SummDiff: Generative Modeling of Video Summarization with Diffusion

Kwanseok Kim[1]*, Jaehoon Hahm[2]*, Jinhwan Sul[3], Byunghak Kim[4], Joonseok Lee[1]†
[1]Seoul National University, [2]UIUC, [3]Georgia Institute of Technology, [4]Hyundai Card
kjvd1009@snu.ac.kr, jh141@illinois.edu,
jsul7@gatech.edu, byunghak.kim@hcs.com, joonseok@snu.ac.kr

Abstract

Video summarization is a task of shortening a video by choosing a subset of frames while preserving its essential moments. Despite the innate subjectivity of the task, previous works have deterministically regressed to an averaged frame score over multiple raters, ignoring the inherent subjectivity of what constitutes a "good" summary. We propose a novel problem formulation by framing video summarization as a conditional generation task, allowing a model to learn the distribution of good summaries and to generate multiple plausible summaries that better reflect varying human perspectives. Adopting diffusion models for the first time in video summarization, our proposed method, SummDiff, dynamically adapts to visual contexts and generates multiple candidate summaries conditioned on the input video. Extensive experiments demonstrate that SummDiff not only achieves the state-of-the-art performance on various benchmarks but also produces summaries that closely align with individual annotator preferences. Moreover, we provide a deeper insight with novel metrics from an analysis of the knapsack, which is an important last step of generating summaries but has been overlooked in evaluation.

1. Introduction

Recently, short-form videos draw significant attention on video sharing platforms, with a trend that consumers increasingly prefer to quickly grasp the content. They often compressively convey contents that are originally in a longer form, summarizing the core contents into a shorter one; *e.g.*, sport games highlights or movie summarization. This task of selecting core parts of a long video to construct a shorter one is called *video summarization*. This task is inherently subjective, since there can be multiple criteria for a 'good summary'; *e.g.*, comprehensively covering the entire storyline or subjectively selecting impressive parts of the video (highlight detection). Due to this inherent subjectivity, most video summarization or highlight detection datasets [20, 71] offer annotations by multiple raters to reflect various perspectives.

Since each annotator may have different opinion on the importance of a frame, most existing methods [3, 17, 22, 34, 67, 82] take the frame-level importance scores averaged across multiple annotators as their target label, and are trained to predict them. This frame-level score aggregation looks reasonable in some sense, but in fact it loses the various perspectives to summarize each video. For instance, suppose half of the annotators select clips from the first quarter of the video while the other half select clips from the last quarter. If one simply averages their frame scores, both the first and last quarters end up with similar importance, obscuring the two distinct valid summaries. That is, this simple regression to the averaged frame-level importance scores fail to preserve multiple viewpoints to summarize the video.

In order to preserve and reflect various views to summarize a video, we pose a *distribution* of its good summaries and let the model to learn it, instead of giving an already-aggregated single ground truth importance score. Then, the video summarization task can be seen from a generative perspective; *i.e.*, a process of learning and inferring the distribution of good summaries conditioned on the input video. Specifically, the model is now in charge of estimating the distribution of plausible summaries for the given video. Once trained, it allows us to sample multiple summaries from the estimated conditional data manifold. To the best of our knowledge, this generative approach to the video summarization has not been extensively studied, except for a few works [2, 50] that applied adversarial losses to construct a summary that looks like the original video.

Formulating the video summarization problem as a conditional generation task, we propose to adopt the generative diffusion [25, 72] mechanism, which has been successfully applied to various conditional generation tasks [24, 54, 61]. Specifically, conditioned on the input video, our proposed **SummDiff** model learns to denoise a random importance

*Equal contribution
†Corresponding author

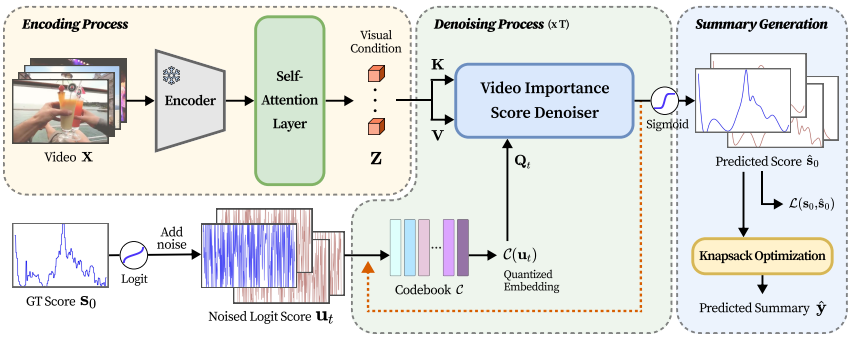


Figure 1. **Overview of SummDiff.** Given an input video, SummDiff generates importance scores conditioned on video frames. T denotes the number of DDIM steps.

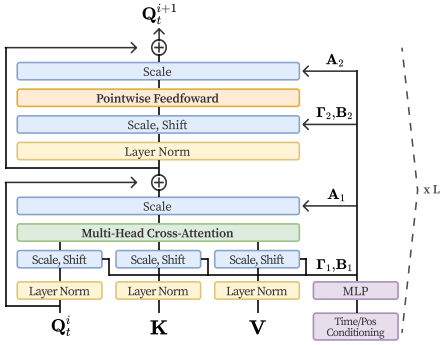


Figure 2. **Video Importance Score Denoiser**. We use AdaLN layer to inject time and positional conditions, following [58].

score vector over the given video into an importance score vector sampled from the distribution that corresponds to a good summary of the video. In contrast to the previous deterministic methods, our approach allows to sample multiple plausible summaries for a given video starting from a different random noise, better aligned with the subjective nature of summarization where we usually have various true labels reflecting multiple views.

Extensive experiments demonstrate that our model outperforms existing baselines across multiple datasets. Also, we revisit the knapsack, a relatively unexplored step in the summarization evaluation in spite of its nontrivial impact on the performance, and propose additional novel metrics based on this analysis to provide deeper insights.

Our contributions can be summarized as follows:

- We propose a novel generative viewpoint of video summarization, better suited for the subjective nature of the task, allowing multiple plausible summaries for a video.
- We innovatively apply diffusion to the video summarization for the first time, integrating learning the distribution of good summaries into the model.
- We analyze the knapsack optimization process and propose additional metrics to quantify the optimality of the predicted importance scores.

2. Problem Formulation

Given a video of N frames, the objective of video summarization is to identify and select $S < N$ frames that effectively encapsulate the essence of the video content. Let $\mathbf{X} \equiv \{\mathbf{X}_i \in \mathbb{R}^{H\times W\times 3} : i = 0, ..., N-1\} \in \mathbb{R}^{N\times H\times W\times 3}$ be a video, where H, W denote the size of the frames.

A video summary, $\mathbf{y} \equiv \{y_i \in \{0,1\} : i = 0, ..., N-1, \sum_i y_i \leq S\} \in \{0,1\}^N$, indicates inclusion (1) or exclusion (0) of each frame. When a video is provided with multiple ground truth annotations, we denote each individual score by $\mathbf{s}^{(r)} \in [0,1]^N$ and the corresponding binary summary by $\mathbf{y}^{(r)} \in \{0,1\}^N$ which is obtained following the procedure explained in Sec. 3.3. The predicted summary is denoted by $\hat{\mathbf{y}} \equiv \{\hat{y}_i \in \mathbb{R} : i = 0, ..., N-1\} \in \mathbb{R}^N$.

Previous models [3, 17, 67] have approached video summarization as a regression task, aiming to predict the importance score $\mathbf{s} \in [0,1]^N$ for each video, often the average of multiple importance scores, $\frac{1}{|R|}\sum_{r\in\mathcal{R}} \mathbf{s}^{(r)}$. In contrast, our approach allows multiple summaries for each video, aiming to learn the distribution of its plausible summaries. The previous scheme can be seen as a special case of ours, where every video has a single golden way of summarization, pointed at $\mathbf{s}$ with zero variance. Under our extended setting, multiple importance scores $\{\mathbf{s}^{(r)} | r \in \mathcal{R}\}$ can be given to the same conditioning video $\mathbf{X}$, forming a probability distribution of plausible importance scores.

3. Diffusion-based Video Summarization

Posing the video summarization as a conditional generation task, we introduce our SummDiff model, designed to adapt the distribution of individual importance scores for a given video by learning to denoise.

3.1. Overall Flow of the Proposed Method

Fig. 1 illustrates the overall flow of our SummDiff model.

Encoding Process. We first encode each frame $\mathbf{X}_i$ for $i = 1, ..., N$ from the input video $\mathbf{X}$ using a pre-trained image encoder. The extracted features are further contextualized through self-attention [80], as seen in Fig. 1. We denote the encoded feature for each individual frame by $\mathbf{z}_i \in \mathbb{R}^D$, and collectively the entire feature matrix by $\mathbf{Z} \in \mathbb{R}^{N\times D}$.

Denoising Process. We then learn to denoise an individual importance score vector from a random noise, conditioned on the visual embeddings. First, the *forward process* adds noise to the ground truth individual importance score $\mathbf{s}_0 \equiv \mathbf{s}^{(r)}$ and sample a noised one $\mathbf{s}_t = \sqrt{\bar{\alpha}_t}\mathbf{s}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t$, where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$, $\bar{\alpha}_t = \prod_{\tau=1}^{t}(1-\beta_\tau)$, and t is the diffusion time step. The perturbation kernel at t, defined as $q(\mathbf{s}_t|\mathbf{s}_{t-1}) = \mathcal{N}(\mathbf{s}_t; \sqrt{1-\beta_t}\mathbf{s}_{t-1}, \beta_t\mathbf{I})$, where β_t is defined by the variance schedule. Since $\mathbf{s}_0$ is bounded within $[0,1]$, it is not straightforward to add Gaussian noise directly to it so we first transform it to its logit, $\mathbf{u}_0 = \log\frac{\mathbf{s}_0}{1-\mathbf{s}_0} \in$

$\mathbb{R}^N$, and perform the noising process in this logit space. For numerical stability, we clip $\mathbf{s}_0$ to $[\epsilon, 1-\epsilon]$, where ϵ is a small constant.

