Computer Fraud and Security
ISSN (online): 1873-7056

Semantic-Conditional Network for Micro-Video Summarization

Xiaowei Gu*
School of Software Engineering, South China University of Technology,
Guangzhou, 510006, Guangdong, China.
Corresponding Author: Xiaowei Gu, Email: amyxwgu@163.com

Abstract: The goal of video summarization is to extract key information from a raw video so that long videos can be interpreted in a short time without losing much semantic information. Previous methods primarily consider the diversity and representation of the obtained summary without paying sufficient attention to the semantic information of the resulting frame set, especially when generating summaries motivated by user queries. In this paper, we break the conventions in conditional video summarization and propose a new model to accept user queries semantically, namely Semantic-Conditional Network (SC-Net). Technically, for each video, we first search the semantically relevant video frames via a cross-modal retrieval model to convey the comprehensive semantic information in the user query. The rich semantics are further regarded as semantic prior to trigger the optimization of the summarization network, which produces summaries in a diverse and representative way. Furthermore, a novel one-stage training strategy optimizes the time complexity from polynomial to linear. Extensive experiments on publicly available datasets demonstrate promising results compared with state-of-the-art methods.
Keywords: User query; Semantic web; Video analysis; Micro-video; Conditional video summarization

1. Introduction

As one of the most popular media types, micro-videos have undoubtedly shown an upward trend in recent years. The volume of user-generated micro-videos uploaded on various platforms, including TikTok (https://www.tiktok.com/) and Kwai (https://www.kwai.com/), has witnessed an explosive surge. Taking Kwai as an example, roughly tens of millions of micro-videos are recorded and published every day. The tremendous amount of data brings new challenges to video search within and among videos. Video summarization plays an important role in improving search efficiency by representing videos with concise but semantically informative summaries.

Facing the huge number of micro-videos, users on the micro-video platform conduct millions of querying requests per day to obtain the desired micro-videos. One effective approach for micro-video retrieval is conditional video summarization, also known as query-focused video summarization. Compared with generic video summarization, conditional video summarization takes into account both the relevance to a given search query and the representativeness of the original video. This task naturally connects natural language processing and computer vision by perceiving text query and interpreting it in video summarization. It is helpful for search engines to generate customized snippets of videos according to user queries.

The dominate force in current state-of-the-art methods [10, 16, 20, 23, 26] primarily considers the diversity and representation of the obtained summary without paying sufficient attention to the semantic information of the resulting frame set, especially when generating summaries motivated by user queries. However, the short duration and low quality of micro-videos hinder the efficient association between query intent and videos. In order to obtain more story-telling summaries, high-level semantics should be taken into account.

Our work formulates the problem of conditional video summarization as a Semantic-Conditional Network. We devise an efficient optimization framework and improve the efficiency in processing tremendous micro-videos. The main contributions of this work are listed as follows:

- We propose a novel Semantic-Conditional Network (SC-Net) that provides efficient conditional summarization for micro-videos. Unlike the traditional methods, SC-Net first search the semantically relevant video frames via a cross-modal retrieval model to convey the comprehensive semantic information in the user query.

- Based on the above network structure, we propose an optimization framework for learning the summarization network with rich semantic priors. The time complexity is optimized from polynomial to linear.

- We demonstrate the effectiveness of the proposed optimization framework in conditional micro-video summarization. Two general video features are utilized to verify that our method is robust to feature changes.

2. Materials and Methods

2.1 Related Work

Conditional or query-focused video summarization generates user-oriented summaries according to user queries [12, 26]. Research [20, 21] developed a probabilistic model based on sequential DPP [7, 15] to capture information from lengthy videos. Research [23] leveraged relevance model and submodular functions[9]. Study [16] formatted the conditional video summarization problem as submodular span to remove the redundancy. Research [19] employed non-monotone submodular functions which improved the summarization results. Some studies [10, 11, 25, 26] adopted deep neural networks to generate user preferred video summaries. Research [24] verified that semantic information was helpful in video summarization.

Early semantic awareness works mainly exploited the relevance between a query and a video frame to achieve semantic awareness. Text queries were converted to be represented by a vector generated based on a predefined dictionary [10, 17, 20, 21, 23, 27]. However, too many search words are available in micro-video, and adding new words is time-consuming, which limits the semantic information covered. Some other works computed the visual similarity between the user query and each video frame [16, 19]. The semantically relevant frames would be extracted as the summary. Although the method is innovative, it depends on the visual quality of a micro-video, which is generally low due to the usage of handheld devices. Our work will explicitly address this issue by adopting the semantic information as prior to guide the learning of the summarization network and hence be able to encourage a better semantic alignment between the input query and the micro-video.

2.2 General data

We conduct the experiments on OVP[7] and YouTube[5] datasets. There are 50 videos in the OVP dataset covering several areas: documentary, educational, ephemeral, historical, lecture, etc. All videos are in MPEG-1 format (30 fps, 352 × 240 pixels). The video duration is from 1 to 4 minutes. YouTube dataset has 39 video clips collected from websites like YouTube. It covers various categories, including news, sports, commercials, TV-shows, and home videos. We exclude the cartoon videos from the original dataset. The videos are 1-10 minutes long.

