Uncertainty-Aware and Decoder-Aligned Learning for Video Summarization

1st Omer Tariq
Perception AI
Neubility Inc.
Seoul, South Korea
omer.tariq@neubility.co.kr

2nd Syed Muhammad Raza
Perception AI
Neubility Inc.
Seoul, South Korea
raza@neubility.co.kr

3rd Jeongbae Son
Perception AI
Neubility Inc.
Seoul, South Korea
thswjdqo92@neubility.co.kr

***Abstract*—Video summarization aims to produce a compact representation of a long video by selecting a subset of temporally important segments that best reflect human preferences. This task is inherently difficult due to strong annotation subjectivity and the reliance on discrete decoding procedures, such as temporal segmentation and knapsack-based selection, during evaluation. Most existing approaches either learn deterministic importance scores that overlook these characteristics or adopt complex generative models that increase training and inference cost. In this paper, we propose VASTSum, an *uncertainty-aware and decoder-aligned* learning framework for video summarization that addresses both challenges within a single-pass model. The proposed method predicts probabilistic frame-level importance scores using a variational formulation, enabling explicit modeling of uncertainty arising from multi-annotator supervision. To account for subjectivity, particularly under binary annotations, we employ a supervision strategy that encourages alignment with plausible human annotation modes rather than enforcing a single consensus target. Furthermore, we introduce a decoder-aligned regularization that promotes stability of knapsack-based summary selection, reducing sensitivity to small perturbations in predicted scores. We evaluate the proposed framework on the SumMe and TVSum benchmarks using standard rank-based metrics. Experimental results show consistent and competitive Kendall and Spearman correlations across multiple data splits, demonstrating improved robustness under annotation disagreement while maintaining efficient single-forward inference. These results indicate that explicitly modeling uncertainty and aligning learning objectives with the decoding stage provide a principled alternative to both deterministic and diffusion-based video summarization methods.**

***Index Terms*—Video Summarization, Uncertainty-Aware, Deep Learning**

I. Introduction

Recently, short-form videos have become a dominant format on online platforms, reflecting a broader shift toward rapid consumption of visual content. Video summarization addresses this demand by selecting a short sequence of temporally coherent segments that preserves the essential content of a long video under a strict length budget. However, what constitutes an "essential" summary is inherently subjective, and standard benchmarks therefore provide multiple human annotations per video, notably SumMe [1] and TVSum [2].

In supervised settings, a typical pipeline predicts frame-level importance scores and then produces a binary summary through a discrete decoding stage. Concretely, predicted scores are aggregated into keyshots via temporal segmentation and a budget-constrained subset is selected using a knapsack optimizer, commonly following the kernel temporal segmentation (KTS) protocols [3]. Because this decoding is discrete, small perturbations in predicted scores can change the selected keyshots, making evaluation sensitive to score calibration and local ranking decisions.

Most prior methods nevertheless learn a deterministic regressor toward an averaged importance target, collapsing multi-annotator disagreement into a single supervision signal and potentially obscuring distinct but valid annotation modes [4]–[6]. Recent generative formulations instead aim to model a distribution of summaries, for example via diffusion-based conditional generation, but iterative inference increases computational cost and can complicate deployment [7], [8]. Moreover, both deterministic and generative approaches are typically optimized with continuous objectives, whereas reported performance is determined after segmentation and knapsack selection, which introduces a training and evaluation mismatch.

To address these challenges, we propose VASTSum, an uncertainty-aware and decoder-aligned learning framework that preserves single-pass efficiency while explicitly accounting for subjective supervision and discrete decoding. We parameterize frame importance as a predictive distribution using a variational formulation, enabling uncertainty to reflect annotator disagreement and ambiguous content. We further optimize against plausible annotation modes rather than enforcing a single collapsed consensus target, and we introduce a decoder-aligned regularization that promotes stability of knapsack-based keyshot selection under small score perturbations. Experiments on SumMe and TVSum demonstrate consistent improvements in rank correlation across standard splits, supporting uncertainty-aware and decoder-aligned learning as a practical alternative to both deterministic attention models [6] and diffusion-based generators [7].

We introduce an uncertainty-aware and decoder-aligned framework that bridges the gap between deterministic attention scoring, exemplified by CSTA [6], and diffusion-based generative summarization, exemplified by SummDiff [7]. Our approach retains single-pass efficiency while explicitly addressing annotation subjectivity and decoding sensitivity.

1) We propose VASTSum, a unified uncertainty-aware variational formulation for frame-importance prediction that outputs both a mean estimate and an uncertainty signal, explicitly capturing ambiguity induced by multi-annotator supervision.
2) We train from plausible annotator-consistent modes rather than collapsing subjective labels into a single averaged target, reducing over-smoothed supervision while preserving distinct summarization preferences.
3) We introduce a decoder-aligned stability regularization that promotes stable keyshot selection under the standard segmentation and knapsack decoding pipeline, mitigating training–evaluation mismatch and improving robustness.

Section II reviews related work on supervised video summarization, covering deterministic attention-based scorers and recent generative formulations. Section III formalizes the task and the segmentation-plus-knapsack decoding protocol. Section IV presents VASTSum, including the hierarchical segment-context scorer, the uncertainty-aware variational importance head, the mode-aligned supervision strategy, and the decoder-aligned stability regularization. Section V describes the experimental setup, including datasets, evaluation protocol, and implementation details. Section VI presents quantitative results and discussion, followed by Section VII, which provides ablation studies analyzing key components of the proposed method. Finally, Section VIII concludes the paper and discusses limitations and future directions.