Then, the *reverse process* progressively removes noise from $\mathbf{s}_t$ to $\mathbf{s}_0$, formulated by $p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t) = \mathcal{N}(\mathbf{s}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{s}_t, t), \sigma_t^2\mathbf{I})$, where σ_t^2 is determined in relation to β_t, and the posterior mean $\boldsymbol{\mu}_\theta(\mathbf{s}_t, t)$ is modeled with a trainable neural network. In this paper, we extend this estimator of posterior mean to be conditioned on a video, similarly to the conditional generation on diffusion models.

Once trained, our denoiser is able to recover a plausible importance score for a video from a random noise, which is used to generate a summary (Sec. 3.3). Repeated generation across the noise distribution would converge to the true distribution of plausible scores for the video.

3.2. Video Importance Score Denoiser

We formulate the key component of our method, the video importance score denoiser in Fig. 2. Starting from a noised logit score vector $\mathbf{u}_t \in \mathbb{R}^N$, it predicts a plausible importance score, conditioned on the given video. Denoted by $f_\theta(\mathbf{u}_t, t, \mathbf{Z})$, it learns to denoise the given score vector $\mathbf{u}_t$ at the diffusion time step t under the visual condition $\mathbf{Z}$, producing the denoised score $\hat{\mathbf{u}}_{t-1}$ by one time step, where θ is the set of its learnable parameters.

Transformer-based Diffusion. We denoise the logit-transformed score $\mathbf{u}_t$ given t and $\mathbf{Z}$, predicting $\hat{\mathbf{u}}_{t-1}$, using a transformer-based cross-attention [80]. The logit-transformed score $\mathbf{u}_t$ acts as the query, and the visual condition $\mathbf{Z}$ is used as key/value. This setup allows the model to effectively denoise $\mathbf{u}_t$ conditioned on the information from the input video, ensuring consistency between the predicted logit score $\hat{\mathbf{u}}_{t-1}$ and the condition $\mathbf{Z}$.

To apply dot-product cross-attention, the dimensionality should match for queries and keys. To this end, we *quantize* the importance scores into a predefined number (K) of uniform segments. Specifically, we convert the noised logit $\mathbf{u}_t$ back to its original bounded range $[0, 1]$ by applying sigmoid $1/(1+e^{-\mathbf{u}_t})$, and divide them into K equally-binned segments. Each score range is associated with a learnable embedding of size D (codebook in Fig. 1). Based on the codebook $\mathcal{C}$, we map the scores $\mathbf{u}_t \in \mathbb{R}^N$ to their corresponding quantized embedding $\mathcal{C}(\mathbf{u}_t) \in \mathbb{R}^{N\times D}$. Our denoiser $f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z})$ takes this $\mathcal{C}(\mathbf{u}_t)$ as input, instead of $\mathbf{u}_t$. See Sec. 4.5 for anaylsis on the size of codebook.

Query Formation in Cross-Attention. Following [25], the time step t is embedded as $\boldsymbol{\tau} \in \mathbb{R}^D$ using sinusoidal functions and an MLP. Considering the sequential nature of videos, we also introduce temporal positional embeddings $\boldsymbol{\Phi} \in \mathbb{R}^{N\times D}$ using sinusoidal functions.

The simplest way to set the query in the cross-attention would be to add all inputs except for the condition $\mathbf{Z}$; that is, $\mathbf{Q}_t = \mathcal{C}(\mathbf{u}_t) + \boldsymbol{\tau} + \boldsymbol{\Phi}$. However, we leverage AdaLN-Zero block [58] to integrate the time embedding $\boldsymbol{\tau}$ and temporal positional encoding $\boldsymbol{\Phi}$ more effectively by separating them from the query $\mathbf{Q}_t$, preventing information mixing. Hence, the query becomes $\mathbf{Q}_t = \mathcal{C}(\mathbf{u}_t)$ and $\boldsymbol{\tau}, \boldsymbol{\Phi}$ are conditioned via scale-shift operation. See Sec. 4.5 for ablation studies.

We conduct cross-attention with $\mathbf{Q}_t$ as queries and visual condtions $\mathbf{Z} \in \mathbb{R}^{N\times D}$ as keys $\mathbf{K}$ and values $\mathbf{V}$. We take an MLP from the AdaLN output to regress the scale and shift parameters, denoted by $\mathbf{A}_1$, $\mathbf{B}_1$, $\boldsymbol{\Gamma}_1$, $\mathbf{A}_2$, $\mathbf{B}_2$, and $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{N\times D}$. As depicted in Fig. 2, they scale and shift $\mathbf{Q}_t$, $\mathbf{K}$, and $\mathbf{V}$. Then, they are passed through cross-attention with skip connections and a subsequent rescaling. Formally,

$$
\begin{aligned}
\mathbf{H}^{i'} &= \boldsymbol{\Gamma}_1 \odot \mathbf{H}^i + \mathbf{H}^i + \mathbf{B}_1 \\
\mathbf{X}_1 &= \mathbf{A}_1 \odot \text{softmax}(\mathbf{Q}_t^{i'}\mathbf{K}^{i'\top})\mathbf{V}^{i'} + \mathbf{Q}_t^{i'} \\
\mathbf{X}_2 &= \boldsymbol{\Gamma}_2 \odot \mathbf{X}_1 + \mathbf{X}_1 + \mathbf{B}_2 \\
\mathbf{Q}_t^{i+1} &= \mathbf{A}_2 \odot \text{MLP}(\mathbf{X}_2) + \mathbf{X}_2,
\end{aligned}
$$

where $\mathbf{H}^{i^{(\prime)}} \in \{\mathbf{Q}_t^{i^{(\prime)}}, \mathbf{K}^{i^{(\prime)}}, \mathbf{V}^{i^{(\prime)}}\}$ denotes the matrices used for attention, $\odot$ is the (broadcasted) Hadamard product, and $i = 1, ..., L$ is the layer index.

Training. We train the denoiser f_θ to estimate the true importance score $\hat{\mathbf{s}}_0$ after acting upon a fully-connected layer and a sigmoid function σ. We minimize the following loss [12] on each annotator's individual importance score:

$$
\mathcal{L}(\mathbf{s}_0, \hat{\mathbf{s}}_0) = \|\mathbf{s}_0 - \hat{\mathbf{s}}_0\|_2^2 = \|\mathbf{s}_0 - \sigma(\text{FC}(f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z})))\|_2^2.
$$

The learnable embeddings in the codebook $\mathcal{C}$ are compositely optimized, finding an effective representation for $\mathbf{u}_t$ during training.

Inference. Our model generates a logit-transformed importance score from a random noise $\mathbf{u}_T \sim \mathcal{N}(0, \mathbf{I})$ for a given video. Employing the reverse diffusion process [68], it iteratively refines the logit score towards cleaner estimations:

$$
\begin{aligned}
\hat{\mathbf{u}}_{t-1} = \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\,\frac{\hat{\mathbf{u}}_t - \sqrt{\bar{\alpha}_t} f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}))}{\sqrt{1-\bar{\alpha}_t}} \\
+ \sqrt{\bar{\alpha}_{t-1}} f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z})) + \sigma_t \epsilon_t \in \mathbb{R}^N,
\end{aligned}
$$

where $\sigma_t = \sqrt{(1-\bar{\alpha}_{t-1})/(1-\bar{\alpha}_t)}\sqrt{1-\bar{\alpha}_t/\bar{\alpha}_{t-1}}$ (DDPM reverse process). When $t = 1$ at the last step, $\hat{\mathbf{u}}_0 = f_\theta(\mathcal{C}(\mathbf{u}_1), 1, \mathbf{Z})$ is directly used to remove stochasticity at the end of inference process. Finally, we take sigmoid to $\hat{\mathbf{u}}_0$ to derive the final importance score $\hat{\mathbf{s}}_0 \in (0, 1)^N$.

3.3. Summary Generation

From the raw importance score $\hat{\mathbf{s}}_0$, we apply the standard knapsack-based approach to decide which frames to be included in the final summary. To make this summary video more realistic, it is common to choose at a semantic clip

level instead of individual frame level. We adopt a widely-used Kernel Temporal Segmentation (KTS) [59, 84] to partition a video into disjoint temporal intervals. We take the average score among the corresponding frames as the clip-level importance; *i.e.*, $v_i = \sum_{j=t_i}^{t_{i+1}} \hat{\mathbf{s}}_{0,j}/(t_{i+1} - t_i)$ for the i-th clip composed of frames from t_i to t_{i+1}. Subsequently, we select the clips based on v_i by solving the binary knapsack problem (KP) [23] with dynamic programming [71]:

$$\mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho) \equiv \underset{\boldsymbol{s}\in\{0,1\}^M}{\operatorname{argmax}} \sum_{i=1}^{M} v_i s_i \quad \text{s.t.} \sum_{i=1}^{M} w_i s_i \leq \rho N,$$

where $\boldsymbol{v}, \boldsymbol{w}, \rho$ denotes the values of each clip ($v_i \in [0, 1]$), costs of selecting each clip, and the budget constraint ratio (*e.g.*, $\rho = 0.15$; 15% of the video length), respectively. If the argmax has multiple solutions, we will abuse the notation and denote $\mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho)$ as an arbitrary element of the solution set.

