SQL-Like Interpretable Interactive Video Search

Jiaxin Wu[1(✉)], Phuong Anh Nguyen[1], Zhixin Ma[2], and Chong-Wah Ngo[2]

[1] Department of Computer Science, City University of Hong Kong, Hong Kong, China
{jiaxin.wu,panguyen2-c}@my.cityu.edu.hk
[2] School of Computing and Information Systems, Singapore Management University, Singapore, Singapore
zxma.2020@phdcs.smu.edu.sg, cwngo@smu.edu.sg

Abstract. Concept-free search, which embeds text and video signals in a joint space for retrieval, appears to be a new state-of-the-art. However, this new search paradigm suffers from two limitations. First, the search result is unpredictable and not interpretable. Second, the embedded features are in high-dimensional space hindering real-time indexing and search. In this paper, we present a new implementation of the Vireo video search system (Vireo-VSS), which employs a dual-task model to index each video segment with an embedding feature in a low dimension and a concept list for retrieval. The concept list serves as a reference to interpret its associated embedded feature. With these changes, a SQL-like querying interface is designed such that a user can specify the search content (subject, predicate, object) and constraint (logical condition) in a semi-structured way. The system will decompose the SQL-like query into multiple sub-queries depending on the constraint being specified. Each sub-query is translated into an embedding feature and a concept list for video retrieval. The search result is compiled by union or pruning of the search lists from multiple sub-queries. The SQL-like interface is also extended for temporal querying, by providing multiple SQL templates for users to specify the temporal evolution of a query.

Keywords: SQL-like interpretable search · Concept-free search · Concept-based search · Interactive video search · Video browser showdown

1 Introduction

Video Browser Showdown (VBS) is a live interactive video search held in every year [4,5,9]. It is a well-known benchmark to evaluate the video search system in the literature. This benchmarking activities include three tasks: visual known-item search, textual known-item search, and ad-hoc video search. Visual known-item search provides a visual content of the target video as the query,

The original version of this chapter was revised: the acknowledgement section has been corrected. Additionally, the affiliations of the third and last author and the e-mail address of the last author have been corrected. The correction to this chapter is available at https://doi.org/10.1007/978-3-030-67835-7_51

J. Lokoč et al. (Eds.): MMM 2021, LNCS 12573, pp. 391–397, 2021.
https://doi.org/10.1007/978-3-030-67835-7_34

and the participants need to find the corresponding video clip from a large video collection within a short period of time. The textual known-item task textually describes the audio-visual content of a video as the query. In contrast, ad-hoc video search (AVS) does not assume the knowledge of a target video. The task is to search as many video clips as possible that meet the query description in text.

One of the key features in solving these tasks is to perform the cross-modal search by understanding of video semantics and user search attention [9]. With a user' input query, the system should manage to effectively and efficiently find highly relevant video clips at the top of the ranked list. Most of the previous participants in VBS applied concept-based methods as their text-to-video search models [6]. Concept-based methods rely on concept classifiers to index a bunch of concepts in the videos, and in the real scenario, users could retrieve videos by these concepts. Due to the success of deep learning, the accuracy of concept detection has improved tremendously, boosting the performance of interactive search [9]. Recently, concept-free methods which embed video and text in a joint space has shown their supreme performances in the text-to-video retrieval, and become the new state-of-the-art in AVS task [3,10].

However, as concept-free methods perform matching in a black-box manner, the result is not interpretable. For instance, a user might be frustrated for requiring to attempt different ways of expressing a text query in order to obtain a satisfying result. Furthermore, the current concept-free models [3,10] embed features in high-dimensional space. The high demand in memory consumption hinders real-time indexing and search.

To solve the aforementioned shortcomings, in this paper, we introduce a new version of our Vireo video search system (Vireo-VSS), which incorporates a dual-task model [10] for the text-to-video search. The dual-task model trains the concept-free method and concept-based method in an end-to-end deep network. The concept-based method decodes the embedding feature of the concept-free method into a list of concepts for interpretation. In the implementation, we perform dimensionality reduction of the model to allow it suitable for real-time application. Besides, a SQL-like querying interface is designed for users to formulate the query in an explicit way. Instead of providing one text box for inputting the whole query, we allow several kinds of text boxes for querying, e.g., subject, predicate, object text boxes. The motivation is to relieve the user from the trial-and-error way of querying, and instead to focus on expressing the object-of-interest and their relationship in a fill-in-the-blank manner. The interface also allows users to specify time, location, and logical constraints with ease, instead of formulating these constraints into a long sentence. According to what the users put in this interface, the system will generate one or multiple sub-queries, and input them to the dual-task model for search. The following sections describe the Vireo-VSS in details.

