Reinforcement Learning-Based Interactive Video Search

Zhixin Ma[1], Jiaxin Wu[2], Zhijian Hou[2], and Chong-Wah Ngo[1]

[1] School of Computing and Information Systems, Singapore Management University
Singapore
zxma.2020@phdcs.smu.edu.sg, cwngo@smu.edu.sg
[2] Department of Computer Science, City University of Hong Kong
Hong Kong, China
{jiaxin.wu, zjhou3-c}@my.cityu.edu.hk

Abstract. Despite the rapid progress in text-to-video search due to the advancement of cross-modal representation learning, the existing techniques still fall short in helping users to rapidly identify the search targets. Particularly, in the situation that a system suggests a long list of similar candidates, the user needs to painstakingly inspect every search result. The experience is frustrated with repeated watching of similar clips, and more frustratingly, the search targets may be overlooked due to mental tiredness. This paper explores reinforcement learning-based (RL) searching to relieve the user from the burden of brute force inspection. Specifically, the system maintains a graph connecting shots based on their temporal and semantic relationship. Using the navigation paths outlined by the graph, an RL agent learns to seek a path that maximizes the reward based on the continuous user feedback. In each round of interaction, the system will recommend one most likely video candidate for users to inspect. In addition to RL, two incremental changes are introduced to improve VIREO search engine. First, the dual-task cross-modal representation learning has been revised to index phrases and model user query and unlikelihood relationship more effectively. Second, two more deep features extracted from SlowFast and Swin-Transformer, respectively, are involved in dual-task model training. Substantial improvement is noticed for the automatic Ad-hoc search (AVS) task on the V3C1 dataset.

Keywords: Reinforcement Learning · Query Understanding · Feature Enhancement · Interactive Video Retrieval

1 Introduction

Concept-based [11,8] and concept-free [2,13] search paradigms have enabled text-to-video search with either few keywords or a short sentence as query. Despite tremendous progress in this topic since the advancement in cross-modal representation learning, there are still various factors hindering effective video search. These factors can be attributed to system limitations such as the inability to model out-of-vocabulary query terms and user search intention, due to user because of ambiguous query, or dataset bias for having several large clusters of shots with similar background scenes. In general, user-machine interaction is required to pave a way to resolve ambiguity or misinterpretation. Nevertheless, the current VIREO systems [14,8] fall short in dealing with any of these factors effectively. Specifically, the interaction is one-way, where the user interactively refines the query in a trial-and-error manner, while the system only responses to the current refined query by ignoring the search history and video browsing pattern.

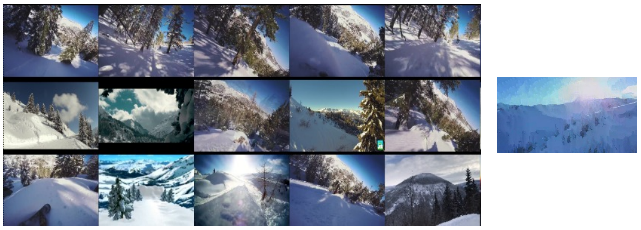

Fig. 1. The retrieval result for a known-item visual query using the query terms: snow mountain, tree, sun. The shot on the right shows the search target. User has to manually inspect the search list to locate the search target.

In VBS 2021 [9], there are several snow-relevant queries, e.g., "There are snow covered mountains outside the hut.". Due to the limited query terms that can be expressed for these queries, most returned shots are snow landscapes. Figure 1 shows the result retrieved by VIREO search engine [14] for a visual known-item query. The query searches for a snow mountain scattered with trees on a sunny day. Due to the lack of distinctive terms to textually describe the visual scene, the returned shots are similar to each other with snow mountains and sunlight. As a consequence, a user has to exhaustively browse through the rank list, which is tiring and time-consuming especially when the ground-truth shot is ranked low in the list. Hierarchical clustering of shots for browsing could be a promising solution, which has been extensively explored in various VBS systems [1,10]. However, in the case when the shots are highly similar in terms of semantics and color-texture statistics, clustering may not be able to properly capture the subtle changes among shots. Relevance feedback, such as adopted by [12], is effective in distinguishing positive and negative examples but cannot adequately recommend satisfactory results when "the devil is in the details". This is not mentioning the fact that known-item-search involves only one true positive and relevance feedback is not always directly applicable to this task.