4. Experiments

4.1. Experimental Settings

Datasets. We evaluate our method on three benchmarks. TVSum [71] and SumMe [20] are traditional datasets, containing 50 and 25 videos respectively and manually labeled annotations by up to 20 raters. Mr. HiSum [74] is a large-scale dataset, composed of 31,892 videos and importance scores derived from YouTube Most Replayed statistics. It provides an importance score $\mathbf{s}_0$ and a corresponding summary $\mathbf{y}_0$ for each video, averaged over 50,000+ viewers. We use the default features for each dataset, GoogLeNet [75] for TVSum and SumMe, and Inception-v3 [76] PCA-ed to 1024D [1] for Mr. HiSum.

We adopt two dataset splits. First, we randomly split TVSum and SumMe into 7:1:2 for train, validation, and test set, and report averaged test scores for 5 random splits (namely, TVT). Also, following prior works [67, 77], we include five-fold cross-validation results (5FCV), although these may suffer from overfitting to the test set due to the lack of explicit validation set, and thus may not generalize well on unseen videos. For Mr. HiSum, we use its original train/validation/test split. We run all experiments 5 times.

Evaluation Metrics. Following recent practice [22, 40, 55, 67], we report rank order statistics, Kendall's τ [37] and Spearman's ρ [96], to asses how well the model ranks frame importance. F1 score had been widely used in video summarization, but recent literature [67, 77] reports that it is excessively sensitive to video segmentation and unreasonably favors summaries composed of many short shots [55], while disregarding longer key shots [67]. As an alternative to F1, we propose additional metrics that represent the influence of importance score through an analysis of the knapsack algorithm in Sec. 4.4.

Method	SumMe		TVSum	
	τ	ρ	τ	ρ
Random	0.000	0.000	0.000	0.000
Human	0.205	0.213	0.177	0.204
A2Summ [22]	0.088	0.096	0.157	0.206
VASNet [17]	0.089	0.099	0.153	0.205
PGL-SUM [3]	0.104	0.116	0.141	0.186
CSTA [67]	0.108	0.120	0.168	0.221
SummDiff (Ours)	**0.133**	**0.148**	**0.173**	**0.226**

Table 1. **Comparison of models trained under TVT (train/val/test split)** on SumMe [20] and TVSum [71]. Best and second-best results are **boldfaced** and underlined, respectively.

Method	SumMe		TVSum	
	τ	ρ	τ	ρ
Random	0.000	0.000	0.000	0.000
Human	0.205	0.213	0.177	0.204
DSNet-AF [93]	0.037	0.046	0.113	0.138
DSNet-AB [93]	0.051	0.059	0.108	0.129
SUM-GAN [50]	0.049	0.066	0.024	0.031
AC-SUM-GAN [2]	0.102	0.088	0.031	0.041
CLIP-It [52]	-	-	0.108	0.147
iPTNet [34]	0.101	0.119	0.134	0.163
A2Summ [22]	0.108	0.129	0.137	0.165
VASNet [17]	0.160	0.170	0.160	0.170
PGL-SUM [3]	-	-	0.157	0.206
AAAM [77]	-	-	0.169	0.223
MAAM [77]	-	-	0.179	0.236
VSS-Net [85]	-	-	0.190	0.249
DMASum [81]	0.063	0.089	**0.203**	**0.267**
SSPVS [40]	0.192	0.257	0.181	0.238
CSTA [67]	0.246	0.274	0.194	0.255
SummDiff (Ours)	**0.256**	**0.285**	0.195	0.255

Table 2. **Comparison of models trained under 5FCV (5-Fold Cross Validation)** on SumMe [20] and TVSum [71]. Training under the 5FCV setting tends to overfit the test set.

Model	$\tau \uparrow$	$\rho \uparrow$	MAP$_{\rho=50\%} \uparrow$	MAP$_{\rho=15\%} \uparrow$
SUM-GAN [50]	0.067	0.095	56.62	23.56
VASNet [17]	0.069	0.102	58.69	25.28
AC-SUM-GAN [2]	0.012	0.018	55.35	21.88
SL-module [82]	0.060	0.088	58.63	24.95
PGL-SUM [3]	0.097	0.141	61.60	27.45
iPTNet [34]	0.020	0.029	55.53	22.74
A2Summ [22]	0.121	0.172	63.20	32.34
CSTA [67]	0.128	0.185	63.38	30.42
SummDiff	**0.175**	**0.238**	**65.44**	**33.83**

Table 3. **Evaluation on Mr. HiSum [74].** See Appendix D for the full table including standard deviations.

We further evaluate our method on video highlight detection following [74]. First, we uniformly divide the input video into 5-second-long shots and calculate the average frame scores for each shot. The top $\rho \in \{0.15, 0.5\}$ of these shots are designated as ground truth highlights. We report Mean Average Precision (MAP), following [28, 56, 84]. See Appendix E for more implementation details.

4.2. Results and Analysis

Considering the innate subjectivity of video summarization, we train our model to learn from each individual score, allowing it to capture multiple ways of summarization for each video. Specifically, we use 20 individual importance scores per video in TVSum [71] and 15–18 individual binary summaries in SumMe [20], since SumMe does not provide individual importance scores. Tab. 1 and 2 show that SummDiff achieves the best performance on SumMe and TVSum, with the sole exception of TVSum in 5FCV. While DMA-SUM [81] performs the best on TVSum in 5FCV, it significantly underperforms on SumMe, indicating that its high performance on TVSum does not generalize well. Comparing TVT and 5FCV, most models exhibit a considerable performance gap. This demonstrates how much existing models have overfitted to the test set under the conventional 5FCV protocol. We adopt TVT as a remedy to this, but due to the extremely small size of these datasets, the reliability of evaluation is still limited [74].

On the larger Mr. HiSum, we train our model using the aggregated annotation, as it does not provide individual annotations. This evaluation mainly aims to verify scalability of SummDiff on a large-scale dataset. Tab. 3 presents the performance on Mr. HiSum, where SummDiff consistently outperforms all baselines across all metrics, significantly surpassing the strongest competitor, CSTA [67] with a large margin. These results highlight scalability and effectiveness of our method even under the single-label setting.

4.3. Qualitative Demonstration

We demonstrate how our method discovers a variety of possible summaries for a video. Specifically, we generate video summaries for the 50 videos in TVSum using CSTA, PGL-SUM, VASNet, and ours. For each generated summary, we measure the Kendall's τ with all 20 ground truth annotations, respectively. Considering $\tau \geq 0.25$ as the threshold for the summary to be matched with the annotator, Fig. 3 shows if the generated summary for a video (row) matches with each annotator (column). As the baselines (CSTA, PGL-SUM, and VASNet) summarize a video deterministically, each cell is either completely matched (1) or not (0). If the summary matches with majority of annotators (*e.g.*, video 29 for CSTA), it means the deterministic summary matches with the dominant way of summarization for the video. If it matches only with a few annotators (*e.g.*, video 24 for CSTA), it means the generated summary matches with a minor opinion. For our method, we generate 100 summaries per video from different Gaussian noise vectors, and mark each cell with the ratio of summaries that match with each annotator. The darker the cell colors are for each row, the more various ways of summaries have been discovered by our method. Comparing the heat maps, we clearly observe that SummDiff generates significantly more various

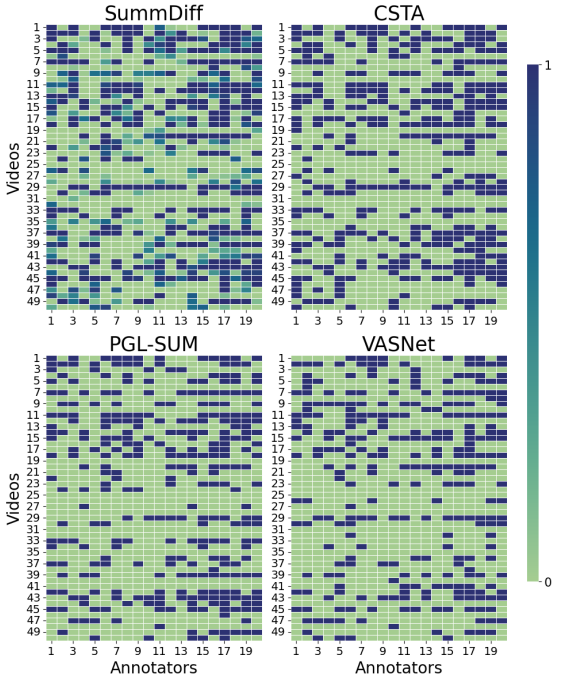


Figure 3. **Ratio of summaries with $\tau \geq 0.25$ for each video-annotator pair.** The heatmap illustrates how closely each method's summary matches with individual annotations. This comparison reveals the extent to which each method captures the varied human summaries. SummDiff covers a larger area of the heatmap, which indicates better coverage over various summaries.

