MAMS: MODEL-AGNOSTIC MODULE SELECTION FRAMEWORK FOR VIDEO CAPTIONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-modal transformers are rapidly gaining attention in video captioning tasks. Existing multi-modal methods extract a fixed number of frames, but this has a few critical challenges. If a limited number of frames is extracted, it is challenging to retrieve sufficient information for caption generation. Conversely, extracting an excessive number of frames can lead to the frames containing redundant information. We refer the aforementioned challenges as information loss and excessive information similarity, respectively. This paper proposes the new model-agnostic module selection framework that can choose a module with an appropriate size through the flow selector and token selector. The proposed framework can select an appropriate size of features for each video data during training and inference. Using this framework, we moderate the issues of information loss and excessive information similarity that arise from extracting a fixed number of frames. In addition, we further moderate the excessive information similarity issue in each flow by adding diversity promoting losses. Our numerical experiments with two different datasets demonstrate that the proposed framework significantly improves the performances of three different existing representative/state-of-the-art video captioning models.

1 INTRODUCTION

The video captioning task generates a description for a provided video in natural language (Li et al., 2021b; Wang et al., 2019). To improve video captioning performances, it is pivotal to introduce multi-modal transformers (Sun et al., 2019). Many recent studies extract an identical number of frames regardless of video, to use a consistent input size for transformer-based models (Chen et al., 2023; Yang et al., 2023). Selecting a fixed number of frames can be divided into sparse and dense sampling methods, and each approach has its own limitations.

The sparse sampling approach extracts a small number of frames in a random manner or with some criterion (Fu et al., 2021; 2023; Wang et al., 2022). For videos with abundant information, if a sparse sampling method extracts a limited number of frames, caption generation performances can degrade (Lin et al., 2022). We refer to this issue as the information loss limitation. In particular, CLIPBERT (Lei et al., 2021) argues that extracting numerous frames from videos can cause excessive information similarity issues, and dense sampling is not essential for the visual language task. Nevertheless, its applicability to the video captioning task remains questionable. In video captioning, recent high-performance models predominantly adopt the dense sampling approach. (Kuo et al., 2023; Xu et al., 2023) The dense sampling approach extracts many frames. For example, SWINBERT (Lin et al., 2022) shows that extracting more frames from videos yielded enhanced results compared to prior sparse sampled approaches. However, for videos with little information (e.g., video with little dynamics), the dense sampling approach could extract similar frames, i.e., superfluous information. We

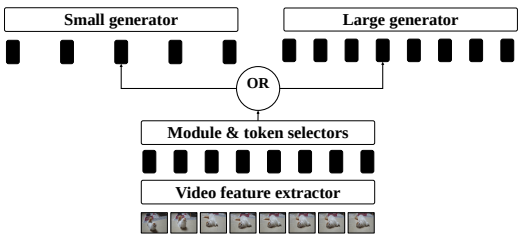

Figure 1: Overview of our framework

refer to this issue as excessive information similarity. To address the excessive information similarity issue, SWINBERT uses a learnable sparse attention mask. Yet, there exists a room for improving video captioning performances with different approaches. Recent video captioning models in both approaches extract a fixed number of frames for *all* videos. Extracting a fixed number of frames can cause information loss when sparsely sampling frames from videos with rich information, and lead to the excessive similarity issue when densely sampling frames from videos with poor information.

This paper proposes the new **Model-Agnostic Module Selection framework (MAMS)** that is applicable to existing captioning models. It diversifies the size/complexity of existing models and chooses a module with appropriate size. The contributions of the paper are summarized as follows:

- The proposed framework selects a module with an appropriate size, among caption generation models with different sizes. See the overview of the proposed framework in Figure 1(c). We first extract a substantial amount of video features and then construct a subset/subsets using a token selector/token selectors. We process each subset of features in the corresponding module. In addition, we determine which module is appropriate using a module selector(s). Different from existing models that extract a fixed number of frames, the proposed framework selects/uses an appropriate number of video features. Consequently, it can moderate the aforementioned information loss and excessive information similarity limitations in existing methods.
- To better address the excessive information similarity issue, we introduce **diversity promoting losses** for each module.
- We applied the proposed framework to three representative/state-of-the-art models: SwinBERT (Lin et al., 2022), UniVL (Luo et al., 2020), and mPLUG-2 (Xu et al., 2023). Our numerical experiments with the MSVD, MSRVTT and YOUCOOKII datasets show that the proposed framework significantly improves all the existing models. In particular, we achieved a new state of the arts benchmark by applying the proposed method to the mPLUG-2 model that is the current state-of-the-art in the MSVD and MSRVTT datasets.

