverage different SEA++/SEA models trained by diverse training configurations, we obtain run 2 with an higher infAP of 0.340. Our best runs, both run1 and run4, is generated by re-ranking strategy based on the late-fusion result, achieves infAP of 0.349 and are ranked second among all submitted runs.

Table 1: Performance of our four runs on the TRECVID 2019–2021 AVS tasks.

	2019	*2020*	*2021*
Our TV21 submissions:			
Run 3 (single SEA++ model)	0.206	0.354	0.332
Run 2 (Late fusion)	0.211	**0.362**	0.340
Run 4 (Late fusion and re-ranking)	**0.241**	0.360	**0.349**
Run 1 (Late fusion and re-ranking)	0.239	0.358	**0.349**

3 Conclusions

In this paper, we summarize the details of our team's solution in TRECVID 2021 AVS task. Our contributions are three-fold: 1) We propose an improved video retrieval model, namely SEA++, which built a solid backbone for our best run. 2) The traditional model fusion strategy is still effective for AVS task. 3) Re-ranking by CLIP is an effective method to gain a higher performance.

References

[1] G. Awad, J. Fiscus, D. Joy, M. Michel, A. Smeaton, W. Kraaij, G. Quénot, M. Eskevich, R. Aly, R. Ordelman, G. Jones, B. Huet, and M. Larson. Trecvid 2016: Evaluating video search, video event detection, localization, and hyperlinking. In *TRECVID Workshop*, 2016.

[2] G. Bertasius, H. Wang, and L. Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021.

[3] A. Chen, F. Hu, Z. Wang, F. Zhou, and X. Li. What matters for ad-hoc video search? a large-scale evaluation on trecvid. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[4] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[6] J. Dong, X. Li, and C. G. M. Snoek. Predicting visual features from text for image and video caption retrieval. *IEEE Transactions on Multimedia*, 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] D. Ghadiyaram, D. Tran, and D. Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*.

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

[10] X. Li, F. Zhou, and A. Chen. Renmin university of china at trecvid 2020: Sentence encoder assembly for ad-hoc video search.

[11] X. Li, F. Zhou, C. Xu, J. Ji, and G. Yang. SEA: Sentence encoder assembly for video retrieval by textual queries. *IEEE Transactions on Multimedia*, 2021.

[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] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *ICLR*, 2013.

[15] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

[16] X. Wang, J. Wu, J. Chen, L. Li, Y.-F. Wang, and W. Y. Wang. VaTeX: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019.

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

[18] Y. Zhao, Y. Song, S. Chen, and Q. Jin. Ruc_aim3 at trecvid 2020: Ad-hoc video search & video to text description. TRECVID, 2020.

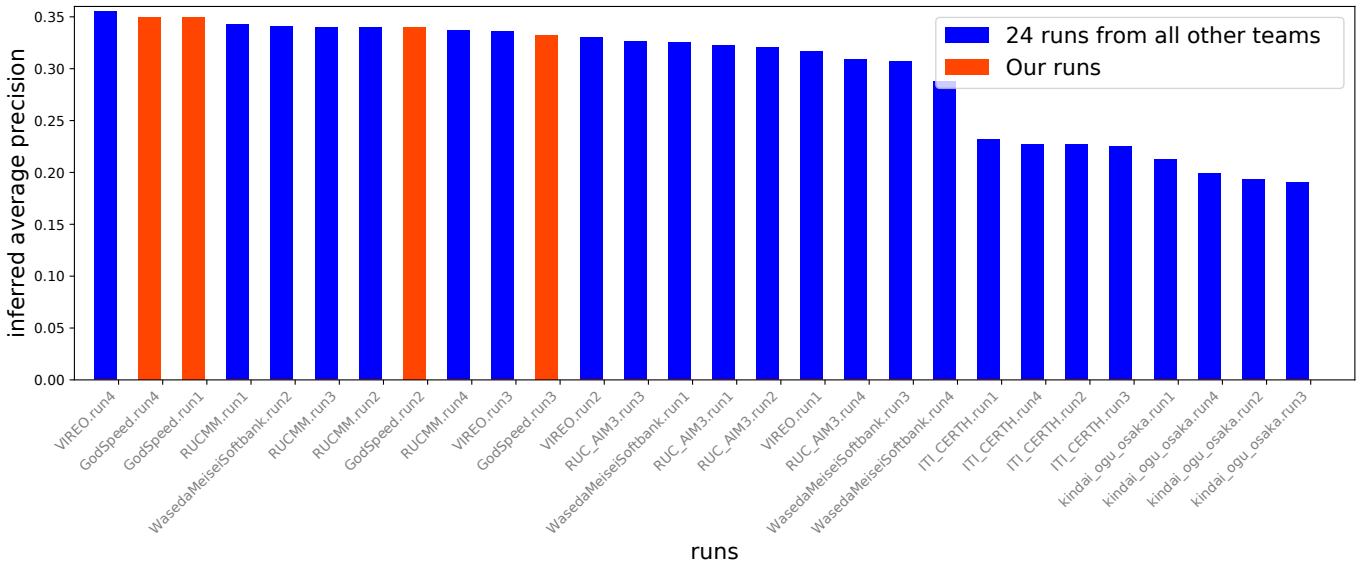

Figure 2: Overview of the TRECVID 2021 AVS benchmark evaluation. The red indicates the runs submitted by our team.