2.3 Evaluation Metric

Following the previous studies[5], we compute the pairwise distances between a generated summary and the user-annotated ground truth. A frame is limited to appearing in the matched pairs at most once. Two frames are similar if the distance between them is less than a predetermined threshold, which is equal to 0.5 in our experiment. After the matching, we adopt F-score (F), Precision (P), and Recall (R) as the evaluation metric. The calculation of the scores is as follows:

$$\mathrm{P} = \frac{length(gs \cap gt)}{length(gs)}, \mathrm{R} = \frac{length(gs \cap gt)}{length(gt)}, \mathrm{F} = \frac{2 \times \mathrm{P} \times \mathrm{R}}{\mathrm{P}+\mathrm{R}}. \quad (1)$$

2.4 Implementation Details

We preprocess the experimental data by pre-sampling the videos at 2 fps uniformly. We use the deep feature extracted from the Pool 5 layer of the GoogLeNet model, which is pre-trained with ImageNet. The feature

dimension of each frame is 1024. During the training stage, all the parameters used in our network are learned using AdaGrad optimizer and L1 loss. The learning rate is 10^{-5} with a weight decay of 10^{-5}. The network is implemented under the PyTorch framework.

2.5 Preliminaries

This section introduces submodular functions, their optimization process, and matroids. Given two sets $A \subseteq B \subseteq V \setminus \{j\}$ and any element $j \in V \setminus B$, the definition of a submodular function is $f(A \cup j) - f(A) \geq f(B \cup j) - f(B)$. This is the diminishing returns property, i.e. the incremental value of adding a new element decreases with the growth of the set size. It is useful for data summarization[3]. A submodular function is monotone if $f(A \cup j) - f(A) \geq 0$ for all $A \subseteq V$ and $j \in V \setminus A$. In this paper, we allow f to be non-monotone.

Although the traditional maximization method is discrete, it is helpful to lift it to a continuous domain. The continuous extension is a function $F: [0,1]^V \rightarrow \mathbb{R}_+$, whose value agrees with f for the integer elements. The general framework for continuous optimization consists of three steps: 1) to lift the submodular function to a continuous extension, 2) to design a maximization algorithm and optimize the continuous utility function, 3) to round the fractional results of step 2) to the integer solution set. Our optimization process follows this general framework in the continuous domain.

A matroid is a pair $\mathcal{M} = (V, \mathfrak{I})$, where $\mathfrak{I}$ is a family of independent sets of ground set V that satisfies the following properties: 1) $\emptyset \in \mathfrak{I}$; 2) heredity property, i.e., $I_1 \subseteq I_2 \in \mathfrak{I} \Rightarrow I_1 \in \mathfrak{I}$; 3) exchange property that is $I_1, I_2 \in \mathfrak{I}, |I_1| < |I_2| \Longrightarrow \exists v \in I_2 - I_1 : I_1 \cup v \in \mathfrak{I}$. The uniform and partition matroids are examples of matroids. A uniform matroid is the family of all subsets with cardinality at most k, where $k \leq n$ is a nonnegative integer (n is the total number of elements in ground set V). The partition matroid is built on the uniform matroid. A partition of V is the collection of disjoint nonempty subsets V_i of the ground set V, indexed by the integer i. A partition matroid includes all subsets $S \subseteq V$ where $V_i \cap S$ has a cardinality at most k. It is a uniform matroid of V_i for each index i. The matroid contains subsets with independent elements analogous to a summary, offering the desirable property for data summarization. Furthermore, matroids can be generalized to the continuous domain as matroid polytopes, which are easy to optimize. We adopt the partition matroid as the constraint of the submodular function and leverage it in semantic retrieval to capture semantic information from user queries. The details are presented in Section 2.6.

Our Semantic-Conditional Network (SC-Net) aims to maximize a non-monotone submodular function subject to partition matroid constraints for conditional video summarization. We partition the original video according to the frame-level similarity to the user query. A continuous extension is utilized to find the best summary set that maximizes the value of the submodular function and satisfies the constraint. Semantic-Conditional projection and local rounding methods are proposed for semantic-focused query processing. We use V to denote a video with n frames. The X_V is the frame-level features extracted from the video V, and $S \subseteq V$ is any possible summary matching user query Q_V. The functions and variables with overlines are in the continuous domain, e.g., $\bar{F}, \bar{S}$.

