Exquisitor at the Video Browser Showdown 2021: Relationships Between Semantic Classifiers

Omar Shahbaz Khan[1], Björn Þór Jónsson[1], Mathias Larsen[1], Liam Poulsen[1], Dennis C. Koelma[2], Stevan Rudinac[2], Marcel Worring[2], and Jan Zahálka[3]

[1] IT University of Copenhagen, Denmark
[2] University of Amsterdam, Netherlands
[3] Czech Technical University in Prague, Czech Republic

Abstract. Exquisitor is a scalable media exploration system based on interactive learning, which first took part in VBS in 2020. This paper presents an extension to Exquisitor, which supports operations on semantic classifiers to solve VBS tasks with temporal constraints. We outline the approach and present preliminary results, which indicate the potential of the approach.

Keywords: Interactive learning · Video browsing · Temporal relations.

1 Introduction

The Video Browser Showdown (VBS), now in its 10th anniversary edition, has emerged as an important vehicle for the evolution of the multimedia field [5]. During VBS, researchers are given a series of never-before-seen task descriptions, based on a collection of 7,475 video clips [9], and asked to interactively retrieve either one specific video segment or multiple relevant segments, depending on the task type. VBS allows researchers working on media exploration and search tools to apply their techniques in a realistic setting and better understand the pros and cons of both the underlying techniques and the interfaces. The lessons learned during the competition can then inspire new methods and further research. In addition, the competitive setting makes for an exciting event where the ranking of systems can also give hints to their usability and applicability.

Exquisitor, a prototype media exploration system based on interactive learning, took part for the first time in VBS 2020, where it placed 5th out of 11 systems [2]. The goal of Exquisitor, as applied to VBS, is to build a semantic classifier for the information need represented in each task, and use that classifier—along with metadata filters and a video timeline explorer—to solve the task. Exquisitor uses the video segmentation supplied with the VBS collection and represents each video segment independently by semantic features derived from its keyframe. When building the semantic classifier, Exquisitor suggests keyframes to the user and asks for feedback on those suggestions. Once the user spots a potentially relevant keyframe, the video explorer can then be used to explore the actual content and internal structure of the full video clip.

For many VBS tasks, the task description applies to more than one video segment, often focusing on different semantic concepts in different segments, and sometimes providing an explicit temporal relationship. Unsurprisingly, therefore, all the strongest VBS competitors provide temporal queries as a major technique [4, 6, 10, 7]. Since video segmentation tends to split the video by semantic concepts, a classifier built to find one segment may not find the other, and the system should provide support to utilise the relationship between concepts in video segments.

In this paper, we present a new version of Exquisitor, where the major extension is the support for utilising relationships between semantic classifiers. While each semantic classifier is developed in the same manner as before, using independent video segments, the results of two semantic classifiers can now be combined in various ways, with an optional temporal relationship specification. In this paper we briefly outline the method and interface for combining two semantic models and show how two models combined could be used to solve two VBS 2020 tasks, one of which the team failed to solve during the competition. We will present and evaluate the methods in more detail in a later publication.

2 Exquisitor

Exquisitor is a user relevance feedback approach capable of handling large scale collections in real time [3, 8]. The Exquisitor system used for VBS consists of three parts: (1) a web-based user interface for receiving and judging video suggestions; (2) an interactive learning server, which receives user judgments and produces a new round of suggestions; and (3) a web server which serves videos and video thumbnails. Due to the computational efficiency of the system, all three components can run locally on a laptop.

Exquisitor Server: Exquisitor is fueled by a semantic model that combines interactive multimodal learning with cluster-based indexing. Each keyframe in each modality is represented by an efficient representation containing the most important semantic features, compressed using an index-based method [11]. This representation is further clustered using a cluster-based indexing approach [1]. When building a semantic classifier C, a linear SVM classifier is iteratively refined based on user interactions (positive and negative examples). In each round of interaction, the resulting separating hyperplane forms k-farthest neighbour queries posed to the cluster-based indexes. Finally, late fusion is performed on the retrieved results, to produce the 25 top-ranked results to suggest to the user.