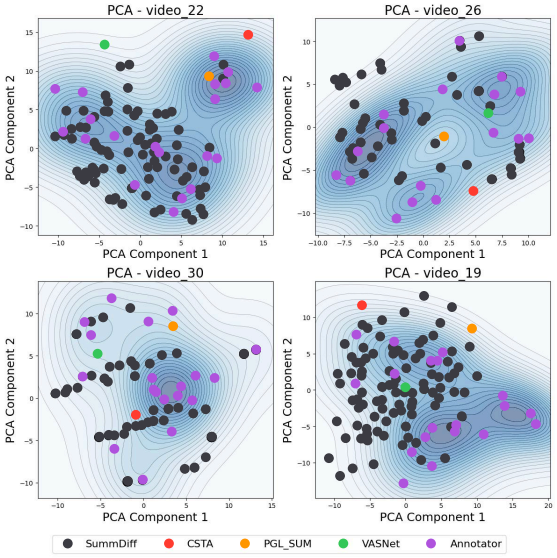


Figure 4. **Distribution of true and predicted summaries of given videos.** SummDiff can generate various summaries and cover the targeted distribution of summaries, while baselines deterministically predicts a single summary.

summaries covering multiple viewpoints, better reflecting the distribution of annotated summaries.

We further illustrate the variation and quality of generated summaries in Fig. 4. The summaries annotated by human raters (● Annotator) and those generated by ours (●

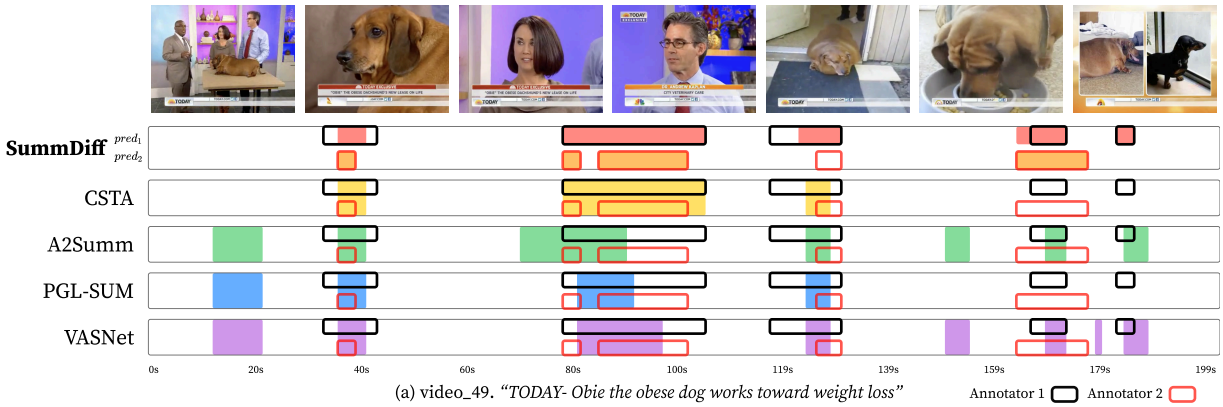


Figure 5. **Demonstration of video summaries generated by competing methods on a TVSum video**. Shaded parts indicate the segments selected by each method, and the two rows of edged boxes within each method indicate the two distinct true annotations. The results clearly demonstrate the effectiveness of SummDiff in capturing multiple plausible summaries for a video. See another example in Appendix F.

SummDiff) and baselines (• CSTA, • PGL-SUM, and • VASNet) are projected onto the first two principal components, which are computed using only the human-annotated summaries. The distribution of true summaries are displayed using a contour map, obtained by Gaussian kernel density estimation. It demonstrates that our SummDiff generates multiple summaries closely aligned with the distribution of true annotations. For example, video 22 (top-left) roughly has three ways to summarize, according to the human raters. SummDiff generates summaries dominantly covering the two modes in the lower-left region, and one case in the upper-right mode. In contrast, baselines produce a single, less accurate summary, failing to account for the variety inherent in human-annotated summaries. We observe similar patterns in the other three plots as well. Fig. II provides more examples in Appendix C.

Fig. 5 illustrates example summaries generated by CSTA, A2Summ, PGL-SUM, VASNet, and our SummDiff, for a video selected from TVSum test set. Shaded parts indicate the segments selected by each method (note that SummDiff can generate two different summaries while baselines always produce a single summary), and the two rows of edged boxes within each method indicate two different true annotations. These results demonstrate that SummDiff produces more accurate and various summaries, effectively capturing multiple plausible summarizations for each video.

4.4. Metrics Inspired from Knapsack Optimality

In spite of nontrivial impact of the knapsack at the end of the summary generation, previous evaluation metrics have focused only on the accuracy of the importance scores. Through a thorough analysis on the knapsack problem (KP), we provide additional metrics that measure the contribution of importance scores. By accounting for knapsack optimality and clip-level weights, our new metrics are capable of assessing the predicted importance score more accurately than existing metrics such as F1 or ranking-based ones.

Confidence of Importance Score. First, we analyze the conditions under which the same optimal KP solution remains valid despite some perturbations to the profits $\boldsymbol{v}$. These perturbations can be a modeling to the imperfect estimation of the importance scores in video summarization.

Let $\boldsymbol{y}^*$ be the optimal solution to the original KP (*i.e.*, $\boldsymbol{y}^* = \mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho)$; see definition in Sec. 3.3) and $\Gamma \subseteq [N] \equiv \{n \in \mathbb{N} \mid 1 \leq n \leq N\}$ be an arbitrary subset of items associated with the perturbed profits $\boldsymbol{v}'_i = \boldsymbol{v}_i + \Delta v_i$, where $\Delta v_i = 0$ if $i \notin \Gamma$ and N is the number of items. In our case, $\boldsymbol{v} \in \mathbb{R}^N$ is the ground truth importance scores, $\boldsymbol{y}^* \in [0,1]^N$ is the ground truth summary, and the perturbed profit $\boldsymbol{v}' \in \mathbb{R}^N$ corresponds to the imperfectly predicted importance scores. We further define two disjoint subsets of Γ. First, $\Gamma_0^+ \equiv \{i \in \Gamma \mid y_i^* = 0, \Delta p_i > 0\}$ refers to the set of items excluded in the original optimal solution but undergoing increased profit, and $\Gamma_1^- \equiv \{i \in \Gamma \mid y_i^* = 1, \Delta p_i < 0\}$ is the opposite, the set of items included in the optimal solution but experiencing a decreased profit. Then, Hifi *et al.* [23] claims that the solution $\boldsymbol{y}^*$ remains optimal for the perturbed KP, *i.e.*, $\boldsymbol{y}^* = \mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho) = \mathcal{KP}(\boldsymbol{v}', \boldsymbol{w}, \rho)$, if

$$\sum_{i \in \Gamma_0^+} \Delta v_i - \sum_{i \in \Gamma_1^-} \Delta v_i \leq \boldsymbol{v} \cdot \mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho) - \max(\Psi^+, \Psi^-),$$

where $\Psi^+ \equiv \max_{i \in \Gamma_0^+} [\boldsymbol{v} \cdot \mathcal{KP}(\boldsymbol{v} - v_i \hat{e}_i, \boldsymbol{w}, \rho - w_i/N)]$ is the maximum value of the KP when removing any item from Γ_0^+ and adjusting the constraint, and similarly, $\Psi^- \equiv \max_{i \in \Gamma_1^-} [\boldsymbol{v} \cdot \mathcal{KP}(\boldsymbol{v} - v_i \hat{e}_i, \boldsymbol{w}, \rho)]$ is the maximum value of the KP when removing any item from Γ_1^-.

Based on this theorem we propose a new metric, **Confidence of Importance Score (CIS)**, to estimate how close the predicted scores are to the ground truth:

$$\text{CIS} = \sum_{i \in \Gamma_0^+} \Delta v_i - \sum_{i \in \Gamma_1^-} \Delta v_i - \boldsymbol{v} \cdot \mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho) + \max(\Psi^+, \Psi^-).$$

If CIS ≤ 0, the theorem indicates that the predicted importance score $\hat{\boldsymbol{v}}$ is guaranteed to induce the true KP solution $\boldsymbol{y}^*$. If CIS > 0, $\hat{\boldsymbol{v}}$ does not guarantee to induce the KP solution, but a lower CIS would indicate a solution closer to $\boldsymbol{y}^*$ than a higher one. That is, a smaller CIS indicates a stronger *confidence* in satisfying the inequality, indicating a higher chance for the solution of the perturbed KP to be identical to that of the original KP.

Unlike F1 that simply measures how close the predicted summary is to the ground truth, our CIS accounts for the *confidence* that the predicted score will exactly induce the true summary. This is particularly important in video summarization, since the importance score itself is subjective and often noisy, requiring robust selection of frames.

Weighted Inclusion Ratios. Analyzing the sensitivity of the KP optimum to the perturbation of the importance score, Belgacem and Hifi [5] established bounds of the perturbed profits to retain the original optimum. These intervals are

$$\boldsymbol{y}_i^* = 1 \rightarrow \Delta v_i \in I_i^1 = \left[\max\left(\Delta_i^-, w_i \max_{k\in\Gamma} \mu_k - p_i\right), +\infty\right)$$
$$\boldsymbol{y}_i^* = 0 \rightarrow \Delta v_i \in I_i^0 = \left(-\infty, \min\left(\Delta_i^+, p_i - w_i \min_{k\in\Gamma} \mu_k\right)\right],$$

where $\Delta_i^+ = \boldsymbol{v} \cdot (\mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho) - \mathcal{KP}(\boldsymbol{v} - v_i\hat{e}_i, \boldsymbol{w}, \rho - w_i/N), \Delta_i^- = \boldsymbol{v} \cdot (\mathcal{KP}(\boldsymbol{v} - v_i\hat{e}_i, \boldsymbol{w}, \rho) - \mathcal{KP}(\boldsymbol{v}, \boldsymbol{w}, \rho))$, and μ_k denotes the critical ratio of $k \in \Gamma$: $\mu_k = \mathcal{CR}([N] \setminus \{k\}, \rho - \mathbf{1}_{\{y_k^*=0\}} \cdot w_k/N)$, with $\mathcal{CR}(A, \gamma) = v_s/w_s$ and $s = \min\{l | \sum_{i=1}^{l} v_s/w_s > \rho\}$ is the index of the critical item. The items in A are considered in descending order of v_s/w_s. To sum up, if the perturbed profit resides in the interval $v_i' \in I_i^{\boldsymbol{y}_i^*}$, it is guaranteed that the solution of perturbed KP is identical with that of the original KP.