2.6 Semantic-Conditional Network (SC-Net)

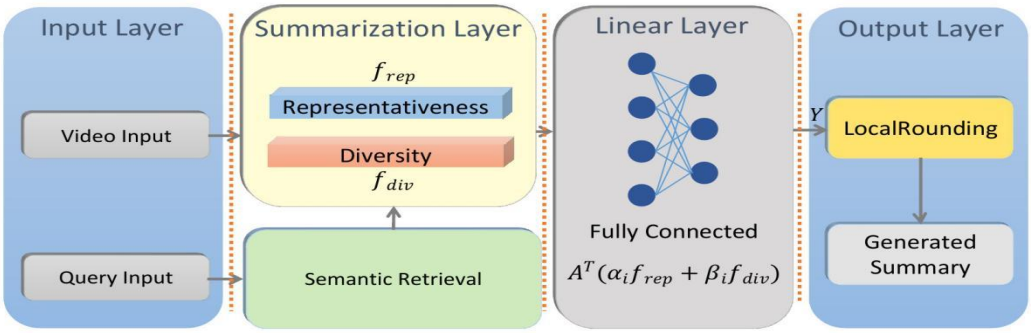

Figure 1 The overall network architecture of our Semantic-Conditional Network (SC-Net)

2.6.1 Overall Network

We formulate the conditional micro-video summarization task as maximizing a non-monotone submodular function subject to a partition matroid constraint as follows:

$$S^* = \underset{S \in \mathcal{M}}{\operatorname{argmax}} f(X_V, S, Q_V), \tag{2}$$

where S^* is the selected optimal summary, $\mathcal{M}$ is the partition matroid constraint, $\mathrm{X_V}$ and $\mathrm{Q_V}$ are the features and user queries of video V.

As illustrated in Figure 1, our network includes multi-layer submodular functions similar to deep neural networks (DNNs), including an input layer, a summarization layer, a linear layer, and an output layer. Like DNNs, the multi-layered architecture enables interaction within multiple layers and extracts data representation at a more abstract level[3]. The f_{rep} and f_{div} are functions to model representativeness and diversity, respectively. They both satisfy submodularity but are not necessarily to be monotone. We define the objective function f as a combination of the f_{rep} and f_{div} after a fully connected layer with non-negative weights, since submodularity is preserved under non-negative linear combinations[14]:

$$f = A^T Y(X_V, S, Q_V), \tag{3}$$

where $Y(X_V, S, Q_V) = [f_1(X_V, S, Q_V), f_2(X_V, S, Q_V), \ldots, f_L(X_V, S, Q_V)]^{\mathrm{T}}$, with $l = 1, 2, \ldots, L$, each $f_l(X_V, S, Q_V) = \alpha_l f_{rep}(X_V, S, Q_V) + \beta_l f_{div}(X_V, S)$, L is the number of nodes in the fully connected layer. We use $L = 5$ in our experiment. A, α_l, and β_l are the trainable weight parameters. During optimization, all the parameters are constrained to be non-negative to preserve submodularity.

Representativeness functions measure how well the selected summary represents the original video. One can score the subset highly if it contains the major information of the ground set V. We use Equation (4) to model the representativeness as a facility location problem[22]. A video frame is represented with its closest frame in the summary.

$$f_{rep}(X_V, S, Q_V) = \sum_{i \in V} \left(1 - \min_{j \in S} s_{i,j}(Q_V, S)\right), \tag{4}$$

where $s_{i,j}(Q_V, S) = \left\| X_{Vi}(Q_V) - X_{Vj}(S) \right\|_2$ is the Euclidean distance, and $1 - s_{i,j}$ is the measure of similarity. $X_{Vi}(Q_V)$ returns the feature X_{Vi} if the ith frame is in query set Q_V. Otherwise, it returns zero. And $X_{Vj}(S)$ works the same way. The value of $s_{i,j}$ is normalized to $[0, 1]$.

Diversity functions aim to include the various content in the video. They measure the repulsiveness among the summary frames and eliminate redundancy. Although submodularity is still quite natural in diversity functions, monotonicity sometimes is not. The Determinantal Point Process (DPP)[15] is a powerful tool for modeling diversity in video summarizations. It is a model quantifying the discrete probabilistic distribution.

$$f_{div}(X_V, S) = \frac{det(L_S)}{det(I+L)}, \tag{5}$$

where the $n \times n$ kernel L is the pairwise frame-level similarity, $L_S = \left[L_{i,j}\right]_{i,j \in S}$, is the principal minor with rows and columns selected according to the indices in S, and I is the $n \times n$ identity matrix. Assuming there are two identical frames in the selected subset, L_S will have duplicate rows and columns, which will result in a zero-valued determinant.

We don't take the user query Q_V as an input of $f_{div}(\cdot)$ because the partition matroid constraint of the objective function will keep the summary set S relevant to the query.

2.6.2 Semantic Retrieval

Our Semantic Retrieval model consists of several steps. First of all, the semantic information is recognized from user queries. Then, the rich semantics are taken as semantic prerequisites using partition matroid to trigger further processing. Before learning the SC-Net, the semantic conditions are extended to the continuous domain. In the following learning procedure, the semantic information is learned through the network parameters.

Query Processing A query set $Q_V \subseteq V$ refers to one or more visual or text queries. To support the partition matroid constraint, we calculate the relevance of each frame to the query set. The frame-level relevance to the query set can be estimated via average similarity measurements. We adopt the cosine similarity

$$f_{sim\_i}(X_{Vi}, Q_V) = 1 - \frac{1}{q}\sum_{j=1}^{q} \frac{X_{Vi} \cdot Q_{Vj}}{\|X_{Vi}\|\|Q_{Vj}\|}, \tag{6}$$

where i denotes the ith frame of video V, and q is the number of user inputs in the query set.