2 Related Work

2.1 Video Captioning

The early approach in video captioning research was rule-based, directly extracting subjects, verbs, and objects to construct sentences (Das et al., 2013; Kojima et al., 2002). subsequent methods involved extracting sentences on a frame-by-frame basis and combining them (Bahdanau et al., 2014; Sutskever et al., 2014). In recent research, the paradigm has shifted to consist of a feature extractor and a generation module (Arnab et al., 2021). This paradigm initially began by using fixed video feature embeddings to generate sentences (Aafaq et al., 2019; Pan et al., 2020; Pei et al., 2019; Shi et al., 2020). Due to the different lengths of embeddings generated for each video, this approach employed masking embeddings to align the input dimensions (Luo et al., 2020). This paradigm has shifted from using fixed embeddings to an end-to-end approach in video extractors, extracting features to enhance the captioning model. Recently, many studies have adopted these methodologies (Chen et al., 2019; Li et al., 2021b; Liu et al., 2018; Zhang et al., 2021). we categorize end-to-end methodologies into two primary approaches. The first approach uses sparse sampling, selecting a limited number of frames from the video (Fu et al., 2021; 2023; Wang et al., 2022). However, this method might risk losing information due to the small sample size. The second approach applies dense sampling, extracting more frames from the video (Kuo et al., 2023; Xu et al., 2023; Lin et al., 2022). This method, though, might face frame similarity issues. In this paper, we propose a method that overcomes the limitations of these traditional methods.

2.2 Video transformer

Currently, the transformer architecture has demonstrated exceptional performance across various research domains. Starting with its adoption in the image field through models like ViT and Swin-Transformer, recently, video-based Transformer models such as ViViT, TimeSformer, and VidSwin have consistently shown outstanding results in the video domain (Dosovitskiy et al., 2010; Liu et al., 2021; Arnab et al., 2021; Bertasius et al., 2021; Liu et al., 2022). Recent top-performing models

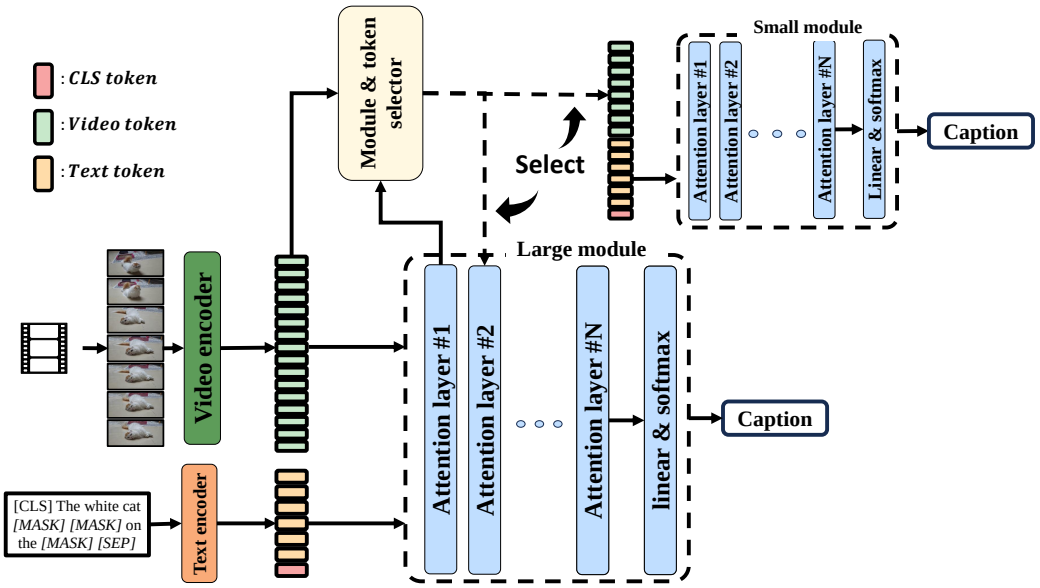

Figure 2: Overall architecture of MAMS

in video captioning incorporate a multi-modal structure that combines video transformers with text transformers. These models extract embeddings from text using text transformers, notably BERT and derive embeddings from video using the previously mentioned video transformers like ViT or VidSwin. We then feed these extracted embeddings into another multi-modal transformer for further learning. Most research has shifted towards employing the transformer architecture, which requires fixed-length inputs. Consequently, video captioning models are adopting methods to extract frames of the same fixed length from different videos (Devlin et al., 2018; Li et al., 2021a; 2022b;a).

3 METHOD

3.1 THE OVERALL ARCHITECTURE OF MAMS

In video captioning, the most popular architecture based on multi-modal transformers consists of three major modules: *1)* a video encoder that transforms a video to a token vector; *2)* a text encoder that transforms a set of texts to a token vector; and *3)* a caption generator that creates captions. In a nutshell, the proposed MAMS framework adds smaller caption generation module(s) to the aforementioned architecture in parallel. By default, we add a smaller generation module compared to the one in existing method and refer to two generation modules as *large* and *small* generation models. We summarize the important features of MAMS below:

1. We calculate significance scores for all video tokens and frames. See 'Significance scores calculator' in Figure 4.
2. We select either a large or small generation module by using calculated significance scores. See 'Module selector' in Figure 4. If a large module is selected, we use all video tokens. As the input to a large generation module, we concatenate video and text tokens. See 'Large generation module' in Figure 4. If a small module is selected, we select only important video tokens and use them as an input. See 'Small generation module' in Figure 4. We apply the module selection process both in training and inference.