Exquisitor Interface: The interface for building classifiers is shown in Figure 1. By hovering over a keyframe, the user can choose to view the video, submit it to the VBS server, label it as a positive/negative example, or mark it as seen. Using the 'next' button, the user can also mark all videos as seen and get a full screen of new videos based on the current semantic classifier. Positive (green column) and negative (red column) examples are immediately used to update the model.

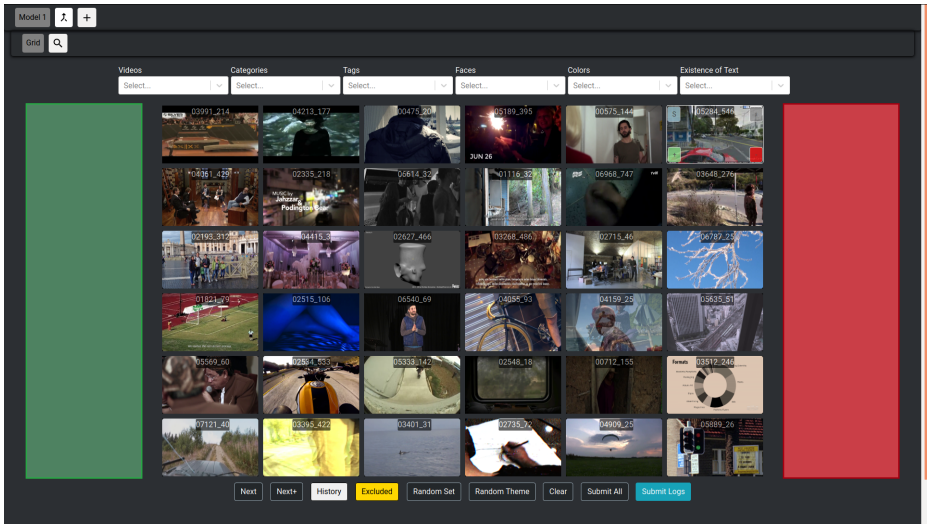

Fig. 1. Exquisitor's interface for building semantic classifiers. See text for details.

Interactive Learning and VBS: The tasks in VBS have three different flavours: Textual Known-Item-Search (KIS) tasks present a gradually evolving text description matching a short video segment; Visual KIS tasks show the video clip sought; and Ad-hoc Video Search (AVS) tasks ask for all segments matching a description. In these tasks, the aim of interactive learning is to create a classifier that is good enough to bring the correct answer(s) to the screen. For KIS tasks, a submitted result is considered as a positive example; once the correct result has been submitted the task is complete. For AVS tasks the process is identical, except that all videos on screen can be submitted at once using a special button, and the process only ends once time has expired.

3 Operations on Semantic Classifier Rankings

To ground the presentation, consider the two textual KIS tasks in Table 1, both of which have a temporal component. Task T_1 was solved by 6 teams during VBS 2020, and was generally considered a difficult task. There are many videos with bridesmaids and brides and grooms, respectively, but in this particular video they do not co-occur in a keyframe during the segment that was considered a solution to the task, and hence we failed to solve this task. Task T_6, on the other hand, was the only text-based task solved by all teams. The Exquisitor team solved it efficiently during the competition by building a classifier for elevators, since (a) the elevator and the bike co-exist in the same keyframe and (b) elevators are rare, so the keyframe is quickly suggested for inspection. Note, however, that since there are many examples of bikes in the collection, but most of them outdoors, building a classifier for bikes is not a productive method to solve T_6.

Table 1. Two example textual KIS tasks from VBS 2020.

Task	Description
T_1	Seven bridesmaids in turquoise dresses walking down a street, and three still images of the bride and couple. The bridesmaids walk on the sidewalk towards the camera. The photos of the couple and bride are taken in a park.
T_6	Red elevator doors opening, a bike leans inside, doors closing and reopening, bike is gone. Zoom-in on bike, zoom-out from empty elevator. The bike is silver, the text 'ATOMZ' is visible.