We segment the video frames into partitions, one containing frames with relevance scores higher than a threshold and the other containing the rest. Partition matroid constraints are imposed when optimizing the summarization result. In our experiment, the relevance threshold is set as the median similarity of all the video frames. The details will be covered in Subsection 2.6.3.

Partition matroid constraint We divide the video frames into two parts, V_1 and V_2, according to the relevance threshold computed by Equation (6). Suppose the set V_1 contains frames with higher similarity to the user query. Then the summary set $S \subseteq V$ satisfies $S \cap V_1 \leq k$ and $S \cap V_2 = 0$, where k is the cardinality constraint. In our experiment, we set k as 15% of the total video frames.

Following the general framework in Section 2.5, the above partition matroid constraint is generalized to a polytope in the continuous domain, i.e.,

$$P(\mathcal{M}) = \left\{ x \in [0,1]^V \middle| \forall j \in \{0,1\}: \sum_{i \in V_j} x_i \leq k_j \right\}, \tag{7}$$

where $k_0 = k$, and $k_1 = 0$.

Continuous extension According to the general framework in Section 2.5, we need to lift the problem specified in Equation (2) to a continuous domain. The Deep Submodular Function (DSF) concave extension [1, 3] is adopted because it is easy to obtain and optimize. The continuous objective function is

$$\bar{Y}^* = \underset{\bar{Y} \in P(\mathcal{M})}{\operatorname{argmax}} \bar{F}(X_V, \bar{Y}, \bar{Q}_V), \tag{8}$$

where $\bar{F}: [0,1]^n \longrightarrow \mathbb{R}_+$ is the objective function in the continuous domain, P is the matroid polytope constraint specified in Equation (7), and the overline represents continuous variables. To obtain the natural concave extension $\bar{F}$ of f, the discrete variables, S and Q_V, are replaced with real values. $\bar{S}$ and $\overline{Q_V}$ are n-dimensional vectors with $\bar{S} = 1_S$ and $\bar{Q}_V = 1_{Q_V}$. The $1_S \in \mathbb{R}_+^V$ is 0 if a frame is not in set S, and 1 if a frame is in set S. And 1_{Q_V} works similarly.

Semantic Learning The parameters in the network are learned using annotated summary frames to capture the underlying frame selection criteria. Unlike the existing methods of conditional video summarization, our training requires only one stage and is more computationally efficient. We employ the modern DNN training mechanism to train our submodular network. The training details can be found in the supplementary material.

2.6.3 Optimization

This section presents how to generate an optimal result as the output summary. Our approach is inspired by research[8], which showed that similar results could be achieved with much lower computation costs using the monotone algorithms to solve non-monotone submodular function maximization. Despite the effectiveness, their algorithm is for greedy methods only. It is not applicable to continuous methods because the intermediate

variables in the continuous optimization process are fractional. We cannot directly identify which elements are selected by the corresponding monotone algorithm. In addition, conditional video summarization requires similarity between the search queries and the summarized results. The constraint cannot be incorporated straightforwardly into the method of [8]. We now show how to optimize non-monotone cases using the continuous monotone method.

The overall algorithm includes two iterations in the main procedure as two occasions are considered. One occasion is that the current subset $\bar{S}_1$ includes a reasonable fraction of the optimal result. The other is that the optimal result is mainly included in the discarded frames 1-$\bar{S}_1$. More details are in supplementary material. The final summary set S is the better performing one of the two iterations. Here we describe each step in detail.

Mon-Max is a continuous method for monotone submodular function maximization. We adopt the DSF concave extension to lift the problem to the continuous domain by replacing the discrete variables with real values. The details can be found in Subsection 2.6.2.

The Mon-Max module includes an iterative operation: updating $\bar{S}_0$, i.e., the $\bar{X}_1$ input, and projecting the updated $\bar{S}_0$ to the constrained matroid polytope. Since the DSF extension is concave, it can be efficiently maximized via supergradient ascent and projected to the constraint space. The updated value of $\bar{S}_0$ is calculated using a supergradients. The supergradient of $\bar{F}$ is defined as below:

$$\partial\bar{F}(x) = \{g \in \mathbb{R}^n | g^{\mathrm{T}}(x' - x) \geq \bar{F}(x') - \bar{F}(x), \forall x' \in P\}, \tag{9}$$

where P is a compact convex set. The supergradient of $\bar{F}$ is its derivative at its current valuation if it is differentiable.

Specifically, we start from the initial summary set $\bar{S}_0$ and iteratively update $\bar{S}_0$ as below:

$$\bar{S}_0^{(t)} = \bar{S}_0^{(t-1)} + \eta \cdot \partial\bar{F}\left(X_V, \bar{S}_0^{(t-1)}, \bar{Q}_V\right), \tag{10}$$

where η is the learning rate, and the suffix (t) stands for the fractional summary set computed in the tth iteration. We use 100 iterations in our experiment. The constraint is not considered when updating $\bar{S}_0$ and is taken into account in the subsequent projection. The details on projecting the updated $\bar{S}_0$ to the constrained matroid polytope can be found later in this subsection.