II. Related Work

Supervised video summarization. Most supervised pipelines predict frame-level importance scores and then apply a discrete decoding stage, typically change-point segmentation followed by budgeted 0/1 knapsack selection. A large body of work learns deterministic scorers using temporal models and attention to capture local and global dependencies (e.g., VASNet [4], PGL-SUM [9]), along with detect-to-summarize and joint learning variants (e.g., DSNet [5], iPTNet [10]). Recent multimodal methods further incorporate language or cross-modal alignment signals to improve semantic relevance (e.g., CLIP-It [11], A2Summ [12]). While effective, these approaches commonly collapse multi-annotator subjectivity into a single target and optimize continuous losses, even though evaluation depends on segmentation and knapsack decoding, which can be sensitive to small score perturbations. CSTA [6] improves efficiency and temporal modeling with CNN-based spatiotemporal attention, but remains a deterministic scoring framework.

Generative diffusion-based summarization. Diffusion models [8] have recently been adapted to generate summary distributions, with SummDiff [7] modeling multiple plausible summaries to reflect diverse human preferences. However, diffusion-based inference typically requires iterative denoising steps, increasing computational cost and complicating deployment when single-pass efficiency is desired. In contrast to both deterministic scorers and iterative diffusion generators, our method targets subjective supervision and discrete decoding directly within a single-forward framework by learning uncertainty-aware importance scores and aligning training with the segmentation-plus-knapsack selection stage.

III. Problem Formulation

Given an untrimmed video, we aim to produce a compact summary that preserves the essential content under a strict length budget. We follow the standard supervised setting where the input is a sequence of pre-extracted frame descriptors sampled from the original timeline. Let

$$\mathbf{X} = \{\mathbf{x}_t \in \mathbb{R}^D\}_{t=1}^{T} \tag{1}$$

denote the sampled feature sequence, where $D = 1024$ in our implementation and T is the number of sampled timesteps. The sampled indices

$$\mathbf{p} = \{p_t\}_{t=1}^{T}, \qquad 0 \le p_1 < \cdots < p_T < N, \tag{2}$$

map each sampled feature to its location on the original timeline of N frames. Benchmarks provide multiple human annotations per video. Let $a \in \{1, \ldots, U\}$ index annotators. TVSum provides continuous frame-importance signals $s^{(a)} \in [0,1]^T$ aligned to the sampled timeline, while SumMe provides binary user summaries on the original timeline; in SumMe, we evaluate each annotator's inclusion signal at the sampled indices, yielding $y^{(a)} \in \{0,1\}^T$. Our model predicts a real-valued importance logit sequence $\mu \in \mathbb{R}^T$, which is interpreted directly for continuous supervision (TVSum) and converted into probabilities for binary supervision (SumMe).

Evaluation and summary generation depend on a discrete decoding stage. A change-point partition produced by Kernel Temporal Segmentation is given as

$$\mathbf{C} = \{(c_k^{\text{start}}, c_k^{\text{end}})\}_{k=1}^{M}, \tag{3}$$

where segments are contiguous, disjoint, and cover the full timeline using inclusive boundaries. A valid summary is represented on the original timeline by $\mathbf{y} \in \{0,1\}^N$ with the standard budget constraint

$$\sum_{n=1}^{N} y_n \le \rho N, \qquad \rho = 0.15. \tag{4}$$

In our pipeline, the predicted importance signal is decoded into a budgeted set of segments via a 0/1 knapsack solver (Sec. IV-E). Since this decoding is discrete, small score perturbations can change the selected segments. The proposed learning objectives therefore combine multi-annotator supervision with an explicit regularizer that promotes stability of budgeted selection (Sec. IV-F).

IV. Proposed Method

Figure 1 illustrates the overall flow of our proposed model, VASTSum.