We propose **Inclusion Ratio (IR)** to estimate how much the predicted scores fall into the safe bounds; for the i-th importance score, $\text{IR}_i \equiv |\{\Delta v_i \in I_i^{y_i^*} \mid i \in [N]\}|/N$. This ratio reflects the proportion of the predicted scores that retain the KP solution. We propose the **Weighted average of the Inclusion Ratios (WIR)** as a metric to measure the proximity of the predicted score to the ground truth: $\text{WIR} \equiv \sum_{i=1}^{n} \left(\frac{w_i}{\sum_{j=1}^{n} w_j} \text{IR}_i\right)$, where w_i denotes the segment length of each shot.

Weighted Sum of Errors (WSE). Additionally, we compare a weighted sum of the errors $\sum_i w_i \Delta p_i$, where w_i is the segment length of each clip.

Comparison. Tab. 4 compares several competitive methods using these new metrics. Our SummDiff achieves the best performance in all metrics, indicating that the summaries generated by our method are closer to the optimal knapsack result, under the theoretical analysis in literature.

Discussion. Our newly proposed metrics, WIR and CIS, evaluate the quality of the generated importance scores in two aspects. WIR measures how many importance scores are individually trustworthy, weighted by the duration. Specifically, WIR first computes the safe interval I_i for each importance score such that the true summary remains unchanged. Then, given the predicted importance scores $\hat{\mathbf{v}}$, it measures how many of them fall within their corresponding safe intervals $\hat{v}_i \in I_i$.

On the other hand, CIS quantifies the risk that the predicted importance scores will lead to a different summary than the one generated using the true importance scores. In other words, CIS evaluates the overall chance how likely the predicted score vector $\hat{\mathbf{v}}$, as a whole, is to give a summary different from the true one. With the analysis of knapsack optimality, these metrics would better measure the accuracy of predicted importance score, considering the fidelity of final summary *after* knapsack to the ground truth, than F1 or ranking-based metrics.

Method	$\tau \uparrow$	$\rho \uparrow$	CIS $\downarrow$	WIR $\uparrow$	WSE $\downarrow$
Uniform-Random	0.000	0.000	9.27	0.45	46.82
SL-module [82]	0.060	0.088	6.83	0.56	30.09
CSTA [67]	0.128	0.185	6.23	0.57	25.76
PGL-SUM [3]	0.097	0.141	6.14	0.58	26.22
VASNet [17]	0.069	0.102	6.25	0.59	26.79
A2Summ [22]	0.121	0.172	6.65	0.55	30.55
SummDiff	**0.175**	**0.238**	**5.96**	**0.61**	**25.24**

Table 4. Evaluation using our new metrics proposed in Sec. 4.4.

Module	$\tau \uparrow$	$\rho \uparrow$
Encoder Only	0.071	0.104
(+) Video Importance Denoiser	0.079	0.116
(+) Learnable Embedding	0.086	0.124
(+) AdaLN ($\boldsymbol{\tau}$)	0.145	0.204
(+) AdaLN ($\boldsymbol{\tau}$, $\boldsymbol{\Phi}$) 1 layer	0.171	0.232
(-) Quantization	0.125	0.174
(+) Classifier-free Guidance (CFG) [24]	0.175	0.238
(+) Self-attention Guidance (SAG) [30]	0.177	0.239

Table 5. Effect of individual components

4.5. Ablation Study

We report ablation study on Mr. HiSum [74] in Tab. 5, comparing performance adding a component step-by-step. Starting from a two-layer transformer encoder predicting the importance score, adding our denoiser with a naively summed query $\mathbf{Q}_t = \mathcal{C}(\mathbf{u}_t) + \boldsymbol{\tau} + \boldsymbol{\Phi}$ improves the Kendall's τ (0.071 $\rightarrow$ 0.079). We observe further slight improvement (0.079 $\rightarrow$ 0.086) when we replace the fixed random score embeddings with learnable ones (Learnable Embedding). Introducing AdaLN transformation [58] on the time ($\boldsymbol{\tau}$) and positional ($\boldsymbol{\Phi}$) encodings further improves τ to 0.145 and 0.171, respectively. As we adopt the quantization idea, grouping nearby scores into a unified embedding, in Sec. 3.2 with this change, we experiment without it (using a sinusoidal function [80] instead), and observe significant performance drop (0.171 $\rightarrow$ 0.125). This confirms that quantization is superior to treating the score

Model	$\tau \uparrow$	$\rho \uparrow$
CSTA [67]	0.128	0.185
SummDiff (1 step)	0.170	0.234
SummDiff (10 steps)	0.175	0.238
SummDiff (100 steps)	**0.182**	**0.245**

Table 6. Performance with various numbers of DDIM steps

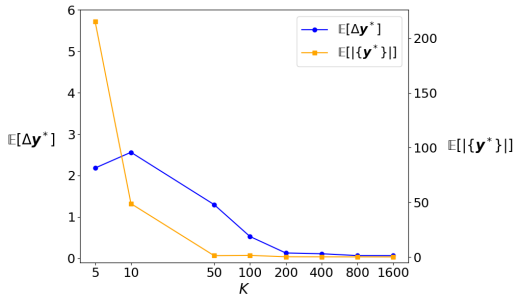


Figure 6. **Measured $\mathbb{E}|\{\boldsymbol{y}^*\}|$ and $\mathbb{E}[\Delta \boldsymbol{y}^*]$ with respect to the quantization strength K.** A larger K is associated with a smaller number of allowed solutions for the given KP.

as a continuous scalar. On top of this, we combine our model with two widely-known ideas, classifier-free guidance (CFG) [24] and self-Attention guidance (SAG) [30], for further improvement. See Appendix B for details.

Number of DDIM Steps. Tab. 6 suggests that SummDiff outperforms CSTA [67], the best-performing baseline, even with a single DDIM [68] step. Beyond this, SummDiff further improves its performance with more DDIM steps.

Effect of Quantization Strength. Solving a KP usually has a unique solution if the profits v_i are continuous. As quantization unifies importance score values within the same interval to a single embedding, it may lead to multiple solutions to $\mathcal{KP}(\tilde{\boldsymbol{v}}, \boldsymbol{w}, \rho)$, where $\tilde{\boldsymbol{v}}$ is a quantized version of $\boldsymbol{v}$.

To analyze this effect, we measure the average number of solutions $\mathbb{E}_{\boldsymbol{v}_1 \sim U(0,1)^{\otimes N}}[|\{\boldsymbol{y}^* \in \{0,1\}^N | \boldsymbol{y}^* = \mathcal{KP}(\tilde{\boldsymbol{v}}, \boldsymbol{w}, \rho)\}|]$ while varying K, taking expectation over $\boldsymbol{v}$, which is a value vector sampled from a joint uniform distribution. We also consider the average L_1 difference between two summaries, $\mathbb{E}_{\boldsymbol{v}_1 \sim U(0,1)^{\otimes N}}[\Delta \boldsymbol{y}^*] \equiv \mathbb{E}_{\boldsymbol{v}_1 \sim U(0,1)^{\otimes N}, \boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{v}_1, \frac{1}{K}\mathbf{I}_N)}[\|\mathcal{KP}(\tilde{\boldsymbol{v}}_1, \boldsymbol{w}, \rho) - \mathcal{KP}(\tilde{\boldsymbol{v}}_2, \boldsymbol{w}, \rho)\|_1]$, obtained by solving the KP with two similar but distinct value vectors, *i.e.*, $\boldsymbol{v}_2 = \boldsymbol{v}_1 + \Delta \boldsymbol{v}$, where $\Delta \boldsymbol{v} \sim \mathcal{N}(0, \frac{1}{K}\mathbf{I}_N)$.

We numerically analyze the relation between these two values, which indicate the scale of multiplicity of the solutions to the KP, and the quantization strength K. As shown in Fig. 6, both the number of solutions $|\{\boldsymbol{y}^*\}|$ and the summary deviation $\Delta \boldsymbol{y}^*$ decrease with larger K, suggesting more accurate summaries.

To sum up, increasing the quantization strength K reduces the number of multiple optimal solutions, leading to a more accurate summary. However, an excessively higher K leads to insufficient samples per bin, degrading the overall performance. See Appendix A for more ablation studies.

5. Related Work

Video Summarization. Early models primarily rely on heuristic unsupervised learning [13, 15, 29, 38, 47, 49–51, 53, 71, 83] to select important or diverse frames, struggling with generalization. Benefiting from annotated datasets, supervised methods [8, 38, 57, 59, 60] have emerged. DSNet [93] and iPTNet [34] improve keyframe selection using frame-level annotations, and SL-Module [82] captures high-level features. With the advent of deep learning, RNNs [84, 88–90] and attention mechanism [4, 6, 33, 35, 39, 42, 77, 85, 94] have been adopted to capture temporal dependencies in videos. Particularly, VASNet [17] and PGL-SUM [3] capture both local and global frame dependencies using self-attention. DMASum [81] introduces mixture of attention layer mitigate the key Softmax Bottleneck. SUM-GAN [50] and AC-SUM-GAN [2] leverage generative adversarial networks.