LocalRounding takes the fractional summary set $\bar{S}$ from Mon-Max and the continuous extension $\bar{F}$ as inputs. T is the set of frame index in partition V_1. Two sets A_0 and B_0 are initialized for further processing. The initial value of A_0 is set to $\emptyset$. And set B_0 is initialized to include all frames in partition V_1. As specified in Subsection 2.6.2, the set V_1 contains frames with higher similarity to the user query. LocalRounding includes an iterative process with n_1 repetitions, where n_1 is the number of frames in V_1. Each iterative process starts by picking two fractional values from the input set. Rounding is performed to change at least one of them to an integer. The frames that can increase the value of the continuous objective function $\bar{F}$ will remain in the final summary set A_{n_1}. Otherwise, it will be discarded.

Rounding is a subprogram of LocalRounding. The algorithm simplifies the randomized pipage rounding[4] to finish in linear time under partition matroid constraint. The $\hat{y}_{i,j}^{+}(\delta)$ means computing $\hat{y}_i + \delta$ and $\hat{y}_i - \delta$ simultaneously. The $\hat{y}_{i,j}^{-}(\delta)$ just switch the symbol $+$ and $-$. In [4], δ is calculated by the minimal rank difference to ensure that the updated set remains tight. In partition matroid, for a pair of fractional variables, the maximum increase in one variable and the corresponding decrease in the other will not break the constraint.

Let $\mathcal{M}$ be the partition matroid constraint and OPT be the optimal solution. The LocalRounding performs local search for maximizing continuous submodular functions under $\mathcal{M}$ and returns an integral set $S \in V$ with $f(S) \geq \frac{1}{3+1/\alpha}$OPT in $\mathcal{O}(n)$ time. The proof is provided in the supplementary material.

Projection is to find a point on the constraint space $\mathcal{M}$ to minimize the distance of the two points.

$$\mathrm{X}_{i+1} = \underset{x \in P(\mathcal{M})}{\operatorname{argmin}} \frac{1}{2} \|x - \bar{S}_i\|_2^2, \tag{11}$$

where $\bar{S}_i$ is the point to be projected. It is a convex optimization problem and can be solved with the KKT conditions and Lagrangian function.

3 Results

3.1 Quantitative Results

In this subsection, we compare our network with several state-of-the-art video summarization algorithms. VSUMM[5] is a methodology for producing static video summaries based on color feature extraction from video frames and k-means clustering algorithm. The seqDPP[7] treated video summarization as a supervised subset selection problem and overcame the deficiency of the standard DPP by incorporating the sequential structure of video data. It is the summarization part of the conditional video summarization method[20]. The Fantom[18] maximizes a submodular function (not necessarily monotone) for personalized data summarization. The StreamingLS[19] extracted and conditionally summarized the data streams (e.g., video streams) with non-monotone submodular function maximization. The QCVS[10] is an end-to-end method for conditional video summarization based on deep learning. It consists of a video summary controller, video summary generator, and video summary output module. The S3[16] is the submodular span problem that involves finding a large set of elements with small gain relative to a given query set. See Table 1 for details.

Table 1 Comparison with state-of-the-art video summarization methods on OVP and YouTube datasets

Method	OVP			YouTube		
	F-score	Precision	Recall	F-score	Precision	Recall
seqDPP[7]	77.7	75.0	87.2	60.3	59.4	64.9
Fantom[18]	78.0	75.1	**88.6**	60.3	59.1	64.7
StreamingLS[19]	75.6	71.8	86.5	59.8	58.6	64.2
QCVS[10]	79.1	75.8	85.4	61.8	65.0	64.9
S3[16]	78.5	73.9	86.6	61.6	57.7	68.3
TSML[13]	73.4	-	-	62.2	-	-
GADL[2]	77.2	79.2	76.9	69.5	**71.6**	69.6
SVS_MCO[6]	83.3	79.5	84.0	-	-	-
SC-Net (GoogleNet)	**83.4**	**79.8**	87.3	**69.9**	64.2	**76.8**
SC-Net (color)	83.2	**81.1**	85.4	**69.6**	69.0	**70.2**

We compare the running time with the traditional greedy method. The results are shown in Table 2.

Table 2 Analysis results of time complexity with greedy

Time per video (sec)	0.05	0.1	0.15	0.2	0.25	0.3	Avg.
Greedy	24	37	50	63	75	87	56
Ours	28	31	34	35	38	40	34.3

3.2 Ablation Study

We conduct ablation study to understand the contribution of each component in our proposed network and algorithm. The following models are considered: 1) woRep. To understand the contribution of the representativeness part in the full model, we remove it and report the evaluation scores. 2) woDiv. It is a model similar to woRep. The diversity part is removed for further understanding. 3) woTrn. We keep the proposed full model here but ignore the training stage. The parameters in the network are randomly assigned when initialization. It will help us understand the benefit of the learning methodology. 4) woLR. We remove the optimization algorithm LocalRounding for non-monotone submodular functions. The conventional optimization technique of projected supergradient ascent is utilized to get an optimal result. The optimization uses the trained full model. 5) wTLR. This model replaces the proposed LocalRounding with traditional local search and rounding methods. The comparison results are shown in Figure 2.