3.2 SIGNIFICANCE SCORES CALCULATION

In video captioning models, a video encoder transforms frames into video tokens. A generation module then takes these video tokens to produce sentences. As adjacent frames are similar, it is

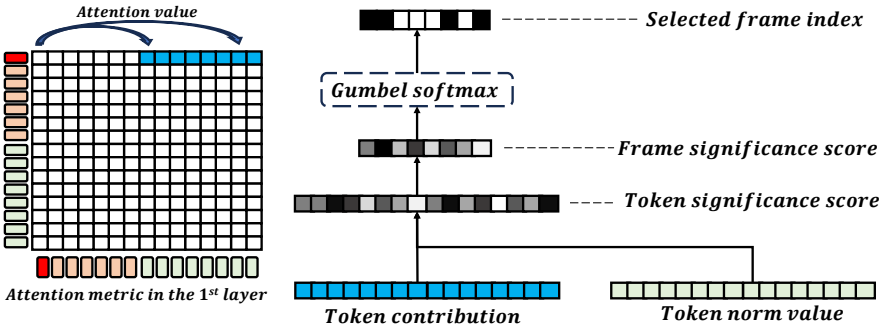

Figure 3: Architecture of the proposed module and token selector

natural that their video tokens possess similar values. We assume, however, that their contributions to caption generation are different.

To quantify the contribution of each token to caption generation, we define the token significance score inspired by (Cao et al., 2023). Specifically, we define the pth token significance score at the ith frame as follows:

$$t_{i,p} = \frac{a_{i,p} \cdot \|\mathbf{x}^{\text{v}}_{i,p}\|}{\sum_{i,p} a_{i,p} \cdot \|\mathbf{x}^{\text{v}}_{i,p}\|}, \quad i = 1, \ldots, T,\ p = 1, \ldots, P, \tag{1}$$

where $a_{i,p}$ denotes the attention value between the special classification (CLS) token and the pth video token at the ith frame, $\mathbf{x}^{\text{v}}_{i,p}$ denotes the pth video token at the ith frame, T is the number of total frames, and P is the number of tokens per each frame. We calculate $\{a_{i,p} : \forall i, p\}$ from the first attention layer of a generation module. Considering the CLS token as representing the starting point of the caption, the attention values between the CLS token and a video token can quantify the contribution of video tokens to the entire caption (Cao et al., 2023). In (1), we additionally assume that not only the attention values but also video tokens themselves influence caption generation, and use the norm value of each token in computing the token significance score.

Using calculated $\{t_{i,p} : \forall i, p\}$ in (1), we define the significance score for the ith frame as follows:

$$f_i = \sum_{p=1}^{P} t_{i,p}, \quad i = 1, \ldots, T, \tag{2}$$

where by default, we consider that a video encoder generates multiple tokens from a single frame. Note that calculating the proposed scores in (1)–(2) does *not* require ground-truth labels. Calculating the defined quantities (1)–(2), we use them to select important tokens and frames for module selection. Depending on the ratio of numbers of frames and tokens in video encoding, we modify (2).

We construct a set of indices of important frames, S^{frm}, and then a set of indices of important tokens, S^{tk}. In constructing S^{frm} to select the most important T' frames, we use the gumbel-softmax (Jang et al., 2016) operator to $\{f_i : i = 1, \ldots, T\}$ T' times. (We later describe details of the proposed algorithm using gumbel-softmax in Section 4.4.) In constructing S^{tk}, we use all video token indices from selected important frames with S^{frm}, i.e., $S^{\text{tk}} = \{(i, p) : i \in S^{\text{frm}}, p = 1, \ldots, P\}$.

In the next section, for module selection, we use S^{frm} constructed by using (2). If a small module is selected, we select important tokens based on S^{tk} and use them as its input.

3.3 MODULE SELECTION

3.3.1 INFERENCE

Using the frame significance scores for T frames, $\{f_i : i = 1, \ldots, T\}$, and S^{frm} with $|S^{\text{frm}}| = T'$, we evaluate the overall significance of T' selected frames by $\tilde{f} = \sum_{i \in S^{\text{frm}}} f_i$, and select a caption generation module with an appropriate size using $\tilde{f}$. A higher $\tilde{f}$ value implies that selected important frames have more "information."

In the default setup that selects between small and large generation modules, we use the following selection rule using $\tilde{f}$ defined above:

$$\begin{cases} \text{We select a small module,} & \text{if} \tilde{f} > \lambda, \\ \text{We select a large module,} & \text{otherwise,} \end{cases} \tag{3}$$

where the decision threshold λ is defined by $\lambda = T'/T + \epsilon$. The decision rule in (3) implies the following. The decision criterion $\tilde{f} > \lambda$ implies that selected important frames have sufficient information, so we select a small module. Conversely, the condition $\tilde{f} \leq \lambda$ implies that selected frames have insufficient information, so we select a large module and use all frames.

After a module is selected, we select video tokens as follows:

- If a large module is selected, we use all the video tokens $\{\mathbf{x}^{\text{v}}_{i,p} : \forall i, p\}$.
- If a small module is selected, we select only important video tokens $\{\mathbf{x}^{\text{v}}_{i,p} : \forall (i, p) \in S^{\text{tk}}\}$.

We construct the input to either model by concatenating selected video tokes above with text tokens.

3.3.2 TRAINING

We train both large and small modules using the loss function in the following format:

$$\lambda_{\text{large}} \cdot \mathcal{L}_{\text{large}} + \lambda_{\text{small}} \cdot \mathcal{L}_{\text{small}}, \quad (\lambda_{\text{large}}, \lambda_{\text{small}}) = \begin{cases} (1, 0), & \text{if } \tilde{f} \leq \lambda, \\ (0, 1), & \text{if } \tilde{f} > \lambda, \end{cases} \tag{4}$$

where $\mathcal{L}_{\text{large}}$ and $\mathcal{L}_{\text{small}}$ are losses for training a large and small generation model, respectively. By setting the module selection weighting parameters $(\lambda_{\text{large}}, \lambda_{\text{small}})$ as in (4), we nullify either small or large module training, ensuring that meaningful back propagation flows through only one module.