Multimodal approaches [32, 91] integrate audio or textual data to video summarization. Multimodal transformers are adopted to link frames with corresponding captions, improving context-aware summaries, by CLIP-It [52], MSVA [18], SSPVS [40], and A2Summ [22]. CSTA [67] addresses computational complexity by using CNN-based spatiotemporal attention for efficient frame selection. Unlike these deterministic summarizers, our approach generates multiple plausible summaries by capturing the distribution of video summaries with various perspectives.

Diffusion for Video Tasks. Diffusion [66] have emerged as a groundbreaking tool in high-quality image generation [21, 25, 54, 63, 64, 69, 70] and super-resolution [14, 36, 61, 65]. Recently, they are extended to video generation [7, 11, 16, 19, 26, 27, 44], joint video and audio generation [41, 62, 73], video editing [9, 10, 79], video inpainting [87] and prediction [31, 86]. Furthermore, they are applied to video understanding tasks like moment retrieval [43, 48], video object segmentation [92, 95], action segmentation [45].

6. Summary

Video summarization is inherently subjective since people have different perspectives of a good summary. We suggest a generative viewpoint of this task where the model learns the distribution of good summaries, in contrast to the traditional approach of aggregated importance score regression. Our proposed SummDiff model, adopting diffusion models for video summarization for the first time, is able to generate multiple good summaries conditioned on the input video. Our model not only outperforms other baselines but also demonstrates the ability to generate accurate summaries customized to the individual annotators. We further propose additional metrics that measure the quality of the predicted importance scores through an insight from the actual summary generation using knapsack.

Acknowledgments

This work was supported by Samsung Electronics (IO240512-09881-01), Youlchon Foundation, NRF grants (RS-2021-NR05515, RS-2024-00336576, RS-2023-0022663) and IITP grants (RS-2022-II220264, RS-2024-00353131) by the government of Korea.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8M: A large-scale video classification benchmark. *arXiv:1609.08675*, 2016. 4

[2] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, and Ioannis Patras. AC-SUM-GAN: Connecting actor-critic and generative adversarial networks for unsupervised video summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(8): 3278–3292, 2020. 1, 4, 8, iii

[3] Evlampios Apostolidis, Georgios Balaouras, Vasileios Mezaris, and Ioannis Patras. Combining global and local attention with positional encoding for video summarization. In *IEEE international symposium on multimedia (ISM)*, 2021. 1, 2, 4, 7, 8, i, iii

[4] Evlampios Apostolidis, Georgios Balaouras, Vasileios Mezaris, and Ioannis Patras. Summarizing videos using concentrated attention and considering the uniqueness and diversity of the video frames. In *ICMR*, 2022. 8

[5] Tarik Belgacem and Mhand Hifi. Sensitivity analysis of the knapsack problem: Tighter lower and upper bound limits. *Journal of systems science and systems engineering*, 17:156–170, 2008. 7

[6] Manjot Bilkhu, Siyang Wang, and Tushar Dobhal. Attention is all you need for videos: Self-attention based video summarization using universal transformers. *arXiv:1906.02792*, 2019. 8

[7] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 8

[8] Sijia Cai, Wangmeng Zuo, Larry S Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *ECCV*, 2018. 8

[9] Duygu Ceylan, Chun-Hao P Huang, and Niloy J Mitra. Pix2video: Video editing using image diffusion. In *ICCV*, 2023. 8

[10] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. StableVideo: Text-driven consistency-aware diffusion video editing. In *ICCV*, 2023. 8

[11] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. VideoCrafter2: Overcoming data limitations for high-quality video diffusion models. In *CVPR*, 2024. 8

[12] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. In *ICLR*, 2022. 3

[13] Wen-Sheng Chu, Yale Song, and Alejandro Jaimes. Video co-summarization: Video summarization by visual co-occurrence. In *CVPR*, 2015. 8

[14] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction. In *CVPR*, 2022. 8

[15] Sandra Eliza Fontes De Avila, Ana Paula Brandao Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern recognition letters*, 32(1):56–68, 2011. 8

[16] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *ICCV*, 2023. 8

[17] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *ACCV Workshop*, 2019. 1, 2, 4, 7, 8, i, iii

[18] Junaid Ahmed Ghauri, Sherzod Hakimov, and Ralph Ewerth. Supervised video summarization via multiple feature sets with parallel attention. In *ICME*, 2021. 8, ii

[19] Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Fei-Fei Li, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. In *ECCV*, 2025. 8

[20] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *ECCV*, 2014. 1, 4, 5

[21] Jaehoon Hahm, Junho Lee, Sunghyun Kim, and Joonseok Lee. Isometric representation learning for disentangled latent space of diffusion models. In *ICML*, 2024. 8

[22] Bo He, Jun Wang, Jielin Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. Align and attend: Multimodal summarization with dual contrastive losses. In *CVPR*, 2023. 1, 4, 7, 8, i, iii

[23] Mhand Hifi, Hedi Mhalla, and Slim Sadfi. Sensitivity of the optimum to perturbations of the profit or weight of an item in the binary knapsack problem. *Journal of Combinatorial Optimization*, 10:239–260, 2005. 4, 6

[24] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021. 1, 7, 8, i

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 3, 8

[26] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv:2210.02303*, 2022. 8

[27] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022. 8

[28] Fa-Ting Hong, Xuanteng Huang, Wei-Hong Li, and Wei-Shi Zheng. Mini-net: Multiple instance ranking network for video highlight detection. In *ECCV*, 2020. 4

[29] Richang Hong, Jinhui Tang, Hung-Khoon Tan, Shuicheng Yan, Chongwah Ngo, and Tat-Seng Chua. Event driven summarization for web videos. In *SIGMM workshop on Social media*, 2009. 8

[30] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *ICCV*, 2023. 7, 8, i

[31] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv:2206.07696*, 2022. 8

[32] Tzu-Chun Hsu, Yi-Sheng Liao, and Chun-Rong Huang. Video summarization with spatiotemporal vision transformer. *IEEE Transactions on Image Processing*, 2023. 8, ii

[33] Jia-Hong Huang, Luka Murn, Marta Mrak, and Marcel Worring. GPT2MVS: Generative pre-trained transformer-2 for multi-modal video summarization. In *ICMR*, 2021. 8

[34] Hao Jiang and Yadong Mu. Joint video summarization and moment localization by cross-task sample transfer. In *CVPR*, 2022. 1, 4, 8, iii

[35] Yunjae Jung, Donghyeon Cho, Sanghyun Woo, and In So Kweon. Global-and-local relative position embedding for unsupervised video summarization. In *ECCV*, 2020. 8

[36] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *NeurIPS*, 2022. 8

[37] Maurice G Kendall. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251, 1945. 4

[38] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In *CVPR*, 2013. 8

[39] Haopeng Li, Qiuhong Ke, Mingming Gong, and Rui Zhang. Video joint modelling based on hierarchical transformer for co-summarization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3904–3917, 2022. 8

[40] Haopeng Li, Qiuhong Ke, Mingming Gong, and Tom Drummond. Progressive video summarization via multimodal self-supervised learning. In *WACV*, 2023. 4, 8, ii

[41] Judith Li, Xuchan Bao, Zhong Yi Wan, Kun Su, Timo Denk, Dima Kuzmin, Joonseok Lee, and Fei Sha. Diff4steer: Steerable diffusion prior for generative music retrieval with semantic guidance. In *ICASSP*, 2024. 8

[42] Ping Li, Qinghao Ye, Luming Zhang, Li Yuan, Xianghua Xu, and Ling Shao. Exploring global diverse attention via pairwise temporal relation for video summarization. *Pattern Recognition*, 111:107677, 2021. 8

[43] Pandeng Li, Chen-Wei Xie, Hongtao Xie, Liming Zhao, Lei Zhang, Yun Zheng, Deli Zhao, and Yongdong Zhang. MomentDiff: Generative video moment retrieval from random to real. In *NeurIPS*, 2024. 8

[44] Jingyun Liang, Yuchen Fan, Kai Zhang, Radu Timofte, Luc Van Gool, and Rakesh Ranjan. MoVideo: Motion-aware video generation with diffusion model. In *ECCV*, 2025. 8

[45] Daochang Liu, Qiyue Li, Anh-Dung Dinh, Tingting Jiang, Mubarak Shah, and Chang Xu. Diffusion action segmentation. In *ICCV*, 2023. 8

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017. ii

[47] Zheng Lu and Kristen Grauman. Story-driven summarization for egocentric video. In *CVPR*, 2013. 8

[48] Dezhao Luo, Shaogang Gong, Jiabo Huang, Hailin Jin, and Yang Liu. Generative video diffusion for unseen cross-domain video moment retrieval. *arXiv:2401.13329*, 2024. 8

[49] Yu-Fei Ma, Lie Lu, Hong-Jiang Zhang, and Mingjing Li. A user attention model for video summarization. In *ACM MM*, 2002. 8

[50] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In *CVPR*, 2017. 1, 4, 8, ii, iii

[51] Padmavathi Mundur, Yong Rao, and Yelena Yesha. Keyframe-based video summarization using delaunay clustering. *International Journal on Digital Libraries*, 6:219–232, 2006. 8

[52] Medhini Narasimhan, Anna Rohrbach, and Trevor Darrell. Clip-it! language-guided video summarization. In *NeurIPS*, 2021. 4, 8

[53] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Automatic video summarization by graph modeling. In *ICCV*, 2003. 8

[54] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. 1, 8

[55] Mayu Otani, Yuta Nakashima, Esa Rahtu, and Janne Heikkila. Rethinking the evaluation of video summaries. In *CVPR*, 2019. 4