2 Dual-Task Model for Real-Time Interactive Search

Our Vireo-VSS employs a dual-task model [10] for cross-modal search. The dual-task model is comprised of two tasks that are learnt end-to-end with neural

Table 1. Performance comparison (mean xinfAP) when reducing the dimensionality of embedding features from 2,048 to 256 on TRECVid AVS datasets.

Datasets	IACC.3			V3C1	Mean
Query sets	tv16	tv17	tv18	tv19	
2,048-dimensional models					
Concept search	0.148	0.147	0.091	0.115	0.125
Embedding search	0.163	0.232	0.118	0.160	0.168
Fusion search	0.185	0.241	0.123	0.185	0.184
256-dimensional models					
Concept search	0.140	0.144	0.087	0.111	0.121
Embedding search	0.146	0.229	0.121	0.160	0.164
Fusion search	0.166	0.243	0.126	0.174	0.177

networks. The first task is the textual-visual embedding matching which aims to minimize the distances between matched video-text pairs in a joint space. The other task is the multi-label concept classification to recover semantic concepts from the visual embedding. Two tasks are trained simultaneously to ensure the semantic consistency such that the embedding feature can properly reflect the video content when being decoded.

The dual-task model provides three schemes for search: embedding search, concept search and fusion search. Embedding search is based on the visual-textual embedding matching task. By inputing a textual query q, the model measures the similarity between the embeddings of the query $\tau(q)$ and a video $\phi(v_i)$. A score is computed for each video based on their cosine similarity:

$$score_{embedding}(q, v_i) = sim(\tau(q), \phi(v_i)). \tag{1}$$

Concept search is based on the trained model in the multi-label concept classification task. In the testing stage, the trained dual-task model provides each test video v_i a predicted concept vector $\hat{y}(v_i) \in \mathbb{R}^{n+}$. The dimension n is equal to the number of concepts in the concept bank, and each value of $\hat{y}(v_i)$ gives the predicted probability of a concept appearing in the video v_i. Given the user' s input query q, a vector $c_q \in \{0,1\}^n$ will be formed composing of concepts extracted from q. Then, a concept score will be computed:

$$score_{concept}(q, v_i) = sim(c_q, \hat{y}(v_i)). \tag{2}$$

The fusion search uses a linear function to combine the embedding and concept scores as:

$$score_{combined}(q, v_i) = \theta * score_{concept}(q, v_i) + (1 - \theta) * score_{embedding}(q, v_i). \tag{3}$$

The fusion weight $\theta \in [0, 1]$ can be defined by the user in the interactive search. A value of 0 or 1 corresponds to the pure concept-based or pure concept-free search respectively.

However, the high-dimensional embedding features in the original dual-task model [10] hinders its usage in real-time index and search. Thus, we develop a dual-task model in low dimensional space in this paper for interactive search. We change the output dimension of the embedding feature from 2,048 to 256. As a result, the memory consumption is significantly reduced from about 16 GB to 2 GB, and the search speed is improved to 0.3 seconds per query on a standard PC. The empirical results also verify that the dimension reduction only slightly degrades the performance. Table 1 shows the comparison results on the TRECVid AVS task. We test them on two benchmarks datasets [1,2] across four query sets released in the years of 2016–2019. Although the dimensionality reduction brings some drops in the performances on most query sets, the drops only happen on a relatively low ratio of queries. Only 38 out of 120 queries happen to drop in both the embedding search and concept search. Most of them are complex queries which describing rich interaction between human and object. For example, the query "Find shots of a person holding, talking or blowing into a horn" has degraded from 0.246 to 0.031 on embedding search. A big drop also happens on the query "Find shots of a person holding, opening, closing or handing over a box". The performance degradation might due to the lower capacity of the model in low-dimensional space in encoding complex information. We also try the indexing method KGraph[1]. While the speed is improved to 0.1 seconds per query, the retrieval performance drops further.