3.4 DIVERSITY PROMOTING LOSS

As the number of generation module choices is limited compared to the diversity of data, the proposed module selection approach itself may have a limitation in mitigating the excessive information similarity issue. To further moderate the excessive similarity issue, we proposed a new loss function inspired by (Chen et al., 2022; Gong et al., 2021). We consider that transformer-based models often use several attention layers, and tokens in different layers are with the same dimension.

First, we propose a new loss term that can promote the diversity between output tokens from the last attention layer:

$$\mathcal{L}_{\text{within}} = -\frac{1}{TP} \sum_{i=1}^{T} \sum_{j=1}^{P} \log \frac{\exp(\tilde{\mathbf{x}}_{i,j}^{\top} \tilde{\mathbf{x}}_{i,j})}{\exp(\tilde{\mathbf{x}}_{i,j}^{\top} \tilde{\mathbf{x}}_{i,j}) + \exp(\tilde{\mathbf{x}}_{i,j}^{\top} (\frac{1}{TP-1} \sum_{(i,j) \neq (k,l)} \tilde{\mathbf{x}}_{k,l})}, \tag{5}$$

where $\tilde{\mathbf{x}}_{i,j}$ is the jth token at the ith frame and $\{\tilde{\mathbf{x}}_{i,j} : \forall i, j\}$ are output tokens from the last attention layer. Second, we proposed a new loss term that can promote the diversity between input tokens to the first attention layer and output tokens from the last attention layer:

$$\mathcal{L}_{\text{cross}} = -\frac{1}{TP}\sum_{i=1}^{T}\sum_{j=1}^{P}\log\frac{\exp(\tilde{\mathbf{x}}_{i,j}^{\top}\mathbf{x}_{i,j}^{\text{v}})}{\exp(\tilde{\mathbf{x}}_{i,j}^{\top}\mathbf{x}_{i,j}^{\text{v}}) + \exp(\tilde{\mathbf{x}}_{i,j}^{\top}(\frac{1}{TP-1}\sum_{(i,j)\neq(k,l)}\mathbf{x}_{k,l}^{\text{v}})}. \tag{6}$$

The video tokens extracted from the video encoder undergo positional encoding and a multi-layer perceptron layer for dimension alignment, before entering the first attention layer.

The below is the proposed caption generation loss function that can promote the diversity between video tokens:

$$\mathcal{L} = \mathcal{L}_{\text{cap}} + \lambda_{\text{within}} \cdot \mathcal{L}_{\text{within}} + \lambda_{\text{cross}} \cdot \mathcal{L}_{\text{cross}}, \tag{7}$$

where $\mathcal{L}_{\text{cap}}$ is a conventional caption generation loss used in existing models (e.g., in SwinBERT, the combination of losses in (Lin et al., 2022; Devlin et al., 2018)), and λ_{within} and λ_{cross} are balancing parameters of different loss terms.

In training a large model, we use (5)–(6) as they are; in training a small model, we replace T with T' in (5)–(6). We incorporate these into the training form (4).

4 Experimental results and discussion

4.1 Experimental setup

We incorporated the following models into the proposed MAMS framework:

- Two representative models, SwinBERT and UniVL;
- The state-of-the-art model, mPLUG-2.

4.1.1 Datasets

We conduct experiments with three different video captioning datasets. The Microsoft research video description (MSVD) dataset (Chen & Dolan, 2011) is a widely used benchmark in video captioning that has over 2,000 short video clips, each paired with an average of 40 human-annotated textual descriptions. The Microsoft research video-to-text (MSRVTT) dataset (Xu et al., 2016) is another representative benchmark in video captioning that features a diverse collection of over 10,000 video clips, and each matched with approximately 20 human-generated textual descriptions. The YOUCOOKII (Zhou et al., 2018) dataset is the largest task-oriented video dataset in the vision community. It contains 2,000 long untrimmed videos from 89 cooking recipes. On average, each distinct recipe has 22 videos.

4.1.2 evaluation metric

The bilingual evaluation understudy (BLEU) metric (Papineni et al., 2002) (Bilingual Evaluation Understudy) measures the extent to which machine-generated sentences match human references, focusing on precision and n-gram overlap. The metric for evaluation of Translation with explicit ordering (METEOR) (Banerjee & Lavie, 2005) evaluates sentence quality by considering synonymy, stemming, and word order alignment in machine-generated text. The recall-oriented understudy for gisting evaluation (ROUGE) metric (Lin & Och, 2004) assesses sentences by measuring shared n-grams, common subsequences, and F1-scores, providing insight into content overlap. The consensus-based image description evaluation (CIDEr) metric (Vedantam et al., 2015) evaluates the sentence diversity and quality based on consensus judgments, capturing the richness and relevance of machine-generated video descriptions.

4.1.3 Implementation details

We implemented the proposed framework using the PyTorch environment (Paszke et al., 2019) and used NVIDIA RTX A100 GPUs for the experiments. For comparisons with the baseline models, we kept all settings (e.g., such as batch size and epochs) consistent, except for learning rate. We incorporated the MAMS methodology with minor variations, as each model is implemented in a

slightly different setup. We provide detailed implementation specifics for each model in Appendix A.2.