[56] Rameswar Panda and Amit K Roy-Chowdhury. Collaborative summarization of topic-related videos. In *CVPR*, 2017. 4

[57] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, and Amit K Roy-Chowdhury. Weakly supervised summarization of web videos. In *ICCV*, 2017. 8

[58] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 2, 3, 7

[59] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *ECCV*, 2014. 4, 8

[60] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *CVPR*, 2019. 8

[61] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 8

[62] Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, and Baining Guo. MM-diffusion: Learning multi-modal diffusion models for joint audio and video generation. In *CVPR*, 2023. 8

[63] Suho Ryu, Kihyun Kim, Eugene Baek, Dongsoo Shin, and Joonseok Lee. Towards scalable human-aligned benchmark for text-guided image editing. In *CVPR*, 2025. 8

[64] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour,

Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022. 8

[65] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022. 8

[66] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 8

[67] Jaewon Son, Jaehun Park, and Kwangsu Kim. CSTA: CNN-based spatiotemporal attention for video summarization. In *CVPR*, 2024. 1, 2, 4, 5, 7, 8, i, ii, iii

[68] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2020. 3, 8, ii

[69] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 8

[70] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. In *NeurIPS*, 2020. 8

[71] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. TVSum: Summarizing web videos using titles. In *CVPR*, 2015. 1, 4, 5, 8

[72] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2020. 1

[73] Kun Su, Judith Li, Qingqing Huang, Dima Kuzmin, Joonseok Lee, Chris Donahue, Fei Sha, Aren Jansen, Yu Wang, Mauro Verzetti, and Timo Denk. V2Meow: Meowing to the visual beat via video-to-music generation. In *AAAI*, 2024. 8

[74] Jinhwan Sul, Jihoon Han, and Joonseok Lee. Mr. HiSum: A large-scale dataset for video highlight detection and summarization. In *NeurIPS*, 2024. 4, 5, 7

[75] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 4

[76] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 4

[77] Hacene Terbouche, Maryan Morel, Mariano Rodriguez, and Alice Othmani. Multi-annotation attention model for video summarization. In *CVPR*, 2023. 4, 8

[78] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. ii

[79] Shuyuan Tu, Qi Dai, Zhi-Qi Cheng, Han Hu, Xintong Han, Zuxuan Wu, and Yu-Gang Jiang. MotionEditor: Editing video motion via content-aware diffusion. In *CVPR*, 2024. 8

[80] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 2, 3, 7

[81] Junyan Wang, Yang Bai, Yang Long, Bingzhang Hu, Zhenhua Chai, Yu Guan, and Xiaolin Wei. Query twice: Dual mixture attention meta learning for video summarization. In *ACM MM*, 2020. 4, 5, 8

[82] Minghao Xu, Hang Wang, Bingbing Ni, Riheng Zhu, Zhenbang Sun, and Changhu Wang. Cross-category video highlight detection via set-based learning. In *ICCV*, 2021. 1, 4, 7, 8, i, iii

[83] Hong Jiang Zhang, Jianhua Wu, Di Zhong, and Stephen W Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern recognition*, 30(4):643–658, 1997. 8

[84] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *ECCV*, 2016. 4, 8, ii

[85] Yunzuo Zhang, Yameng Liu, Weili Kang, and Ran Tao. VSS-Net: visual semantic self-mining network for video summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 4, 8

[86] Zhicheng Zhang, Junyao Hu, Wentao Cheng, Danda Paudel, and Jufeng Yang. ExtDM: Distribution extrapolation diffusion model for video prediction. In *CVPR*, 2024. 8

[87] Zhixing Zhang, Bichen Wu, Xiaoyan Wang, Yaqiao Luo, Luxin Zhang, Yinan Zhao, Peter Vajda, Dimitris Metaxas, and Licheng Yu. AVID: Any-length video inpainting with diffusion model. In *CVPR*, 2024. 8

[88] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hierarchical recurrent neural network for video summarization. In *ACM MM*, 2017. 8

[89] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. HSA-RNN: Hierarchical structure-adaptive rnn for video summarization. In *CVPR*, 2018.

[90] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization. *IEEE Transactions on Industrial Electronics*, 68(4):3629–3637, 2020. 8

[91] Bin Zhao, Maoguo Gong, and Xuelong Li. Hierarchical multimodal transformer to summarize videos. *Neurocomputing*, 468:360–369, 2022. 8

[92] Chuanwei Zhou, Chunyan Xu, Jun Li, Zhen Cui, and Jian Yang. Quality-aware pattern diffusion for video object segmentation. *Neurocomputing*, 528:148–159, 2023. 8

[93] Wencheng Zhu, Jiwen Lu, Jiahao Li, and Jie Zhou. DSNet: A flexible detect-to-summarize network for video summarization. *IEEE Transactions on Image Processing*, 30:948–962, 2020. 4, 8

[94] Wencheng Zhu, Yucheng Han, Jiwen Lu, and Jie Zhou. Relational reasoning over spatial-temporal graphs for video summarization. *IEEE Transactions on Image Processing*, 31: 3017–3031, 2022. 8

[95] Zixin Zhu, Xuelu Feng, Dongdong Chen, Junsong Yuan, Chunming Qiao, and Gang Hua. Exploring pre-trained text-to-video diffusion models for referring video object segmentation. *arXiv:2403.12042*, 2024. 8

[96] Daniel Zwillinger and Stephen Kokoska. *CRC standard probability and statistics tables and formulae*. CRC Press, 1999. 4

SummDiff: Generative Modeling of Video Summarization with Diffusion

Supplementary Material

A. Additional Ablation Studies

Effect of Quantization. We explore various number of segments (K) for the codebook $\mathcal{C}$, uniformly splitting the score range $[0, 1]$. According to Tab. I, the best performance is achieved around 200 to 400. With $K < 200$, the performance degrades because it limits the ability of our model to distinguish different scores, treating a wide range of scores with the same embedding. With too large K, on the other hand, the model would suffer from the lack of samples per each bin, degrading the performance.

K	τ	ρ
5	0.145	0.200
10	0.147	0.202
50	0.147	0.202
100	0.171	0.235
200	0.175	0.238
400	0.173	0.237
800	0.172	0.235

Table I. Effect of quantization strength K

Visualization of Histograms from Quantization. To further visualize the effect of quantization with varying K, we illustrate the distribution of quantized scores for different values of K using histograms, and count the number of scores falling into each bin on a subset of Mr. HiSum dataset shown in Fig. I. When K is too small (left), quantization becomes too coarse, collapsing diverse scores into the same bin and causing more multiple optimal solutions observed in Fig. 6. When K is too large (right), bins become too sparse, leading to unstable estimates with insufficient samples per bin. Both extremes hurt performance. As shown in Appendix A, the best results are achieved when K strikes a balance between granularity and robustness.

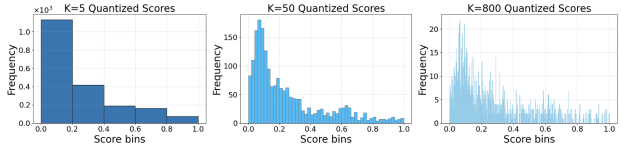


Figure I. Histogram of quantized importance scores for varying K on a subset of the Mr. HiSum dataset.

Inference Time. The iterative sampling process in generative diffusion might raise concerns about slower inference time. We report the average inference time of top-performing models under the same condition in Tab. II. SummDiff (1 step) takes comparable time to others including CSTA [67], and Tab. 6 confirms that this setting outperforms other baselines. In short, SummDiff (1 step) achieves moderately fast inference time with a reasonably strong summarization performance, effectively balancing them.

Model	Inference Time (ms)
CSTA [67]	11.70 ± 0.11
PGL SUM [3]	19.32 ± 0.50
SL_Module [82]	5.20 ± 0.62
VASNET [17]	1.11 ± 0.23
A2Summ [22]	9.10 ± 2.25
SummDiff (1 step)	11.02 ± 1.92
SummDiff (10 steps)	49.73 ± 1.55

Table II. Comparison on inference time

Training on aggregated scores. To further investigate the importance of training on individual annotator scores, we conduct additional experiments on SumMe with various number of annotations, $|\mathcal{R}| \in \{5, 10, \text{All}\}$. We train a model on a randomly selected set of annotations of size $|\mathcal{R}|$, either on individual annotations or on their aggregated scores. As seen in the table, training on individual scores consistently outperforms. This supports our claim that using individual scores aligns well with our generative approach and leads to higher-quality summaries.

$\lvert\mathcal{R}\rvert$	Training	τ	ρ
5	Agg	0.130	0.144
	Ind	**0.211**	**0.236**
10	Agg	0.145	0.161
	Ind	**0.227**	**0.253**
All	Agg	0.176	0.196
	Ind	**0.256**	**0.285**

Table III. Performance comparison on SumMe when training with aggregated (Agg) versus individual (Ind) annotator scores, across different numbers of annotations $|\mathcal{R}| \in \{5, 10, \text{All}\}$.

B. Classifier-free Guidance and Self-attention Guidance

We integrate two widely-known ideas for further improvement of SummDiff. First, we adopt the classifier-free guidance (CFG) [24], $\tilde{f}_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}) = (1 + w) f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}) - w f_\theta(\mathcal{C}(\mathbf{u}_t), t, \varnothing)$, where $\varnothing$ is the null (black) video, and w determines the extent to which unconditioned information is used at inference.