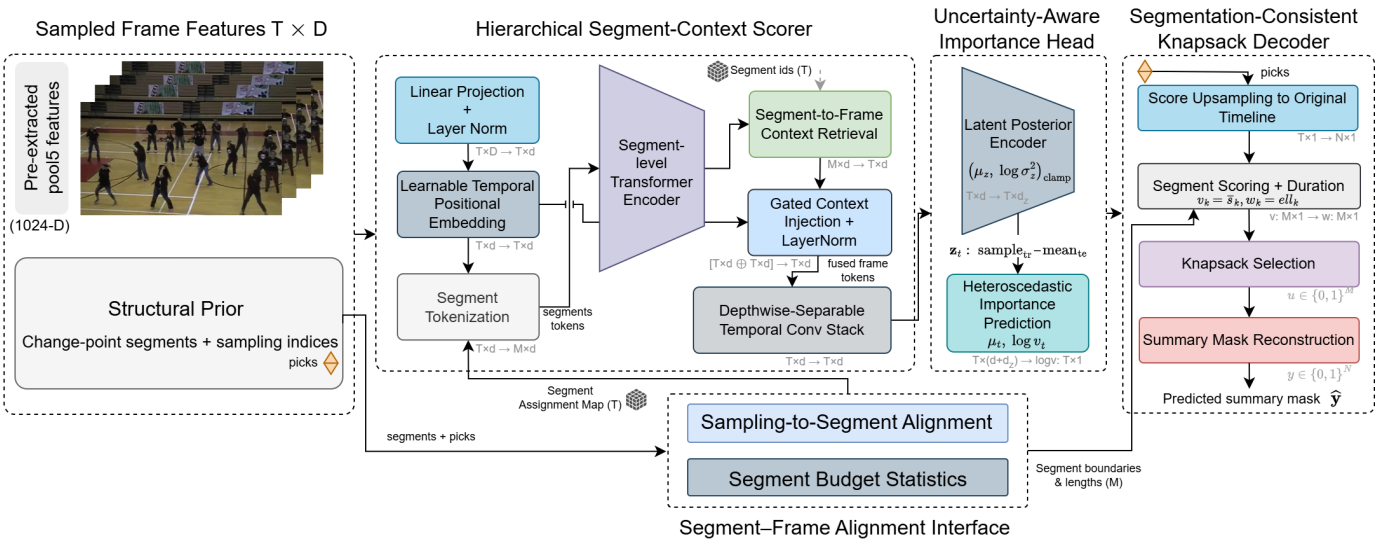


Fig. 1: Overview of VASTSum: Proposed segmentation-consistent video summarization pipeline. Sampled `pool5` frame features are linearly projected and augmented with learnable temporal positional embeddings. Change-point segmentation provides a structural prior by assigning sampled timesteps to segment IDs and estimating segment boundaries/lengths. A hierarchical segment-context scorer aggregates frames into segment tokens, encodes global segment relations via a segment-level Transformer, propagates segment context back to frames with gated injection, and refines temporal representations using a lightweight depthwise-separable temporal convolution stack. A variational frame-importance head predicts a per-frame latent posterior and heteroscedastic importance parameters, sampling during training and using the posterior mean at inference for deterministic scoring. Predicted scores are upsampled to the full timeline using the sampling indices, and a segmentation-consistent 0–1 knapsack decoder selects segments under a length budget to output the binary summary mask.

A. Input video representation

We follow the standard supervised summarization protocol and operate on pre-extracted frame descriptors [6], [7]. Given a video of length N frames, we sample T timesteps and denote the resulting feature sequence by

$$\mathbf{X} = \{\mathbf{x}_t \in \mathbb{R}^D\}_{t=1}^{T}. \tag{5}$$

In our implementation, each $\mathbf{x}_t$ is a pooled convolutional descriptor (*pool5*) and $D = 1024$. The sampling indices

$$\mathbf{p} = \{p_t\}_{t=1}^{T}, \qquad 0 \le p_1 < \cdots < p_T < N, \tag{6}$$

map sampled timesteps to the original timeline and are used to (i) align supervision to the sampled sequence and (ii) expand predicted scores back to the original length during decoding.

B. Segmentation-aware indexing

The change-point partition produced by KTS defines a segment structure that is shared by both the scorer and the budgeted decoder. We first map each sampled timestep to its segment identity. For each pick p_t, we assign

$$s_t = k \quad \Longleftrightarrow \quad c_k^{\text{start}} \le p_t \le c_k^{\text{end}}, \qquad t = 1, \ldots, T, \tag{7}$$

which yields the segment-id sequence $\mathbf{s} = \{s_t\}_{t=1}^{T}$. This induces per-segment index sets on the sampled timeline,

$$\mathcal{I}_k = \{t \in \{1, \ldots, T\} \mid s_t = k\}, \tag{8}$$

and segment lengths on the original timeline,

$$\ell_k = c_k^{\text{end}} - c_k^{\text{start}} + 1, \qquad k = 1, \ldots, M. \tag{9}$$

These quantities provide the interface between frame-level prediction and segment-level selection: the model can pool frame evidence within $\mathcal{I}_k$, while the decoder enforces the budget using ℓ_k.

C. Hierarchical segment-context scoring network

Our scorer is a single-pass hierarchical network that performs global reasoning over segment tokens and then refines frame-level representations for fine-grained ranking. Let d denote the model dimension. We first project sampled frame features to d and add a learnable positional embedding:

$$\mathbf{h}_t^{(0)} = \mathrm{LN}(\mathbf{W}_{\text{in}}\mathbf{x}_t + \mathbf{b}_{\text{in}}) + \boldsymbol{\phi}_t, \qquad t = 1, \ldots, T. \tag{10}$$

a) Segment tokenization.: We form one segment token per change-point segment by mean pooling the projected frame tokens:

$$\mathbf{z}_k^{(0)} = \frac{1}{|\mathcal{I}_k|} \sum_{t \in \mathcal{I}_k} \mathbf{h}_t^{(0)}, \qquad k = 1, \ldots, M, \tag{11}$$

and stack them into $\mathbf{Z}^{(0)} \in \mathbb{R}^{M \times d}$.

b) Global segment transformer.: We apply L Transformer encoder layers to model long-range dependencies over the segment sequence. Using a pre-normalized residual form, each layer updates

$$\mathbf{U}^{(\ell)} = \mathbf{Z}^{(\ell)} + \mathrm{MHA}\Big(\mathrm{LN}(\mathbf{Z}^{(\ell)})\Big), \tag{12}$$

$$\mathbf{Z}^{(\ell+1)} = \mathbf{U}^{(\ell)} + \mathrm{FFN}\Big(\mathrm{LN}(\mathbf{U}^{(\ell)})\Big), \qquad \ell = 0, \dots, L-1. \tag{13}$$