4.2 VIDEO CAPTIONING PERFORMANCE COMPARISONS BETWEEN DIFFERENT MODELS

Models	MSVD				MSRVTT			
	B4	M	R	C	B4	M	R	C
*EMCL-Net (Jin et al., 2022)	-	-	-	-	45.3	30.2	63.2	54.6
*CLIP-DCD (Yang et al., 2022)	-	-	-	-	48.2	31.3	64.8	58.7
*TextKG Gu et al. (2023)	60.8	38.5	75.1	105.2	43.7	29.6	62.4	52.4
*CoCap (Shen et al., 2023)	60.1	41.4	78.2	121.5	44.4	30.3	63.4	57.2
*VIOLETv2 (Fu et al., 2023)	-	-	-	139.2	-	-	-	58
SwinBERT (Lin et al., 2022)	58.2	41.3	77.5	120.6	41.9	29.9	62.1	53.8
SwinBERT + MAMS	62.8	41.9	79	127.3	43.8	29.5	62.9	55.2
	(+4.6)	(+0.6)	(+1.5)	(+6.7)	(+1.9)	(-0.4)	(+0.8)	(+1.4)
mPLUG-2 (Xu et al., 2023)	75.0	48.4	85.3	165.8	57.9	**34.9**	70.1	80.3
mPLUG-2 + MAMS	**80.5**	**48.7**	**87.9**	**176.1**	**60.0**	34.7	**71.2**	**82.9**
	(+5.5)	(+0.3)	(+2.6)	(+10.3)	(+2.1)	(-0.2)	(+1.1)	(+2.6)

Table 1: **Result of MAMS Applied to SwinBERT and mPLUG-2 on MSVD & MSRVTT Dataset** We have selected EMCL-Net, CLIP-DCD, TextKG, CoCap, and VIOLETv2 as the benchmark models for comparison. These state-of-the-art video captioning models are trained exclusively on the captioning data from MSVD and MSRVTT without additional data. The * indicates that the results are taken from the paper. We denote B4 as BLEU4, M as METEOR, R as ROUGE, and C as CIDEr.

This section presents the results of applying the MAMS methodology to existing video captioning models that have available official implementations. Tables 1–2 show the proposed MAMS framework significantly improves the video captioning performances of the existing models across all three datasets. Notably, in experiments with the MSVD dataset, proposed MAMS improved the BLEU and CIDEr metrics by 4.6 and 6.7, respectively. In experiments with the MRVTT dataset, MAMS improved the BLEU and CIDEr metrics by 1.7 and 1.4, respectively. On the Youcook2 dataset, the improvement of using MAMS was 3.5 and 7.7 in BLEU and CIDEr, respectively. The mPLUG model is the existing state-of-the-art benchmark in the MSVD dataset without additional training data (excluding MaMMUTs of which official code is unavailable) and in the MSRVTT. Incorporating mPLUG into the MAMS framework led to significant improvements. For the MSVD dataset, the BLEU and CIDEr metric improvements were 4.5 and 10.3, respectively. For the MRVTT dataset, the incorporation improved the BLEU and CIDEr metrics by 2.1 and 2.6, respectively. (The mPLUG model shows high performances with the modified training method discussed in Section 4.6.) For the UniVL model that is a baseline for the YOUCOOK dataset, using the MOMO framework led to 3.2 and 6.2 improvements in BLEU and CIDEr, respectively.

Models	YouCookII			
	B4	M	R	C
SwinBERT	9.0	15.6	37.3	109.0
SwinBERT + MAMS	**12.5**	**15.9**	**40.8**	**116.7**
UniVL Luo et al. (2020)	11.2	17.6	40.1	127.0
UniVL + MAMS	**14.4**	**17.8**	**44.3**	**133.2**

Table 2: Result of MAMS Applied to SwinBERT and UniVL on YOUCOOKII Dataset

4.3 ABLATION STUDY FOR DIVERSITY PROMOTING LOSS IN PROPOSED MAMS

As previously mentioned, extracting frames with a fixed number from every video leads to two major issues: information loss and excessive information similarity. The MAMS framework moderated

Models	LOSS			MSVD				MSRVTT			
	L_{cap}	L_{within}	L_{cross}	B4	M	R	C	B4	M	R	C
SwinBERT	✓	×	×	58.2	41.3	77.5	120.6	41.9	29.9	62.1	53.8
SwinBERT + MAMS	✓	×	×	62.6	41.7	78.0	125.6	42.7	29.4	62.5	54.4
SwinBERT + MAMS	✓	✓	×	62.3	**42.1**	78.8	126.0	42.6	**30.1**	62.7	54.7
SwinBERT + MAMS	✓	×	✓	**62.8**	41.6	**79.0**	126.7	43.0	29.3	62.8	54.7
SwinBERT + MAMS	✓	✓	✓	**62.8**	41.9	**79.0**	**127.3**	**43.8**	29.5	**62.9**	**55.2**

Table 3: Comparisons of MAMS framework variations with different diversity promoting losses for SwinBERT