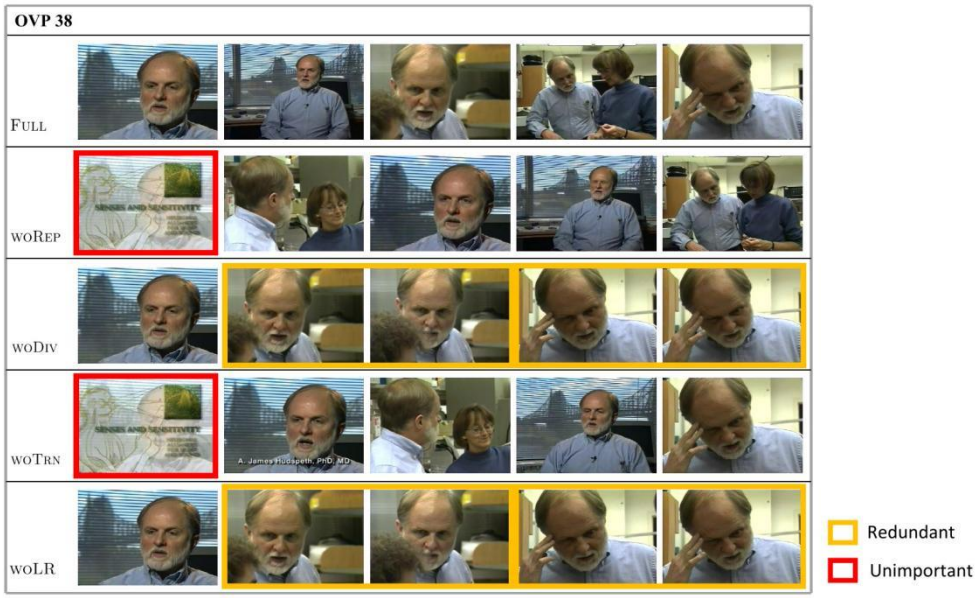

Figure 2 Conditional summary produced by our full model and the ablation models for OVP video 38

3.3 Qualitative Results

Figure 3 demonstrates the selected frames for a person in the query.

Figure 3 Qualitative results on video 39 of the OVP dataset

4 Discussion

4.1 Quantitative Results

Our findings from Table 1 are three-fold: 1) Overall, our algorithm outperforms other baseline methods, which verifies the effectiveness of our SC-Net and the approach of optimization. 2) Across the three evaluation scores, our method has higher Recall rates than Precision, especially for the YouTube dataset. Since the YouTube dataset has fewer videos with more diverse content, it is generally more difficult to summarize. The results

indicate that our method is efficient in capturing the diverse content of the video. 3) Our method outperforms the DNN-based method QCVS, demonstrating the effectiveness of using submodular functions for subset selection.

In Table 2, the first line is the length of the summaries. For example, 0.05 means the summary length is $0.05* n$, where n is the length of the original video. We can find from the table that the time keeps stable for our proposed method as it updates the summary from the subset level. On average, our method is about 40% faster than greedy, which validates the efficiency of our optimization method.

4.2 Ablation Study

We can see that our full model performs best, indicating the effectiveness of each component. From the network point of view, removing the diversity part, woDiv, has more impact than woRep on the evaluation scores. It is essential to capture the diverse video content in a generated summary, and that is why we need to solve the non-monotone diversity problem.

The results of the models woLR and woTLR in the table demonstrate the benefit of our optimization algorithm. For both OVP and YouTube datasets, the model woLR produces the worst performance scores, which validates our optimization method for non-monotone submodular functions is crucial to generate an optimized summary. The performance of woTrn is not so good as the well-trained full model. We believe that the performance gap is attributed to the network parameters that model the internal structure of the video data. Replacing the proposed LocalRounding with the conventional local search and rounding approach significantly decreases the performance, which verifies the effectiveness of our method.

4.3 Qualitative Results

The summary consists of different situations of the person in the query. For example, with captions, eyes closed, smiling, and so on. Overall, our method can generate summaries closely related to the user query with diversity.

Data Sharing Agreement

The datasets used and/or analyzed during the current study are available from the corresponding author on reaso nable request.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

References

[1] Bai, W., Noble, W.S., Bilmes, J.A.: Submodular maximization via gradient ascent: the case of deep submodular functions. Advances in Neural Information Processing Systems 2018, 7989 (2018)

[2] Benoughidene, A., Titouna, F., Boughida, A.: Static video summarization based on genetic algorithm and deep learning approach. Multimedia Tools and Applications pp. 1-26 (2024)

[3] Bilmes, J., Bai, W.: Deep submodular functions. arXiv preprint arXiv:1701.08939 (2017)

[4] Calinescu, G., Chekuri, C., Pal, M., Vondrák, J.: Maximizing a monotone submodular function subject to amatroid constraint. SIAM Journal on Computing 40(6), 1740-1766 (2011)