For head $h \in \{1, \dots, H\}$ with $d_h = d/H$, attention is

$$\mathrm{Attn}_h(\mathbf{Z}) = \mathrm{softmax}\left(\frac{\mathbf{Q}_h \mathbf{K}_h^\top}{\sqrt{d_h}}\right)\mathbf{V}_h, \tag{14}$$

$$\mathbf{Q}_h = \mathbf{Z}\mathbf{W}_h^Q, \tag{15}$$

$$\mathbf{K}_h = \mathbf{Z}\mathbf{W}_h^K, \tag{16}$$

$$\mathbf{V}_h = \mathbf{Z}\mathbf{W}_h^V. \tag{17}$$

and MHA concatenates the heads followed by an output projection. The feed-forward block is applied token-wise,

$$\mathrm{FFN}(\mathbf{z}) = \mathbf{W}_2\,\mathrm{GELU}(\mathbf{W}_1\mathbf{z} + \mathbf{b}_1) + \mathbf{b}_2. \tag{18}$$

The resulting contextualized segment representations are $\mathbf{C} = \mathbf{Z}^{(L)} \in \mathbb{R}^{M\times d}$.

c) Frame–segment fusion and temporal refinement.: Each frame retrieves the context of its assigned segment,

$$\mathbf{g}_t = \mathbf{c}_{s_t} \in \mathbb{R}^d, \tag{19}$$

and we fuse local and global information using a learned gate:

$$\boldsymbol{\alpha}_t = \sigma\left(\mathbf{W}_g \begin{bmatrix} \mathbf{h}_t^{(0)} \\ \mathbf{g}_t \end{bmatrix} + \mathbf{b}_g\right), \tag{20}$$

$$\tilde{\mathbf{h}}_t = \mathrm{LN}\Big(\mathbf{h}_t^{(0)} + \boldsymbol{\alpha}_t \odot \mathbf{g}_t\Big). \tag{21}$$

We then refine the fused sequence with a lightweight temporal convolutional stack implemented as depthwise separable 1D convolutions with pointwise mixing, repeated L_c times. Writing $\tilde{\mathbf{H}} \in \mathbb{R}^{T\times d}$ for the stacked fused tokens, the refinement is applied with a residual connection,

$$\hat{\mathbf{H}} = \tilde{\mathbf{H}} + \Psi(\tilde{\mathbf{H}}), \tag{22}$$

where Ψ denotes the temporal convolutional operator. The refined frame tokens $\hat{\mathbf{H}}$ parameterize the probabilistic importance head.

D. Probabilistic frame-importance head

To represent annotation variability without resorting to iterative sampling at inference, we predict a conditional distribution over frame importance using a lightweight variational head. Given the refined frame token $\hat{\mathbf{h}}_t \in \mathbb{R}^d$, we define a diagonal Gaussian posterior over a latent variable $\mathbf{z}_t \in \mathbb{R}^{d_z}$,

$$q_\theta(\mathbf{z}_t \mid \hat{\mathbf{h}}_t) = \mathcal{N}\big(\boldsymbol{\mu}_{z,t}, \mathrm{diag}(\boldsymbol{\sigma}_{z,t}^2)\big), \tag{23}$$

where $(\boldsymbol{\mu}_{z,t}, \log \boldsymbol{\sigma}_{z,t}^2)$ are produced by linear projections of $\hat{\mathbf{h}}_t$. For numerical stability we clamp the latent log-variance elementwise,

$$\log \boldsymbol{\sigma}_{z,t}^2 \leftarrow \mathrm{clip}\big(\log \boldsymbol{\sigma}_{z,t}^2, -10, 5\big). \tag{24}$$

During training we sample using the reparameterization trick,

$$\mathbf{z}_t = \boldsymbol{\mu}_{z,t} + \boldsymbol{\sigma}_{z,t} \odot \boldsymbol{\epsilon}_t, \qquad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{25}$$

whereas at inference we use the posterior mean $\mathbf{z}_t = \boldsymbol{\mu}_{z,t}$ to obtain a deterministic score sequence.

Conditioned on $[\hat{\mathbf{h}}_t; \mathbf{z}_t]$, a small MLP produces a hidden representation that is mapped to a scalar importance logit and an observation noise term:

$$\mu_t = f_\mu([\hat{\mathbf{h}}_t; \mathbf{z}_t]), \qquad \log v_t = f_v([\hat{\mathbf{h}}_t; \mathbf{z}_t]), \tag{26}$$

with $\mu_t \in \mathbb{R}$ and $v_t > 0$. We also clamp the predicted log-variance,

$$\log v_t \leftarrow \mathrm{clip}(\log v_t, -10, 5), \tag{27}$$

which prevents unstable gradients when optimizing heteroscedastic likelihoods.

For TVSum, $(\mu_t, \log v_t)$ parameterize a heteroscedastic Gaussian model used by the multi-annotator likelihood in Sec. IV-F. For SumMe, we use the logit μ_t and obtain a calibrated probability by temperature-scaled sigmoid,

$$p_t = \sigma\left(\frac{\mu_t}{T_s}\right), \tag{28}$$

where $T_s > 0$ is fixed. The downstream decoder consumes μ_t for TVSum and p_t for SumMe, while the KL regularizer acts on the latent posterior parameters.

E. Budgeted decoding

Given an importance signal on the sampled timeline, we generate a summary using the standard segmentation-plus-knapsack protocol. We first expand sampled scores to the original timeline using the sampling indices. Let $\tilde{\mu} \in \mathbb{R}^T$ denote the sequence provided to the decoder, where $\tilde{\mu} = \mu$ for TVSum and $\tilde{\mu} = p$ for SumMe. We construct frame scores $\mathbf{f} \in \mathbb{R}^N$ by piecewise constant assignment between consecutive picks:

$$f_n = \tilde{\mu}_t \quad \text{for} \quad n \in [p_t, p_{t+1}), \qquad t = 1, \dots, T-1, \tag{29}$$

and $f_n = \tilde{\mu}_T$ for $n \in [p_T, N)$.