Models	MSVD				MSRVTT			
	B4	M	R	C	B4	M	R	C
SwinBERT	58.2	41.3	77.5	120.6	41.9	**29.9**	62.1	53.8
SwinBERT + MAMS (gradient-free)	**63.2**	41.4	78.9	124.2	43.2	29.5	62.6	54.9
SwinBERT + MAMS (gradient-based)	62.8	**41.9**	**79.0**	**127.3**	**43.8**	29.5	**62.9**	**55.2**

Table 4: Impact of gradient flow in Gumbel algorithm on SwinBERT

these challenges and significantly improved the video captioning performances, by using conventional models. Table 3 shows the significant performance improvements when using the MAMS framework, as compared to the original SwinBERT model. To further moderate the issue of excessive information similarity, we introduced two diversity promoting losses in Section 3.4: L_{within} and L_{cross}. This approach can enhance the diversity in video tokens, further moderating the excessive information similarity issue. Table 3 shows that employing L_{within} and L_{cross} independently or in conjunction can achieve significant performance gains.

4.4 GUMBEL ALGORITHM IN MODULE SELECTOR

In the token selector within the MAMS methodology, there is an operation to extract the indices of (T') important frames out of (T) frames. Conventionally, methodologies commonly use the argmax functions or the top-T' algorithm for hard selection tasks requiring index extraction (Yamazaki et al., 2023; Seo et al., 2022; Shi et al., 2019). However, these conventional techniques pose a challenge during training, as they do not allow gradients to flow. To address these limitations, many recent studies (Tan et al., 2020; Dai et al., 2022; Cao et al., 2023) are adopting the Gumbel-Softmax algorithm. We utilize the Gumbel-Softmax trick to allow gradient flow while selecting important frame indices. Our algorithm, distinct from the Gumbel algorithms used in many other studies, implements a Gumbel algorithm through non-replacement. Explanations regarding non-replacement and replacement extraction and the algorithm for the module & token selector utilizing the Gumbel-Softmax are provided along with the pseudocode in Appendix A.3. In Table 4, 'gradient-free' refers to the results of experiments conducted in a setting where backpropagation is not allowed through the Gumbel-Softmax, while 'gradient-based' indicates a setting where backpropagation is enabled. From the results, it is evident that the performance is superior in the gradient-based setting.

4.5 INFERENCE TIME COMPARISONS BETWEEN DIFFERENT VIDEO CAPTIONING MODELS

This section details the inference time when applying the MAMS methodology compared to the original model. Table 5 shows three metrics. Inference Time per Video, the first metric, measures the time taken from the code's execution to generate a sentence for each video. In the MAMS-applied sections of the table, two values are presented. The first value represents the time taken when the small generation module is in use, and the second value indicates the time when the system selects the large generation module. The second metric, Video Count, denotes the number of test videos in each dataset. In the sections where MAMS is applied, the first value accounts for the number of videos selected by the small generation module, and the second value represents those chosen by the large generation module. Total Inference Time, the last metric, measures the entire time needed to process the test dataset, from executing the code to generating sentences for all test videos. After

Model	Dataset	Inference Time per Video	Video Counts	Total Inference Time
SwinBERT	MSVD	20.2 (-)	670	3020.23 (-)
SwinBERT + MAMS	MSVD	14.1 (-30.2%) / 23.5 (+16.3%)	332/338	**2398.22 (-20.6%)**
SwinBERT	MSRVTT	20.2 (-)	2990	467.56 (-)
SwinBERT + MAMS	MSRVTT	14.1 (-30.2%) / 23.5 (+16.3%)	564 / 2426	**420.34 (-10.1%)**

Table 5: Results of the inference time for MAMS framework on SwinBERT, with percentage changes in parenthesis.

	MSVD				MSRVTT			
, Models	B4	M	R	C	B4	M	R	C
SwinBERT	58.2	41.3	77.5	120.6	41.9	**29.9**	62.1	53.8
SwinBERT + MAMS	62.8	**41.9**	79	**127.3**	**43.8**	29.5	**62.9**	**55.2**
SwinBERT + adaptive MAMS	**63.2**	41.6	**79**	125.8	43.3	29.7	62.8	54.9
mPLUG-2	75.0	48.4	85.3	165.8	57.9	**34.9**	70.1	80.3
mPLUG-2 + adaptive MAMS	**80.5**	**48.7**	**87.9**	**176.1**	**60.0**	34.7	**71.2**	**80.9**

Table 6: Impact of adaptive module selection on SwinBERT & mPLUG-2

applying MAMS, we note that the second value of Inference Time per Video exceeds the time taken by the original model. This increase results from the additional computations needed during the module and token select phases, even when using modules of the same size as the original model. However, after applying MAMS, the first Inference Time per Video value significantly undercuts the original model's time. This is because, although the video undergoes the module token select operation, choosing the smaller generation model results in less computation during the sentence generation phase than the original model. Consequently, a review of the Total Inference Time shows that integrating MAMS into the model yields a time-saving benefit.