3 The SQL-Like Interface

The SQL-like interface is presented in Fig. 1. Rather than providing one text box only for users to input the query, we allow users to manually specify the search content (subject, predicate, object) and constraint (logical condition) in a semi-structured way. For the constraints, in each text box, users can specify the "OR" relation between terms using commas, and "AND" relation using semicolons. The "NOT" text box is allocated for the "NOT" statement in the query. Two examples of how a user can express queries into these text boxes are illustrated in Fig. 1. The system will decompose the SQL-like query into multiple sub-queries based on the constraint being claimed. For example, the query in Fig. 1(a) is parsed into two sub-queries: "two people kissing" and "bride and groom". These sub-queries are separately fed into the dual-task model to retrieve two sets of similar videos. In this example, the top-rank videos retrieved by the first query ("two people kissing") will be downgraded to lower rank if they are also ranked high by the second query ("bride and groom"). When a query is short, e.g., "woman wearing glasses", the user can simply input the whole query in one of the text boxes while leaving the remaining boxes blank.

[1] https://github.com/aaalgo/kgraph.

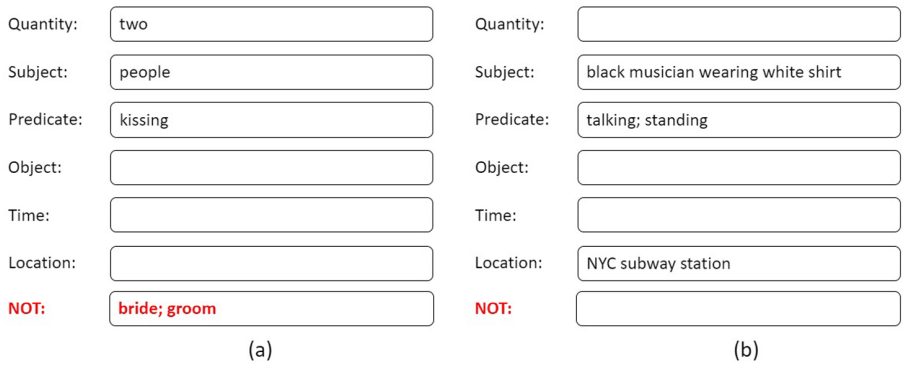

Fig. 1. The SQL-like interface for interactive search. The figure shows two examples of how to express queries. (a) An ad-hoc query "Find shots of two people kissing who are not bride and groom"; (b) A textual known item search (KIS) query "A black musician standing in a NYC subway station and talking to people. He wears a white shirt". The ad-hoc query is decomposed into two sub-queries and the KIS query is translated into "black musician wearing white shirt talking and standing in NYC subway station".

4 The Vireo Video Search System

We integrate the dual-task model presented in Sect. 2 and the SQL interface presented in Sect. 3 into our video search system. In the end, the Vireo-VSS

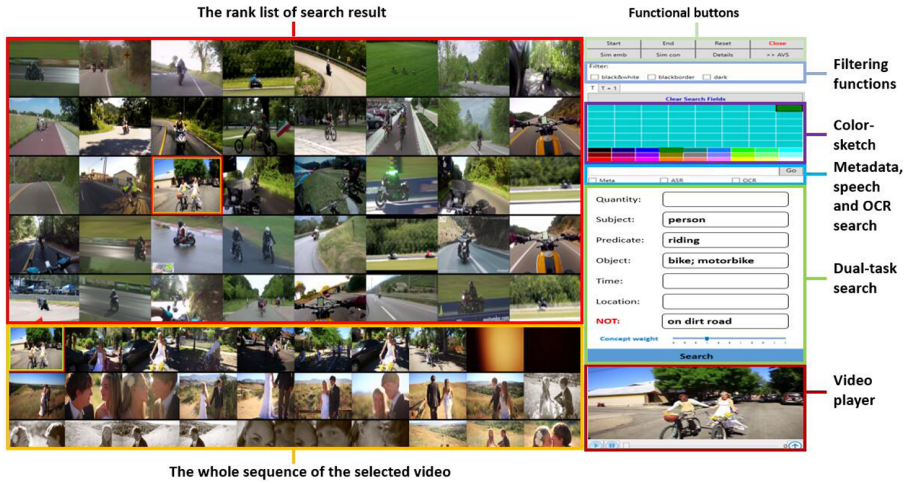

Fig. 2. Vireo-VSS provides multiple querying methods. In this example, a text query is combined with the color-sketch query to search for videos with "A person riding a bike or motorbike while not on a dirt road, and the resulting videos are constrained by the green color on the top-right corner" (Color figure online).