Classifier Ranking Operations: The rankings obtained by two semantic classifiers, C_1 and C_2, can be combined with a keyframe relationship operation, C_1 *op* C_2, where $op \in \{\cap, \cup, \setminus, \dot{-}\}$. Furthermore, a temporal constraint can optionally be added, which requires either a maximum distance between keyframes (*within* <*frames*>) or a minimum distance (*after* <*frames*>). The result of the classifier ranking operation is a list of videos satisfying both the relationship constraint and optional temporal constraint. Each video is represented by a list of keyframes, annotated by the classifier(s) they appear in, and the videos are ranked by an average score based on the accumulated rank of their scenes from each classifier and the total number of scenes.

As an example, consider solving task T_6 by intersection of rankings produced by semantic classifiers for bikes and elevators. A video would be returned as an answer only if both classifiers return a scene from that video. Since the task description indicates that the two elements should be close to each other, a temporal constraint of *within* 1, for example, would avoid videos where bikes and elevators are far apart.

User Interface: Figure 2 shows the interface for classifier ranking operations. As the figure shows, the result of the merge is a list of the 10 top-ranked videos, where each video is represented by three colour-coded keyframes. Yellow keyframes are from C_1 and blue from C_2, while keyframes appearing in both classifiers are shown as green. The interface shows the highest ranked frame of each colour; if no keyframe appears in both classifiers, the third frame is the second highest frame from one classifier. Additionally, summary information on the number of keyframes in the video and classifiers is shown to the left of the keyframes.

Evaluation: To evaluate the usefulness of classifier ranking operations, we attempted to solve the two tasks of Table 1, both by building a single classifier and by building two classifiers and intersecting their rankings. These experiments were carried out in a calm setting, with no time limit, unlike the competitive environment of VBS. Furthermore, for this evaluation, the entire task text was considered. To estimate the user workload, we counted the number of interactions with the system, where an interaction is any action taken by the user, such

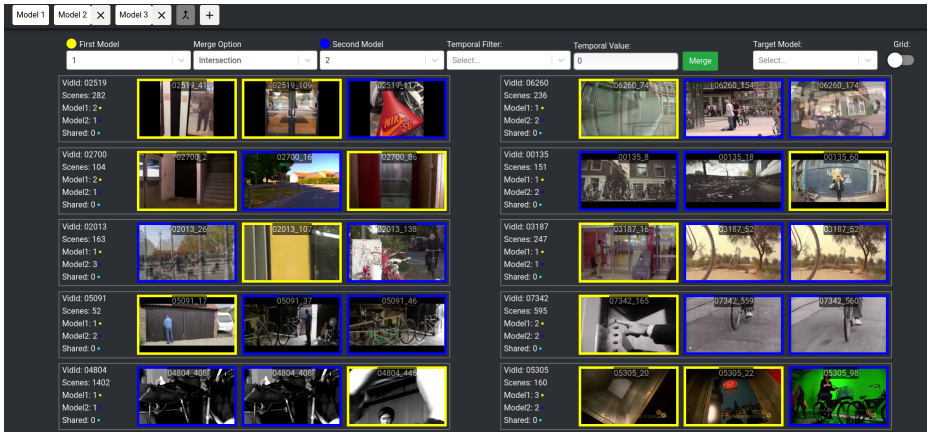

Fig. 2. Exquisitor's interface for semantic classifier operations. See text for details.

as labelling a keyframe as a positive example or changing to a different interface component. We chose to stop after around 75 interactions; once we reached this limit, we considered the task to be unsolved.

Table 2 summarises the results for the two tasks. Consider first T_1, a difficult task which was not solved by Exquisitor during the competition. As the table shows, a simple intersection of the results produced by two classifiers could solve the task. Now consider task T_6, which was significantly easier. Table 2 shows that a single classifier on 'elevator' is the fastest approach to solve this task, due to the composition of the collection; this was fortunately the approach taken during the competition. Had we chosen to focus on 'bike' instead, however, the results suggest we would have failed to solve the task. Building rough classifiers for each concept and intersecting their rankings, however, is also an efficient method to solve the task; since the order in which the models are built does not matter the method is robust.