[5] De Avila, S.E.F., Lopes, A.P.B., da Luz Jr, A., de Albuquerque Araújo, A.: Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. Pattern Recognition Letters 32(1), 56-68 (2011)

[6] Dhanushree, M., Priya, R., Aruna, P., Bhavani, R.: Static video summarization with multi-objective constrained optimization. Journal of Ambient Intelligence and Humanized Computing 15(4), 2621-2639 (2024)

[7] Gong, B., Chao, W.L., Grauman, K., Sha, F.: Diverse sequential subset selection for supervised video summarization. Advances in Neural Information Processing Systems 27, 2069-2077 (2014)

[8] Gupta, A., Roth, A., Schoenebeck, G., Talwar, K.: Constrained non-monotone submodular maximization: Offline and secretary algorithms. In: International Work-shop on Internet and Network Economics. pp. 246-257. Springer (2010)

[9] Gygli, M., Grabner, H., Van Gool, L.: Video summarization by learning submodular mixtures of objectives. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3090-3098 (2015)
[10] Huang, J.H., Worring, M.: Query-controllable video summarization. In: Proceedings of the International Conference on Multimedia Retrieval. pp. 242-250 (2020)
[11] Jia, M., Wei, Y., Song, X., Sun, T., Zhang, M., Nie, L.: Query-oriented micro-video summarization. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
[12] Jiang, P., Han, Y.: Hierarchical variational network for user-diversified & query-focused video summarization. In: Proceedings of the International Conference on Multimedia Retrieval. pp. 202-206 (2019)
[13] Khurana, K., Deshpande, U.: Two stream multi-layer convolutional network for keyframe-based video summarization. Multimedia Tools and Applications 82(25), 38467-38508 (2023)
[14] Krause, A., Golovin, D.: Submodular function maximization. Tractability 3, 71-104 (2014)
[15] Kulesza, A., Taskar, B.: Determinantal point processes for machine learning. arXiv preprint arXiv:1207.6083 (2012)
[16] Kumari, L., Bilmes, J.: Submodular span, with applications to conditional data summarization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 12344-12352 (2021)
[17] Lee, Y.J., Ghosh, J., Grauman, K.: Discovering important people and objects for egocentric video summarization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1346-1353. IEEE (2012)
[18] Mirzasoleiman, B., Badanidiyuru, A., Karbasi, A.: Fast constrained submodular maximization: Personalized data summarization. In: International Conference on Machine Learning. pp. 1358-1367. PMLR (2016)
[19] Mirzasoleiman, B., Jegelka, S., Krause, A.: Streaming non-monotone submodular maximization: Personalized video summarization on the fly. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
[20] Sharghi, A., Gong, B., Shah, M.: Query-focused extractive video summarization. In: Proceedings of the European Conference on Computer Vision. pp. 3-19. Springer (2016)
[21] Sharghi, A., Laurel, J.S., Gong, B.: Query-focused video summarization: Dataset, evaluation, and a memory network based approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4788-4797 (2017)
[22] Tschiatschek, S., Iyer, R.K., Wei, H., Bilmes, J.A.: Learning mixtures of submodular functions for image collection summarization. In: Advances in Neural Information Processing Systems. pp. 1413-1421 (2014)
[23] Vasudevan, A. B., Gygli, M., Volokitin, A., Van Gool, L.: Query-adaptive video summarization via quality-aware relevance estimation. In: Proceedings of the ACM International Conference on Multimedia. pp. 582-590 (2017)
[24] Wei, H., Ni, B., Yan, Y., Yu, H., Yang, X., Yao, C.: Video summarization via semantic attended networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
[25] Wu, G., Lin, J., Silva, C.T.: Intentvizor: Towards generic query guided interactive video summarization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10503-10512 (2022)
[26] Xiao, S., Zhao, Z., Zhang, Z., Guan, Z., Cai, D.: Query-biased self-attentive network for query-focused video summarization. IEEE Transactions on Image Processing 29, 5889-5899 (2020)
[27] Yeung, S., Fathi, A., Fei-Fei, L.: Videoset: Video summary evaluation through text. arXiv preprint arXiv:1406.5824 (2014)
[28] Buchbinder, N., Feldman, M., Seffi, J., Schwartz, R.: A tight linear time (1/2)-approximation for unconstrained submodular maximization. SIAM Journal on Computing 44(5), 1384-1402 (2015).

Supplementary Material of
Semantic-Conditional Network for Micro-Video Summarization

1. Learning

The parameters in the network are learned using annotated summary frames to capture the underlying frame selection criteria. Unlike the existing methods of conditional video summarization, our training requires only one stage and is more computationally efficient. We employ the modern DNN training mechanism to train our submodular network. The parameter θ includes the weights A, α, and β mentioned in Subsection 2.6.1, which can be computed as follows:

$$\theta^* = \arg\min_{\theta \geq 0} \sum_c \sum_j L_{cj}(\theta) + \|\theta\|_1, \tag{1}$$

where $c = 1, 2, \ldots, N_t$ is the N_t videos for training, $j = 1, 2, \ldots, N$ is the N training subsets selected from a training video, and $L_{cj}(\cdot)$ is the loss function. The $l1$-norm of the parameter, $\|\theta\|_1$, is to keep its value small and sparse. And θ remains non-negative to preserve submodularity. Although multiple subsets can be used for training, we found using only one training subset can achieve fairly good results. The training subset $\bar{S}_0 \in \mathbb{R}_+^V$ is a vector with all coordinates equal to $\frac{k}{n}$, where k is the cardinality constraint, and n is the total number of video frames.