(shown in Fig. 2), which is going to be demonstrated in the VBS 2021, includes the following modules and functions:

- *Query-by-sketch.* We keep using the simplified color-sketch retrieval model presented in [6] because of its promising performance in solving visual known-item search task.
- *Query-by-text.* We provide two approaches for text-based search. First, a user can input a query into a text box to search for any text in the metadata, detected on-screen text, or video speech [6]. Second, a user can parse and input a query into the interface presented in Sect. 3 to search using our dual-task model.
- *Query-by-example.* We utilize our approach in [7] which employs the nearest neighbor search for master-shot key-frames in the video dataset. Instead of using the feature extracted from CNN for matching, we use the embedding feature and the concept feature of the video segments extracted from our dual-task model.
- *Temporal query.* As the temporal query is useful in solving textual known-item search query, we keep this function used in [8]. It is noted that this function is implemented for both sketch-based and text-based search, and it allows users to specify queries in time t and time $t + 1$.
- *Filtering.* We provide three filtering functions to filter out black and white, black bordered, and dark video frames.

5 Conclusion

We have presented two new features, dual-task model for cross-modal search and SQL-like querying interface, in the Vireo-VSS. Furthermore, we devise the dual-task model by dimensionality reduction for real-time search, at the expense of a slight drop in search performance. To make search results predictable and tractable, we restrict the way that users formulate a query by filling in a SQL-like template. As the concept-free approach is insensitive to logical relation, the search system addresses this problem by automatically generating multiple sub-queries for feature embedding and search. We expect that these changes will make Vireo-VSS more capable of dealing with complex and verbose queries.

Acknowledgement. The research was partially supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant and the National Natural Science Foundation of China (No. 61872256).

References

1. Awad, G., et al.: Trecvid 2016: evaluating video search, video event detection, localization, and hyperlinking. In: TRECVID 2016 Workshop (2016)
2. Berns, F., Rossetto, L., Schoeffmann, K., Beecks, C., Awad, G.: V3C1 dataset: an evaluation of content characteristics. In: ICMR, pp. 334–338 (2019)
3. Li, X., Xu, C., Yang, G., Chen, Z., Dong, J.: W2vv++: fully deep learning for ad-hoc video search. In: ACM MM (2019)

4. Lokoč, J., et al.: Interactive search or sequential browsing? A detailed analysis of the video browser showdown 2018. ACM TOMM **15**(1), 29:1–29:18 (2019)
5. Lokoč, J., Bailer, W., Schoeffmann, K., Muenzer, B., Awad, G.: On influential trends in interactive video retrieval: video browser showdown 2015–2017. IEEE TMM **20**(12), 3361–3376 (2018)
6. Nguyen, P.A., Lu, Y.-J., Zhang, H., Ngo, C.-W.: Enhanced VIREO KIS at VBS 2018. In: Schoeffmann, K., et al. (eds.) MMM 2018. LNCS, vol. 10705, pp. 407–412. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73600-6_42
7. Nguyen, P.A., Ngo, C.-W., Francis, D., Huet, B.: VIREO @ video browser showdown 2019. In: Kompatsiaris, I., Huet, B., Mezaris, V., Gurrin, C., Cheng, W.-H., Vrochidis, S. (eds.) MMM 2019. LNCS, vol. 11296, pp. 609–615. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05716-9_54
8. Nguyen, P.A., Wu, J., Ngo, C.-W., Francis, D., Huet, B.: VIREO @ video browser showdown 2020. In: Ro, Y.M., et al. (eds.) MMM 2020. LNCS, vol. 11962, pp. 772–777. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-37734-2_68
9. Rossetto, L., et al.: Interactive video retrieval in the age of deep learning - detailed evaluation of VBS 2019. IEEE TMM 1 (2020)
10. Wu, J., Ngo, C.W.: Interpretable embedding for ad-hoc video search. In: ACM MM (2020)