Table 2. Effectiveness experiment results

Task	Models	Interactions	Solved
T_1	'bridesmaid'	76	No
	'bride'	78	No
	'bridesmaid' ∩ 'bride'	60	Yes
T_6	'elevator'	8	Yes
	'bike'	75	No
	'elevator' ∩ 'bike'	15	Yes

4 Conclusions

We have outlined an extension to the Exquisitor system, supporting operations on semantic classifiers to solve VBS tasks with temporal constraints. Our preliminary results indicate that this new approach has significant potential, and we look forward to testing the approach in the competitive setting.

Acknowledgments: This work was supported by a PhD grant from the IT University of Copenhagen and by the European Regional Development Fund (project Robotics for Industry 4.0, CZ.02.1.01/0.0/0.0/15 003/0000470).

References

1. Guðmundsson, G.Þ., Jónsson, B.Þ., Amsaleg, L.: A large-scale performance study of cluster-based high-dimensional indexing. In: Proc. Int. Workshop on Very-large-scale Multimedia Corpus, Mining and Retrieval (VLS-MCM). Firenze, Italy (2010)
2. Jónsson, B.Þ., Khan, O.S., Koelma, D., Rudinac, S., Worring, M., Zahálka, J.: Exquisitor at the Video Browser Showdown 2020. In: Proc. MultiMedia Modeling (MMM). Daejeon, South Korea (2020)
3. Khan, O.S., Jónsson, B.Þ., Rudinac, S., Zahálka, J., Ragnarsdóttir, H., Þorleiksdóttir, Þ., Guðmundsson, G.Þ., Amsaleg, L., Worring, M.: Interactive learning for multimedia at large. In: Proc. European Conference on IR Research (ECIR) (2020)
4. Kratochvíl, M., Veselý, P., Mejzlík, F., Lokoč, J.: SOM-Hunter: Video browsing with relevance-to-SOM feedback loop. In: Proc. MultiMedia Modeling (MMM). Daejeon, South Korea (2020)
5. Lokoč, J., Kovalčík, G., Münche r, B., Schöffmann, K., Bailer, W., Gasser, R., Vrochidis, S., Nguyen, P.A., Rujikietgumjorn, S., Barthel, K.U.: Interactive search or sequential browsing? A detailed analysis of the Video Browser Showdown 2018. ACM TOMM **15**(1) (2019)
6. Lokoč, J., Kovalčík, G., Souček, T.: VIRET at Video Browser Showdown 2020. In: Proc. MultiMedia Modeling (MMM). Daejeon, South Korea (2020)
7. Nguyen, P.A., Wu, J., Ngo, C.W., Francis, D., Huet, B.: VIREO @ Video Browser Showdown 2020. In: Proc. MultiMedia Modeling (MMM). Daejeon, South Korea (2020)
8. Ragnarsdóttir, H., Þorleiksdóttir, Þ., Khan, O.S., Jónsson, B.Þ., Guðmundsson, G.Þ., Zahálka, J., Rudinac, S., Amsaleg, L., Worring, M.: Exquisitor: Breaking the interaction barrier for exploration of 100 million images. In: Proc. ACM Multimedia. Nice, France (2019)
9. Rossetto, L., Schuldt, H., Awad, G., Butt, A.A.: V3C - a research video collection. In: Proc. MultiMedia Modeling (MMM). Thessaloniki, Greece (2019)
10. Sauter, L., Parian, M.A., Gasser, R., Heller, S., Rossetto, L., Schuldt, H.: Combining boolean and multimedia retrieval in vitrivr for large-scale video search. In: Proc. MultiMedia Modeling (MMM). Daejeon, South Korea (2020)
11. Zahálka, J., Rudinac, S., Jónsson, B.Þ., Koelma, D.C., Worring, M.: Blackthorn: Large-scale interactive multimodal learning. IEEE TMM **20**(3) (2018)