This paper aims to address the challenge as illustrated in Figure 1 as a sequential search problem with AI planning. Inspired by [5], we extend the idea of lookahead inference from single video to a video collection. Specifically, in [5], an RL agent learns to locate the moment of interest in a video by interactively moving to the left or right of the current clip under navigation. Similarly but more broadly, our idea is to explore RL to provide a navigation path within and across videos such that user can identify the search target as rapidly as possible. The path is planned in advanced through learning and dynamically updated during search depending on user feedback. Given a rank list, user starts by clicking a shot for browsing. In a normal situation, a user will have few reactions upon watching a shot: video fast forward or backward browsing, retreat by picking another shot for browsing, or restart the search by refining query terms. The current VIREO search engine is considered passive because no learning is conducted to understand user browsing patterns and subsequent queries are treated independently. Reinforcement learning provides a mechanism to address both issues. First, a graph is constructed to connect the candidate shots with edges modeling their transition probabilities. With this graph, the user navigation path is traced with a long-term reward indicating the deviation from an optimal path. Second, a subsequent query is regarded as feedback of the current recommendation and is leveraged to adjust the navigation path. In other words, the search engine provides recommendations by actively tracking, memorizing, and learning-from-mistakes. The core engine of VIREO is based on the dual-task neural model that embeds cross-modal features in an interpretable latent space [13]. Several incremental changes are made to improve the core engine, aiming to reduce the sensitivity of query expression and enhance the underlying video representation. For example, the queries "A protest camp on a public square" and "Protest tents on a public square" express the same information need, but the return results can vary largely. In the current situation, user has to attempt multiple queries in the trial-and-error manner until reaching a satisfactory result. The improvement made is by decoding the embedded query into a list of canonical query terms for search. More concretely, two queries expressing the same information need are expected to be decoded with similar canonical terms such that their results do not vary arbitrarily. Other improvements made include training of dual-task model with a new unlikelihood loss function such that exclusive concepts (e.g., indoor versus outdoor) will not be decoded simultaneously to describe shot or query content.

2 Reinforcement Learning based Interaction

The interaction between user and system is modelled as a graph traversing problem, where the graph provides video navigation paths for model-based reinforcement learning with Markov Decision Process. Precisely, an agent predicts the next navigation by either moving to the left or right of the current video or jumps to a shot of a new video, depending on the user feedback. To lay a skeleton for path navigation, we build a graph connecting all the shots in a video corpus, e.g., TVR and DiDeMo. For each shot, edges are established to link the preceding and succeeding shots, and the shots of other videos whose similarity is larger than an empirical threshold. The similarity is based on the average cosine similarity of both the semantic concepts and visual embeddings. The edge encodes the semantic difference between two shots. Specifically, we take the top-50 concepts in the shots and embed the difference in concepts as the edge representation. The edges principally provide the information deviated from a shot. During the interactive search, a user can provide feedback such as "this is not a red rock mountain" or "the man should hold microphone" for the shot under navigation. The user feedback will be encoded together with the clip currently under browsing as well as the history of query as the representation for a new state. The agent then takes action a by selecting an edge from the current shot that best captures the feedback based on policy π. The policy network π_θ is implemented as a multi-layer perceptron. The corresponding shot will then be prompted to users for browsing.

The agent network is trained by simulation, similar to the idea of relative captioning [4]. Specifically, given a query, a list of candidate shots and the target shot, the network simulates the move at each step by receiving feedback. The simulated feedback is a concept randomly picked from the current or target shot. The concept is either an unwanted concept not present in the target or a missing concept not in the current shot. With this information, the agent receives a reward indicating whether a chosen path is optimal. We define the path optimality based on the shortest distance from the current shot to the target shot. If the selected edge is along the shortest path, the agent will receive a positive reward. We sample the starting nodes and trajectories (i.e., the shortest path) $\tau = (s_1, a_1, s_2, a_2, ...)$ from the graph. Each trajectory is divided into i.i.d. state-action pairs: $\{(s_i, a_i)\}$. We learn the policy using supervised learning by minimizing the loss function $L(a, \pi_\theta(s))$, which is the cross-entropy loss. Given the current policy and the simulated feedback, the value of taking an action can be computed by look-ahead policy based on the actor-critic algorithm [7]. During online search, the system needs extra computation time to encode query and video browsing history. Nevertheless, the time incurred is negligible for involving only encoding of the current feedback and video under browsing with the history observed so far. Overall, the interactive system still runs in real-time. Additional memory space is required to store the graph whose size is controllable depending on the number of edges that is allowed for a shot.

3 Query Understanding

The VIREO search engine is based on the dual-task model [13] to provide the initial search result. The model has only one decoder that interprets the semantic concepts underlying a video embedding. During retrieval, the query terms are matched directly with the decoded concepts for cosine similarity measure. As a consequence, the retrieval result changes depending on the query terms as well as how a query is phrased. During search, user often needs to interactively refine the query until reaching a satisfactory result before browsing. The refinement usually involves addition of new concepts, removal of the existing concepts upon seeing the search results, and rephrasing the original query. To relieve the user from these trial-and-error attempts, the dual-task model is revised to have two decoders. The second decoder interprets the query embedding with relevant semantic concepts. The decoded concepts are usually richer than the original set of query terms, covering subtle keywords overlooked by the user. For example, the subtle concepts decoded for the query “two or more men at a beach scene” are: sand, shore, ocean. These concepts are indirectly relevant and not likely to be included as query terms especially by novice users. Overall, the modification makes the search result relatively insensitive to query expression and is able to improve the search performance of automatic AVS. We empirically compare this method, which we call “Expand”, to direct matching [13] and word2vec which consider only the original query terms. As shown in Table 1, the new methods

Table 1. Comparing different query processing schemes for concept-based AVS search.