Second, we add self-Attention guidance (SAG) [30], leveraging the intermediate self-attention maps from diffusion models to improve stability. It works by selectively blurring the areas that the diffusion models focus on during each step, using an adversarial approach to adjust and guide

the model's attention as it progresses. Specifically,

$$\tilde{f}_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}) = \tilde{f}_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}) + (1+s)(\tilde{f}_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z})) - \tilde{f}_\theta(\mathcal{C}(\hat{\mathbf{u}}_t), t, \mathbf{Z}) \quad (1)$$

where A_t denotes the attention map, $M_t = \mathbb{1}(A_t > \psi)$ is a binary mask indicating where A_t exceeds a threshold ψ, and $\odot$ is the Hadamard product. The intermediate reconstruction $\mathcal{C}(\hat{\mathbf{u}}_t)$ selectively combines the original signal $\mathcal{C}(\mathbf{u}_t)$ and its noised version $\mathcal{C}(\tilde{\mathbf{u}}_t)$ based on M_t by

$$\mathcal{C}(\hat{\mathbf{u}}_t) = (1 - M_t) \odot \mathcal{C}(\mathbf{u}_t) + M_t \odot \mathcal{C}(\tilde{\mathbf{u}}_t), \quad (2)$$

where $\mathcal{C}(\tilde{\mathbf{u}}_0)$ is obtained by convolving $\mathcal{C}(\mathbf{u}_0)$ with a Gaussian kernel G_σ. Finally, $\mathcal{C}(\hat{\mathbf{u}}_0)$ is computed by

$$\mathcal{C}(\hat{\mathbf{u}}_0) = (\mathcal{C}(\mathbf{u}_t) - \sqrt{1-\bar{\alpha}_t} f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z}))/\sqrt{\bar{\alpha}_t}. \quad (3)$$

$\mathcal{C}(\tilde{\mathbf{u}}_t)$ is obtained by diffusing with noise $f_\theta(\mathcal{C}(\mathbf{u}_t), t, \mathbf{Z})$ from $\mathcal{C}(\tilde{\mathbf{u}}_0)$.

In Tab. 5, we observe extra gains with these, achieving 0.177 of τ. A similar pattern of gradual improvement is observed with Spearman's ρ, as seen in Tab. 5.

C. Qualitative Results on Contours

We provide additional examples of generated summaries in Fig. II. The summaries annotated by human raters (● Annotator) and those generated by ours (● SummDiff) and three baselines (● CSTA, ● PGL-SUM, and ● VASNet) are projected to 2D using PCA. The annotated summaries are visualized using a blue contour map created through Gaussian kernel density estimation. This highlights that SummDiff produces a diverse range of summaries that closely match those provided by human annotators. For instance, in the top-right plot for video 48, human evaluators identified three distinct ways to summarize the content. SummDiff successfully generates summaries that capture all three variations. In contrast, the baseline models typically produce a single, less accurate summary, failing to capture the variability seen in human-generated summaries. Similar trends are observed across the other three plots.

D. Evaluation Results with Standard Deviation

Tab. IV shows the full evaluation results from Tab. 3 with standard deviation. As seen in the table, our method is superior to all other baselines statistically significantly.

E. Implementation Details

We uniformly sample frames from each video at 1 fps matching the ground truth label provided in the Mr. HiSum dataset. For TVSum and SumMe, the videos are subsampled to 2 fps as in [18, 32, 40, 50, 67, 84]. For Mr. HiSum, we employ 2 transformer layers for visual encoding and 2 additional layers for denoising. Each layer has a hidden size of 256, 8 attention heads, and feed-forward network with a dimensionality of 1024. For TVSum and SumMe we adopt 1D convolution layer followed by an MLP for the decoder due to the small dataset size. We use AdamW optimizer [46] with cosine annealing [78], gradually reducing the learning rate from $5 \cdot 10^{-5}$ initially. We also used EMA decay of 0.999 for Mr. HiSum. We set the batch size to 256, and train up to 200 epochs. ϵ is set to 10^{-3} when clipping $\boldsymbol{s}_0$. For inference, we take 10 DDIM [68] steps by default. We conduct all experiments on a single NVIDIA A5000 GPU.

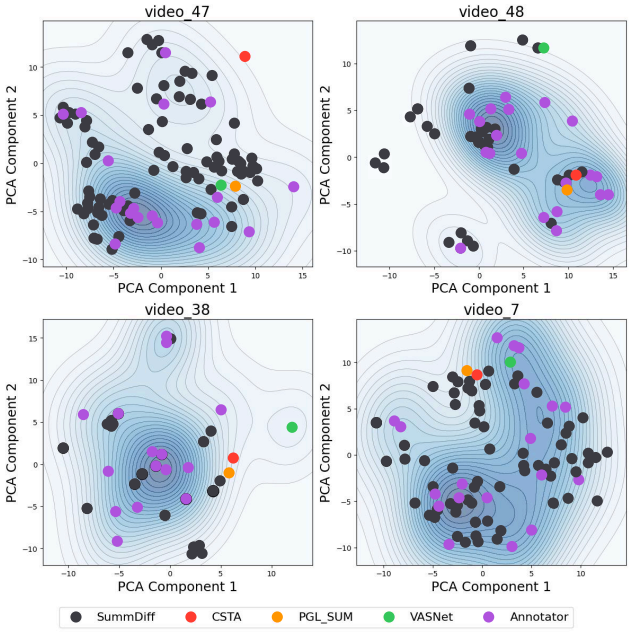


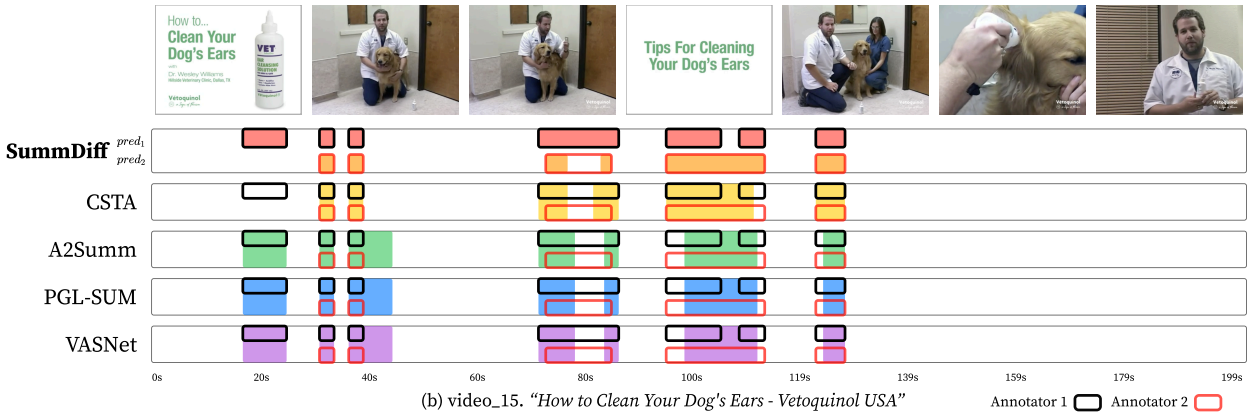
Figure II. Distribution of ground truth and generated summaries of selected videos

We investigate various batch sizes from the set $\{32, 64, 256\}$ and determined that a batch size of 256 yielded the best performance for SummDiff model. We apply tuning procedures consistently across all models. For SumMe and TVSum, batch size of 20 and 40 yielded the best result respectively.

Furthermore, we experiment with learning rates and weight decay within the range of $\{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}, 5 \cdot 10^{-5}, 10^{-5}\}$. Learning rate of $5 \cdot 10^{-5}$ and weight decay of $5 \cdot 10^{-4}$ is found to be optimal for our SummDiff model, and similar tuning procedure has been applied to other models.

Model	Year	$\tau \uparrow$	$\rho \uparrow$	$\text{MAP}_{\rho=50\%} \uparrow$	$\text{MAP}_{\rho=15\%} \uparrow$
SUM-GAN [50]	2017	0.067±0.018	0.095±0.023	59.50±0.16	24.30±0.19
VASNet [17]	2019	0.069±0.000	0.102±0.000	58.69±0.30	25.28±0.40
AC-SUM-GAN [2]	2020	0.012±0.003	0.018±0.003	56.40±0.06	21.70±0.08
SL-module [82]	2021	0.060±0.002	0.088±0.003	58.63±0.13	24.95±0.13
PGL-SUM [3]	2021	0.097±0.001	0.141±0.001	61.60±0.14	27.45±0.15
iPTNet [34]	2022	0.020±0.003	0.029±0.004	55.53±0.25	22.74±0.13
A2Summ [22]	2023	0.121±0.001	0.172±0.001	59.18±0.13	30.70±0.21
CSTA [67]	2024	0.128±0.004	0.185±0.006	62.25±0.15	28.42±0.33
SummDiff	Ours	**0.175**±0.005	**0.238**±0.004	**65.44**±0.19	**33.83**±0.44

Table IV. **Comparison of models trained with Mr. HiSum**



(b) video_15. *"How to Clean Your Dog's Ears - Vetoquinol USA"*

Figure III. **Additional demonstration of video summaries generated by competing methods on a TVSum video**. Shaded parts indicate the segments selected by each method, and the two rows of edged boxes within each method indicate two different true annotations. The results clearly demonstrate the effectiveness of SummDiff in capturing multiple plausible summaries for a video.

F. Additional Qualitative Results

As illustrated in Fig. 5, we visualize the summary of videos generated by various models. Specifically, CSTA [67], A2Summ [22], PGL-SUM [3], VASNet [17] is compared against our model, SummDiff. All videos are selected from the test set of TVSum from 5-fold cross validation experiment. Two different summary annotations are visualized in the first and second rows using black and red boxes. The results demonstrate that SummDiff predicts both more accurate and various summaries compared to other baseline models.