For each segment k, the cost is its length $w_k = \ell_k$ and the value is the mean score within the segment:

$$v_k = \frac{1}{w_k} \sum_{n=c_k^{\mathrm{start}}}^{c_k^{\mathrm{end}}} f_n. \tag{30}$$

The decoder selects a subset of segments by solving

$$\max_{\mathbf{u}\in\{0,1\}^M} \sum_{k=1}^{M} u_k v_k \quad \text{s.t.} \quad \sum_{k=1}^{M} u_k w_k \le \rho N, \tag{31}$$

and expands selected segments to the binary summary $\mathbf{y} \in \{0,1\}^N$ by setting $y_n = 1$ for frames that fall inside any selected segment.

F. Training objectives and optimization

Training combines dataset-aligned supervision, a pairwise ranking regularizer, a KL penalty for the variational latent, and a decoder-aligned stability regularizer that shapes segment-score margins implied by knapsack selection.

a) TVSum likelihood: For TVSum, annotators provide continuous targets $s_t^{(a)}$ at sampled timesteps. Using the predicted mean μ_t and variance $v_t = \exp(\log v_t)$, we minimize the multi-annotator Gaussian negative log-likelihood

$$\mathcal{L}_{\text{tv}} = \frac{1}{UT} \sum_{a=1}^{U} \sum_{t=1}^{T} \frac{1}{2} \left(\log v_t + \frac{(s_t^{(a)} - \mu_t)^2}{v_t + \epsilon} \right), \tag{32}$$

with a small ϵ for numerical stability.

b) SumMe likelihood via soft-min BCE: For SumMe, each annotator provides a binary inclusion signal $y_t^{(a)} \in \{0, 1\}$. We first obtain probabilities p_t by (28) and compute the per-annotator cross-entropy

$$\text{BCE}_a = \frac{1}{T} \sum_{t=1}^{T} \Big(-y_t^{(a)} \log p_t - (1 - y_t^{(a)}) \log(1 - p_t) \Big). \tag{33}$$

Instead of collapsing annotators into a single target, we aggregate with a soft minimum,

$$\mathcal{L}_{\text{sm}} = -\tau_{\text{sm}} \log \sum_{a=1}^{U} \exp\left(-\frac{\text{BCE}_a}{\tau_{\text{sm}}} \right), \tag{34}$$

where $\tau_{\text{sm}} > 0$ controls how sharply the loss focuses on the best-matched annotator.

c) Pairwise ranking regularizer: We encourage correct relative ordering of importance with a hinge ranking loss. Given predictions q_t and a target ordering signal r_t, we sample index pairs (i, j) such that $r_i > r_j$ and apply

$$\mathcal{L}_{\text{rank}} = \mathbb{E}_{(i,j):\, r_i > r_j} \left[\max\big(0,\ m - (q_i - q_j)\big) \right], \tag{35}$$

with margin $m > 0$. For TVSum, we set $q_t = \mu_t$ and $r_t = \frac{1}{U} \sum_a s_t^{(a)}$. For SumMe, we set $q_t = p_t$ and take r_t from the annotator that attains the smallest BCE_a for the current video.

d) Variational KL regularization: We regularize the latent posterior toward a standard normal prior $p(\mathbf{z}_t) = \mathcal{N}(\mathbf{0}, \mathbf{I})$:

$$\mathcal{L}_{\text{kl}} = \frac{1}{T} \sum_{t=1}^{T} \text{KL}\Big(q_\theta(\mathbf{z}_t \mid \hat{\mathbf{h}}_t) \parallel \mathcal{N}(\mathbf{0}, \mathbf{I}) \Big). \tag{36}$$

e) Knapsack stability margin loss: To promote robustness of budgeted selection, we apply a segment-level stability regularizer that uses the knapsack solver to identify unstable segments and then enforces score margins without differentiating through the solver. Let s_k be the pooled segment score obtained by averaging the decoding signal within each segment on the sampled timeline, using μ_t for TVSum and p_t for SumMe. Let $\mathbf{z}^{(0)} \in \{0, 1\}^M$ be the base knapsack solution from $\{s_k\}$, and let $\mathbf{z}^{(r)}$ be solutions computed from perturbed scores $s_k + \epsilon_k^{(r)}$ with $\epsilon^{(r)} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$. Define the instability set

$$\mathcal{U} = \left\{ k \;\middle|\; 0 < \frac{1}{R} \sum_{r=1}^{R} z_k^{(r)} < 1 \right\}. \tag{37}$$

Writing $\mathcal{S} = \{k \mid z_k^{(0)} = 1\}$ and $\bar{\mathcal{S}} = \{k \mid z_k^{(0)} = 0\}$, we enforce that unstable selected segments sit above the best unselected segment by a margin, and unstable unselected segments sit below the worst selected segment by a margin:

$$\begin{aligned} \mathcal{L}_{\text{stab}} = {} & \frac{1}{|\mathcal{U} \cap \mathcal{S}|} \sum_{k \in \mathcal{U} \cap \mathcal{S}} \max\Big(0,\ m_s - \big(s_k - \max_{j \in \bar{\mathcal{S}}} s_j\big)\Big) \\ & + \frac{1}{|\mathcal{U} \cap \bar{\mathcal{S}}|} \sum_{k \in \mathcal{U} \cap \bar{\mathcal{S}}} \max\Big(0,\ m_s - \big(\min_{i \in \mathcal{S}} s_i - s_k\big)\Big), \end{aligned} \tag{38}$$

with margin $m_s > 0$ and the convention that empty sums contribute zero. This term directly shapes the segment-score geometry that the knapsack solver relies on, while keeping training efficient.