	IACC.3 dataset			V3C1 dataset	
Concept selection	tv16	tv17	tv18	tv19	tv20
Word2Vec	0.143	0.137	0.092	0.098	0.206
Direct [13]	0.152	0.132	0.090	0.096	0.173
Expand	0.183	0.243	0.142	0.139	0.260

Table 2. Incremental contributions of each feature enhancement for AVS search.

	Appearance feature		Motion feature	V3C1 dataset	
	ResNet, ResNeXt	Swin-Tranformer	SlowFast	tv19	tv20
L1	✓			0.184	0.262
L2		✓		0.172	0.258
L3	✓	✓		0.203	0.278
L4	✓		✓	0.185	0.306
L5		✓	✓	0.182	0.293
L6	✓	✓	✓	**0.207**	**0.325**

significantly outperform the other methods across five different query sets. Despite using only concepts for AVS, the result has indeed surpassed the results of concept-free search [2].

Two other incremental improvements made includes enriching the current concept vocabulary with phrases and use of unlikelihood loss function to retrain the dual-task model. The dual-task model [13] considers only single words for training. We retrain the model using the phrases parsed from the video captions of the training datasets. In addition, the video decoder in the dual-task model treats each term independently. As the result, the potential "conflict" such as generating outdoor and indoor concepts, night and day concepts simultaneously is not resolved. We modify the unlikelihood loss function to explicit penalize the decoder when conflicting concepts are being fired by the decoder.

VIREO search engine [14] considers only appearance features (ResNet and ResNeXt) for video representation learning. The current system cannot effectively handle action-oriented query such as people clapping. We enhance the features with Swin-Transformer [6] and SlowFast [3], which are the state-of-the-art image and video classifiers, respectively. Table 2 shows the improvement introduced by different features. L1 shows the baseline performance with appearance features only. After appending Swin-Transformer feature, L3 improves 32 out of 50 queries. The queries "one or more picnic tables outdoors" and "a person wearing a backpack" gain the largest increase in average precision. When further appending slowfast feature, L6 improves 28 out of 50 queries. The queries which contain verbs, like "group of people clapping" and "one or more persons exercising in a gym", benefit from additional motion information.

4 Conclusion

We have presented one new feature, based on reinforcement learning, for VIREO search engine to particularly address the issue of "mental tiredness" in interactive search. Specifically, the engine recommends navigation path for user to browse, while dynamically adjusting the path based on feedbacks provided by user. Together with query understanding to smooth out result sensitivity, this two-way system-user interaction is expected to reduce unnecessary trial-and-error querying and exhaustive browsing from the user side. A user study will be conducted in the near future to investigate the degree in which the new feature can cut short the interactive search time and improve user experience. We have also introduced system improvements such as using unlikelihood loss function and state-of-the-art deep features for enhancing the performance of dual-task model. Although the effect of these changes is yet to be studied in VBS 2022, noticeable improvement has been obtained for TRECVid AVS automatic search.

Acknowledgment

This research was supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant.

References

1. Barthel, K.U., Hezel, N., Mackowiak, R.: Navigating a graph of scenes for exploring large video collections. In: International Conference on Multimedia Modeling (2016)
2. Dong, J., Li, X., Xu, C., Ji, S., He, Y., Yang, G., Wang, X.: Dual encoding for zero-example video retrieval. In: CVPR (2019)
3. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proceedings of the IEEE/CVF ICCV
4. Guo, X., Rennie, S., Wu, H., Tesauro, G., Cheng, Y., Feris, R.S.: Dialog-based Interactive Image Retrieval. Advances in Neural Information Processing Systems (2018)
5. He, D., Zhao, X., Huang, J., Li, F., Liu, X., Wen, S.: Read, watch, and move: Reinforcement learning for temporally grounding natural language descriptions in videos. In: AAAI (2019)
6. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)
7. Mnih, V., Badia, A.P., Mirza, L., Graves, A., Harley, T., Lillicrap, T.P., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: 33rd International Conference on Machine Learning, ICML 2016 (2016)
8. Nguyen, P.A., Ngo, C.W.: Interactive Search vs. Automatic Search: An Extensive Study on Video Retrieval. ACM Transactions on Multimedia Computing, Communications and Applications **17**(2) (2021). https://doi.org/10.1145/3429457
9. Schoeffmann, K., Lokoč, J., Bailer, W.: 10 years of video browser showdown. In: Proceedings of the 2nd ACM International Conference on Multimedia in Asia (2021)
10. Schoeffmann, K., Taschwer, M., Boeszoermenyi, L.: The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In: Proceedings of the first annual ACM SIGMM on Multimedia systems (2010)
11. Ueki, K., Hori, T., Kobayashi, T.: Waseda_meisei_softbank at trecvid 2019: Ad-hoc video search. In: TRECVID (2019)
12. Veselý, P., Mejzlík, F., Lokoč, J.: Somhunter V2 at video browser showdown 2021. In: International Conference on Multimedia Modeling (2021)
13. Wu, J., Ngo, C.W.: Interpretable embedding for ad-hoc video search. In: Proceedings of the 28th ACM International Conference on Multimedia (2020)
14. Wu, J., Nguyen, P.A., Ma, Z., Ngo, C.W.: Sql-like interpretable interactive video search. In: International Conference on Multimedia Modeling (2021)