For the loss function $L_{cj}(\cdot)$, we adopt $l1$ loss with the formula shown below:

$$L_{cj}(\theta) = \left\| \bar{F}_{cj} - \bar{F}_{cgt} \right\|, \tag{2}$$

where $\bar{F}_{cj} = \bar{F}\left(X_{V_c}, \bar{S}_{cj}, \theta\right)$, $\bar{F}_{cgt} = \bar{F}\left(X_{V_c}, \bar{S}_{cgt}, \theta\right)$, and $\bar{S}_{cgt}$ is the annotated user summary. The user query vector $\bar{Q}_V$ is set to 1 for each dimension. That is to say, all the video frames are targeted, which is equivalent to generic video summarization. Other loss functions, like mean squared error (MSE) loss, are also applicable. The intuition of the training mechanism is to narrow the distance between the training subset and the ground truth summary set, which makes the optimal subset easier to be discovered in the subsequent optimization stage and expedites the optimization process.

We leverage the projected subgradient gradient descent (SGD) to learn and optimize the objective function in Subsection 2.6.1. After each training step, we project the parameters to $\theta \geq 0$, thus keeping the function non-negative. The rectified linear unit (ReLU) $g(\cdot) = \max(0, \cdot)$ can be used for the projection, which changes the Equation (3) as below:

$$\theta^* = \arg\min \mathrm{ReLU}\left(\sum_c \sum_j L_{cj}(\theta) + \|\theta\|_1\right). \tag{3}$$

2. Explanation of the Overall Algorithm

In this section, we explain why there are two iterations in the overall algorithm in Subsection 2.6.3. For efficiency, a monotone optimization method is utilized to maximize non-monotone submodular functions. The optimal result for monotone submodular function maximization often satisfies $\bar{F}(\bar{S}_1) \geq \gamma \bar{F}(\bar{S}_1 \cup C^*)$, where $0 < \gamma \leq 1$, and $C^* = \mathrm{OPT}$ is the optimal result. When $\bar{F}$ is monotone, $\bar{F}(\bar{S}_1 \cup C^*) \geq F(C^*)$, and we reach the approximation outcome $\bar{F}(\bar{S}_1) \geq \gamma \mathrm{OPT}$. In the non-monotone case, we cannot get the optimization result. So we consider two occasions. If $\bar{F}(\bar{S}_1 \cap C^*) \geq \epsilon \mathrm{OPT}$, the current subset $\bar{S}_1$ includes a reasonable fraction of the optimal result, and we run the local search algorithm LocalRounding to get the final result. Otherwise, if $\bar{F}(\bar{S}_1 \cap C^*) \leq \epsilon \mathrm{OPT}$, running another round of approximation within $1 - \bar{S}_1$ will find a good solution. That is why we need two iterations in the main procedure.

3. Proof of time complexity

According to the explanation in Appendix 2, the monotone submodular function $\bar{F}$ provides $\bar{F}(\bar{S}_1) \geq \alpha \bar{F}(\bar{S}_1 \cup C^*)$, where $\bar{S}_1$ is the selected subset, $0 < \alpha \leq 1$ is the approximation error, and $C^* = \mathrm{OPT}$ is the

optimal result. In the non-monotone case, the solution $\bar{S}_1'$ of the LocalRounding satisfies $\bar{F}(\bar{S}_1') \geq \epsilon \bar{F}(\bar{S}_1' \cap C^*)$, where ϵ is the approximation error of the LocalRounding.

According to the lemma from [1], we get $\mathbb{E}[f(S \cup C^*)] \geq \left(1 - \frac{1}{k}\right) f(C^*)$, where k is the cardinality constraint. Then, we can write $f(S) \geq \frac{k-1}{k\left(\frac{1}{\alpha}-\frac{1}{\epsilon}\right)} f(C^*)$. In our experiment, the value of the ratio $\frac{k-1}{k}$ is close to 1. Following research [1], the approximation for a deterministic unconstrainted submodular maximization is $\frac{1}{3}$. Using $\epsilon = \frac{1}{3}$, we get the desired result $f(S) \geq \frac{1}{3+1/\alpha} f(C^*)$.

Now we verify the time complexity. Research [1] proposed a deterministic unconstrained local search method for submodular maximization, which requires linear time to complete. We expand the result to satisfy the partition matroid constraint. The partition matroid can be taken as uniform matroids in the set V_1, the segment with higher similarity to the user query. When the cardinality of the uniform matroid constraint is k, the task is transformed to find out the k most similar elements. Linear interpolation can do this in linear time.