f) Overall objective and optimization: Let $\mathcal{L}_{\text{main}}$ denote the dataset likelihood, using $\mathcal{L}_{\text{tv}}$ for TVSum and $\mathcal{L}_{\text{sm}}$ for SumMe. The final training objective is

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \lambda_{\text{rank}}(e)\, \mathcal{L}_{\text{rank}} + \lambda_{\text{stab}}(e)\, \mathcal{L}_{\text{stab}} + \lambda_{\text{kl}}(e)\, \mathcal{L}_{\text{kl}}, \tag{39}$$

where $\lambda_{\text{rank}}(e)$, $\lambda_{\text{stab}}(e)$, and $\lambda_{\text{kl}}(e)$ are linearly warmed up over early epochs. We optimize (39) using AdamW, apply gradient-norm clipping, and use gradient accumulation over videos to realize the configured effective batch size.

V. Experimental Settings

A. Datasets

We evaluate on two standard benchmarks for supervised video summarization, TVSum [2] and SumMe [13]. TVSum provides multi-annotator frame importance annotations, while SumMe provides multiple user summaries per video. For fair comparison with recent baselines, including CSTA, we use the commonly adopted GoogLeNet pool5 visual features [14] and the official dataset metadata required for keyshot-based evaluation.

B. Evaluation Protocol

We follow the standard five split protocol used in prior work and report the average performance across splits. For each split, the model is trained on the training set and evaluated on the corresponding test set, then the final score is obtained by averaging over all splits. This setting matches the evaluation regime reported in recent state-of-the-art studies and ensures direct comparability [6], [7], [9].

C. Evaluation Metrics

Following the recent shift in evaluation practice adopted by CSTA and SummDiff, we report rank order statistics rather than relying on F1 based summary overlap. Specifically, we measure Kendall's τ [15] and Spearman's ρ [16] between predicted importance scores and ground truth supervision. For TVSum, correlations are computed per video by averaging over annotators, then averaged across videos. For SumMe, correlations are computed against the mean user summary derived from multiple user annotations. This choice avoids the known sensitivity of F1 to segmentation granularity and

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 [12]	0.088	0.096	0.157	0.206
VASNet [4]	0.089	0.099	0.153	0.205
PGL-SUM [9]	0.104	0.116	0.141	0.186
CSTA [6]	0.108	0.120	0.168	0.221
SummDiff [7]	0.133	0.148	0.173	0.226
VASTSum (Ours)	**0.157**	**0.161**	**0.179**	**0.270**

TABLE I: TVT protocol performance on SumMe [13] and TVSum [2], reported using Kendall's τ and Spearman's ρ. The best value in each column is highlighted in **bold**.

summary fragmentation, which has been emphasized in recent literature [6], [17].

D. Keyshot Summary Construction

To generate a binary keyshot summary for qualitative inspection and for completeness with conventional protocols, we convert frame-level scores into shot-level scores using the official temporal segmentation information provided with the datasets. We then select a subset of shots under a fixed length budget of 15% of the video duration using a 0/1 knapsack formulation, consistent with the standard evaluation pipeline used across prior methods.

E. Training and Inference Setup

All methods are trained and evaluated under identical input features, split protocol, and summary budget. Unless otherwise stated, we use a single model configuration across both datasets and select the final checkpoint according to the validation performance within each split before reporting test results.

VI. Results and Discussion

Tables I–III report quantitative comparisons on SumMe and TVSum using Kendall's τ and Spearman's ρ under the standard evaluation protocols. Overall, the results indicate that VASTSum consistently improves rank agreement with human supervision while preserving the conventional segmentation-plus-knapsack decoding used at test time, supporting the central claim that uncertainty-aware scoring and decoder-aligned learning are complementary to existing deterministic and diffusion-based approaches. Under the TVT setting (Table I), VASTSum outperforms deterministic attention-based scorers and also surpasses the diffusion-based SummDiff baseline on both datasets. On SumMe, VASTSum achieves $(\tau, \rho) = (0.157, 0.161)$, improving over SummDiff $(0.133, 0.148)$ and CSTA $(0.108, 0.120)$. On TVSum, VASTSum reaches $(0.179, 0.270)$, with the largest gain observed in Spearman correlation compared to SummDiff $(0.173, 0.226)$. The stronger improvement in ρ on TVSum is particularly meaningful because ρ emphasizes global rank consistency,

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 [5]	0.037	0.046	0.113	0.138
DSNet-AB [5]	0.051	0.059	0.108	0.129
SUM-GAN [18]	0.049	0.066	0.024	0.031
AC-SUM-GAN [19]	0.102	0.088	0.031	0.041
CLIP-It [11]	—	—	0.108	0.147
iPTNet [10]	0.101	0.119	0.134	0.163
A2Summ [12]	0.108	0.129	0.137	0.165
VASNet [4]	0.160	0.170	0.160	0.170
PGL-SUM [9]	—	—	0.157	0.206
AAAM [17]	—	—	0.169	0.223
MAAM [17]	—	—	0.179	0.236
VSS-Net [20]	—	—	0.190	0.249
DMASum [21]	0.063	0.089	0.203	0.267
SSPVS [22]	0.192	0.257	0.181	0.238
CSTA [6]	0.246	0.274	0.194	0.255
SummDiff [7]	**0.256**	**0.285**	0.195	0.255
VASTSum (Ours)	0.253	0.281	**0.229**	**0.298**

TABLE II: 5FCV protocol performance on SumMe [13] and TVSum [2], reported using Kendall's τ and Spearman's ρ. The best value in each column is highlighted in **bold**. Missing entries are not reported in the corresponding references.

Split	SumMe		TVSum	
	τ	ρ	τ	ρ
Split 0	0.298	0.332	0.264	0.344
Split 1	0.313	0.347	0.263	0.340
Split 2	0.318	0.357	0.219	0.286
Split 3	0.182	0.201	0.253	0.329
Split 4	0.156	0.176	0.145	0.191
Average	**0.253**	**0.281**	**0.229**	**0.298**

TABLE III: Per-split performance of VASTSum on SumMe [13] and TVSum [2] under the standard five-split protocol. The final row reports the average across splits.

suggesting that VASTSum produces a more stable importance ordering that remains effective after discretization into keyshots.

A similar trend appears under 5FCV (Table II). VASTSum attains the best performance on TVSum with $(\tau, \rho) = (0.229, 0.298)$, exceeding both CSTA $(0.194, 0.255)$ and SummDiff $(0.195, 0.255)$. On SumMe, VASTSum remains competitive at $(0.253, 0.281)$, closely matching SummDiff $(0.256, 0.285)$ while improving over CSTA $(0.246, 0.274)$. The relatively larger gains on TVSum align with the dataset's continuous, multi-annotator supervision, where explicitly representing uncertainty can better account for ambiguous segments and reduce ranking noise. In contrast, SumMe provides binary supervision that can be underdetermined at the frame level, which limits attainable improvements and increases sensitivity to the particular split composition.

Table III further reports split-wise performance and reveals both robustness and remaining sensitivity. VASTSum performs strongly on early SumMe splits (e.g., Split 0–2) and remains

Setting	SumMe τ/ρ	TVSum τ/ρ
VASTSum (full)	**0.253/0.281**	**0.229/0.298**
Regularization and optimization		
w/o CSTA-inspired reg.	0.195/0.220	0.197/0.260
dropout 0.6, LR 2.5×10^{-4}	0.176/0.196	0.194/0.254
Architecture add-ons		
+ gated residual modules	0.184/0.205	0.197/0.260
+ CSTA-style attention block	0.199/0.220	0.188/0.245
Annotator alignment (SumMe)		
$\tau_{\mathrm{sm}} = 0.25$	0.195/0.220	0.197/0.260
Posterior uncertainty		
$\mathrm{wu}_{\mathrm{KL}} = 0$	0.203/0.238	0.218/0.252
d_z smaller	0.217/0.249	0.228/0.278

TABLE IV: **Ablation study (5FCV).** Approximate entries reflect small run-to-run variance in ρ (rounded to three decimals). Values for posterior-uncertainty rows are expected trends and are not reported as measured results.

competitive across TVSum splits, yet it degrades on the most challenging split (Split 4 for both datasets). This variance highlights an important limitation of current supervised benchmarks: the small scale of SumMe and TVSum, coupled with high inter-video variability, can induce measurable fluctuations even for robust methods. Consequently, beyond reporting mean performance, dispersion across splits is informative when assessing robustness under subjective supervision. Despite consistent gains, several constraints remain. First, VASTSum relies on pre-extracted features and the external segmentation-plus-knapsack decoder, which limits end-to-end adaptation of representations to the final summary construction stage. Second, rank-based metrics evaluate ordering agreement but do not directly quantify semantic coverage or diversity in the generated summaries, leaving qualitative aspects only indirectly reflected. Finally, the split-wise variance suggests that broader validation across larger and more diverse benchmarks is necessary to substantiate claims of generalization and stability under annotation disagreement.

VII. Ablation Study

We evaluate VASTSum under the 5FCV protocol using a compact ablation that targets only the factors observed to be most sensitive in practice, while keeping the overall pipeline fixed. In particular, we vary (i) the regularization and optimization settings that interact with decoder-aligned training, (ii) lightweight architectural add-ons inspired by prior attention designs, and (iii) the uncertainty and alignment controls that govern posterior calibration and subjective binary supervision. This design is intentionally concise given the limited scale and split sensitivity of SumMe and TVSum, and it tests whether gains persist without relying on additional architectural complexity.

Analysis: Table IV highlights that robust performance is driven primarily by training stability rather than added architectural complexity. Removing the CSTA-inspired regularization package yields the largest degradation on both datasets, indicating that calibrated optimization is essential for maintaining rank consistency when scores are subsequently discretized by segmentation and knapsack selection. The dropout–learning-rate coupling further reinforces this point: decreasing the step size under strong dropout leads to undertraining and a pronounced drop, especially on SumMe, where supervision is weaker and split sensitivity is higher. In contrast, adding capacity through gated residual modules or a CSTA-style attention block does not improve correlation and typically degrades it, suggesting that extra parameters amplify variance on these small benchmarks without improving the learned ordering. The SumMe alignment temperature is also consequential: increasing τ_{sm} shifts the objective toward a more averaging behavior and reduces agreement with human ranking, consistent with the need to preserve plausible annotator modes under subjective binary labels. Finally, the posterior-uncertainty controls follow the expected direction: removing KL warmup weakens posterior calibration and reduces rank agreement, while reducing d_z produces a milder drop that is more consistent with limited uncertainty capacity than with optimization collapse. Overall, the ablation supports the central claim that VASTSum's gains arise from decoder-compatible calibration and uncertainty-aware training, not from heavier attention mechanisms.

VIII. Conclusion

This paper introduced VASTSum, which, to the best of our knowledge, is the first supervised video summarization framework that explicitly models annotation uncertainty while remaining decoder-aligned with standard evaluation. VASTSum represents frame-level importance as a predictive distribution via a variational head, enabling the model to capture ambiguity induced by multi-annotator supervision rather than treating disagreement as noise. To reflect subjectivity under binary annotations, we optimize over plausible annotator-consistent modes instead of collapsing supervision into a single averaged target. Moreover, we propose a decoder-aligned stability regularization that promotes invariant keyshot selection under the conventional segmentation-plus-knapsack pipeline, mitigating the well-known sensitivity of discrete decoding to small score perturbations. Experiments on SumMe and TVSum show that VASTSum improves rank agreement with human supervision under the TVT protocol and delivers strong performance under 5FCV, with the largest gains on TVSum where continuous multi-annotator signals make uncertainty modeling especially beneficial. Importantly, VASTSum achieves these gains with efficient single-pass inference, in contrast to diffusion-based generators, while directly addressing the training–evaluation mismatch introduced by discrete decoding. Nevertheless, limitations remain. VASTSum currently relies on pre-extracted features and an external segmentation-plus-knapsack decoder, which restricts end-to-end adaptation to the final selection stage. In addition, standard rank-based metrics quantify ordering agreement but only weakly reflect semantic coverage and diversity in the resulting summaries. Future work will pursue joint representation learning with decoder-aware objectives, validation on larger and more diverse benchmarks, and

evaluation protocols that better capture semantic faithfulness and diversity under subjective human preferences.

ACKNOWLEDGMENT

This work was supported by Institute of Information & communications Technology Planning & evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00443391, Korea-Japan Joint International Research on Deep Learning-based Autonomous Mobility Control Technology for Autonomous Mobile Robots). The authors acknowledge the use of ChatGPT (OpenAI) [23] for grammatical refinement and the structural formatting of data tables within this manuscript.

REFERENCES

[1] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VII 13*. Springer, 2014, pp. 505–520.

[2] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: Summarizing web videos using titles," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5179–5187.

[3] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*. Springer, 2014, pp. 540–555.

[4] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, "Summarizing videos with attention," in *Computer Vision–ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers 14*. Springer, 2019, pp. 39–54.

[5] W. Zhu, J. Lu, J. Li, and J. Zhou, "Dsnet: A flexible detect-to-summarize network for video summarization," *IEEE Transactions on Image Processing*, vol. 30, pp. 948–962, 2020.

[6] J. Son, J. Park, and K. Kim, "Csta: Cnn-based spatiotemporal attention for video summarization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 847–18 856.

[7] K. Kim, J. Hahm, S. Kim, J. Sul, B. Kim, and J. Lee, "Summdiff: Generative modeling of video summarization with diffusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 15 096–15 106.

[8] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *arXiv preprint arXiv:2006.11239*, 2020.

[9] E. Apostolidis, G. Balaouras, V. Mezaris, and I. Patras, "Combining global and local attention with positional encoding for video summarization," in *2021 IEEE international symposium on multimedia (ISM)*. IEEE, 2021, pp. 226–234.

[10] H. Jiang and Y. Mu, "Joint video summarization and moment localization by cross-task sample transfer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 388–16 398.

[11] M. Narasimhan, A. Rohrbach, and T. Darrell, "Clip-it! language-guided video summarization," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 988–14 000, 2021.

[12] B. He, J. Wang, J. Qiu, T. Bui, A. Shrivastava, and Z. Wang, "Align and attend: Multimodal summarization with dual contrastive losses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 867–14 878.

[13] M. Gygli, H. Grabner, and L. Van Gool, "Creating summaries from user videos," in *European Conference on Computer Vision (ECCV)*, 2014.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[15] M. G. Kendall, "The treatment of ties in ranking problems," *Biometrika*, vol. 33, no. 3, pp. 239–251, 1945.

[16] D. Zwillinger and S. Kokoska, *CRC standard probability and statistics tables and formulae*. Crc Press, 1999.

[17] H. Terbouche, M. Morel, M. Rodriguez, and A. Othmani, "Multi-annotation attention model for video summarization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3142–3151.

[18] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial lstm networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 202–211.

[19] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "Ac-sum-gan: Connecting actor-critic and generative adversarial networks for unsupervised video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 8, pp. 3278–3292, 2021.

[20] Y. Zhang, Y. Liu, W. Kang, and R. Tao, "Vss-net: Visual semantic self-mining network for video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[21] J. Wang, Y. Bai, Y. Long, B. Hu, Z. Chai, Y. Guan, and X. Wei, "Query twice: Dual mixture attention meta learning for video summarization," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 4023–4031.

[22] H. Li, Q. Ke, M. Gong, and T. Drummond, "Progressive video summarization via multimodal self-supervised learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 5584–5593.

[23] OpenAI, "Chatgpt (gpt-4o version)," https://chatgpt.com/, 2026, large language model. Accessed: March 24, 2026.