4.6 A VARIATION OF MODULE SELECTOR TRAINING

This section describes a modified learning process achieved through the module selector. A single video is trained on only one module in the MAMS approach. In our current experimental data, the number of videos selecting the small generation module is well balanced with those selecting the large one. However, if there is a tendency for the dataset to be biased towards one side, the affected module may need to be trained more effectively. We introduce a revised training process to address the issue of unbalanced learning while still allowing each data to select its module during inference. In (4), the possible combinations for $(\lambda_{\text{large}}, \lambda_{\text{small}})$ are $(1, 0)$ and $(0, 1)$. However, in adaptive module selection, the combinations $(\lambda_{\text{large}}, \lambda_{\text{small}})$ can be $(1, 0)$, $(1, 1)$, or $(0, 1)$, with $(1, 1)$ indicating the training of both modules simultaneously. We have employed the Gumbel-Softmax operation twice to implement this modified module selection algorithm. A detailed introduction to this algorithm, along with the pseudocode, can be found in Appendix A.4. As observed in Table 6, there is no significant difference between MAMS and the modified MAMS when applied to SwinBERT. Furthermore, by applying it to the state-of-the-art model mPLUG-2, we have achieved the latest benchmark performance metrics.

5 CONCLUSION

We pointed out the limitations in existing approaches of extracting a fixed number of frames in different videos: information loss and excessive information similarity. We propose a new model-agnostic framework using module selection to overcome these challenges. To further improve the proposed solution, we introduce the diversity promoting loss. Our numerical experiments with different datasets show that the proposed MAMS framework significantly improves existing prominent video captioning models. Our future work includes to improve the module selector training scheme in Section 3.3.2, as its variation in Section 4.6 showed potential improvements.

REFERENCES

Nayyer Aafaq, Naveed Akhtar, Wei Liu, Syed Zulqarnain Gilani, and Ajmal Mian. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12487–12496, 2019.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6836–6846, 2021.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, volume 2, pp. 4, 2021.

Qianwen Cao, Heyan Huang, Minpeng Liao, and Xianling Mao. Ada-swinbert: Adaptive token selection for efficient video captioning with online self-distillation. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 7–12. IEEE, 2023.

David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pp. 190–200, 2011.

Shaoxiang Chen, Ting Yao, and Yu-Gang Jiang. Deep learning for video captioning: A review. In *IJCAI*, volume 1, pp. 2, 2019.

Sihan Chen, Handong Li, Qunbo Wang, Zijia Zhao, Mingzhen Sun, Xinxin Zhu, and Jing Liu. Vast: A vision-audio-subtitle-text omni-modality foundation model and dataset. *arXiv preprint arXiv:2305.18500*, 2023.

Tianlong Chen, Zhenyu Zhang, Yu Cheng, Ahmed Awadallah, and Zhangyang Wang. The principle of diversity: Training stronger vision transformers calls for reducing all levels of redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12020–12030, 2022.

Chengpeng Dai, Fuhai Chen, Xiaoshuai Sun, Rongrong Ji, Qixiang Ye, and Yongjian Wu. Global2local: A joint-hierarchical attention for video captioning. *arXiv preprint arXiv:2203.06663*, 2022.

Pradipto Das, Chenliang Xu, Richard F Doell, and Jason J Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2634–2641, 2013.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arxiv 2020. *arXiv preprint arXiv:2010.11929*, 2010.

Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021.

Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. An empirical study of end-to-end video-language transformers with masked visual modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22898–22909, 2023.

Chengyue Gong, Dilin Wang, Meng Li, Vikas Chandra, and Qiang Liu. Vision transformers with patch diversification. *arXiv preprint arXiv:2104.12753*, 2021.

Xin Gu, Guang Chen, Yufei Wang, Libo Zhang, Tiejian Luo, and Longyin Wen. Text with knowledge graph augmented transformer for video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18941–18951, 2023.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Peng Jin, Jinfa Huang, Fenglin Liu, Xian Wu, Shen Ge, Guoli Song, David Clifton, and Jie Chen. Expectation-maximization contrastive learning for compact video-and-language representations. *Advances in Neural Information Processing Systems*, 35:30291–30306, 2022.

Atsuhiro Kojima, Takeshi Tamura, and Kunio Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50:171–184, 2002.

Weicheng Kuo, AJ Piergiovanni, Dahun Kim, Xiyang Luo, Ben Caine, Wei Li, Abhijit Ogale, Luowei Zhou, Andrew Dai, Zhifeng Chen, et al. Mammut: A simple architecture for joint learning for multimodal tasks. *arXiv preprint arXiv:2303.16839*, 2023.

Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7331–7341, 2021.

Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, Hehong Chen, Guohai Xu, Zheng Cao, et al. mplug: Effective and efficient vision-language learning by cross-modal skip-connections. *arXiv preprint arXiv:2205.12005*, 2022a.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021a.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022b.

Linjie Li, Jie Lei, Zhe Gan, Licheng Yu, Yen-Chun Chen, Rohit Pillai, Yu Cheng, Luowei Zhou, Xin Eric Wang, William Yang Wang, et al. Value: A multi-task benchmark for video-and-language understanding evaluation. *arXiv preprint arXiv:2106.04632*, 2021b.

Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pp. 605–612, 2004.

Kevin Lin, Linjie Li, Chung-Ching Lin, Faisal Ahmed, Zhe Gan, Zicheng Liu, Yumao Lu, and Lijuan Wang. Swinbert: End-to-end transformers with sparse attention for video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17949–17958, 2022.

Sheng Liu, Zhou Ren, and Junsong Yuan. Sibnet: Sibling convolutional encoder for video captioning. In *Proceedings of the 26th ACM international conference on Multimedia*, pp. 1425–1434, 2018.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3202–3211, 2022.

Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020.

Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10870–10879, 2020.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Wenjie Pei, Jiyuan Zhang, Xiangrong Wang, Lei Ke, Xiaoyong Shen, and Yu-Wing Tai. Memory-attended recurrent network for video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8347–8356, 2019.

Paul Hongsuck Seo, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid. End-to-end generative pretraining for multimodal video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17959–17968, 2022.

Yaojie Shen, Xin Gu, Kai Xu, Heng Fan, Longyin Wen, and Libo Zhang. Accurate and fast compressed video captioning. *arXiv preprint arXiv:2309.12867*, 2023.

Botian Shi, Lei Ji, Zhendong Niu, Nan Duan, Ming Zhou, and Xilin Chen. Learning semantic concepts and temporal alignment for narrated video procedural captioning. In *Proceedings of the 28th ACM international conference on multimedia*, pp. 4355–4363, 2020.

Xiangxi Shi, Jianfei Cai, Shafiq Joty, and Jiuxiang Gu. Watch it twice: Video captioning with a refocused video encoder. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 818–826, 2019.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7464–7473, 2019.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Ganchao Tan, Daqing Liu, Meng Wang, and Zheng-Jun Zha. Learning to discretely compose reasoning module networks for video captioning. *arXiv preprint arXiv:2007.09049*, 2020.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. Controllable video captioning with pos sequence guidance based on gated fusion network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2641–2650, 2019.

Zhenhailong Wang, Manling Li, Ruochen Xu, Luowei Zhou, Jie Lei, Xudong Lin, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Derek Hoiem, et al. Language models with image descriptors are strong few-shot video-language learners. *Advances in Neural Information Processing Systems*, 35:8483–8497, 2022.

Haiyang Xu, Qinghao Ye, Ming Yan, Yaya Shi, Jiabo Ye, Yuanhong Xu, Chenliang Li, Bin Bi, Qi Qian, Wei Wang, et al. mplug-2: A modularized multi-modal foundation model across text, image and video. *arXiv preprint arXiv:2302.00402*, 2023.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5288–5296, 2016.

Kashu Yamazaki, Khoa Vo, Quang Sang Truong, Bhiksha Raj, and Ngan Le. Vltint: visual-linguistic transformer-in-transformer for coherent video paragraph captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 3081–3090, 2023.

Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10714–10726, 2023.

Bang Yang, Tong Zhang, and Yuexian Zou. Clip meets video captioning: Concept-aware representation learning does matter. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pp. 368–381. Springer, 2022.

Ziqi Zhang, Zhongang Qi, Chunfeng Yuan, Ying Shan, Bing Li, Ying Deng, and Weiming Hu. Open-book video captioning with retrieve-copy-generate network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9837–9846, 2021.

Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

(T=16) Generated captions : a panda bear is climbing on a rock
(T=32) Generated captions : panda is playing
GT 1 : a baby panda relaxes on some logs
GT 2 : a panda bear is laying down
BLEU score (16/32) : 102.4 / 70.4

(T=16) Generated captions : a man is jumping in a pool
(T=32) Generated captions : a woman is jumping in a pool
GT 1 : a man dives into a swimming pool
GT 2 : a man dives into a pool
BLEU score (16/32) : 89.0 / 17.8

Figure 4: Result of Motivation experiments in SwinBERT

A Appendix

A.1 Motivation experiments

T'	MSVD	MSRVTT
4	50.2	31.7
8	57.4	38.4
16	59.2	42.0
32	61.5	42.5
64	60.5	43.0
128	59.8	41.6

Table 7: The results presented captioning performance with varied frame counts in SwinBERT.

The table 7 shows the results of experiments conducted with the SwinBERT model, excluding using the learnable sparse mask. We ensured a fair comparison by maintaining consistent batch sizes and providing sufficient epochs in a uniform experimental environment. From the table, we can derive two key insights. First, the captioning performance typically improves with the increased number of extracted frames. When the model pulls substantial information from the video, it becomes more adept at generating accurate captions. Second, a noticeable performance decline occurs when an excessive number of frames are extracted. The optimal performance for MSVD is observed with 32 frames, while MSRVTT peaks at 64 frames. These observations lead us to conclude that over-extraction of information can obstruct both the learning and inference processes, ultimately diminishing the model's overall performance.

The image displays the results of experiments conducted on the MSVD dataset using SwinBERT models with 32-frame and 16-frame sizes, respectively. For the MSVD dataset, the 32-frame model outperforms the 16-frame counterpart. However, as observed in the image, there are instances where the 16-frame model yields better results than the 32-frame model We drew the following conclusions considering both the experimental outcomes and these specific instances. Extracting a substantial number of frames aids in conveying ample video information to the model, enabling the generation of more accurate captions. However, extracting excessive frames proves ineffective during the

model's learning and inference stages. Based on these findings, we identified the limitations and potential for improvement in traditional video captioning models that extract a fixed number of frames from every dataset. This discovery prompted the research presented in this paper.

A.2 IMPLEMENTATION DETAILS

A.2.1 UNIVL

Unlike the other two baseline models, the UniVL model trains on the features of given frames, not by extracting frames from raw video. This means we already have extracted frames for each video data at the outset. In the video encoder, we extract tokens, and during this process, we use masking to adjust the length of the input or trim it if it overflows. To apply our model to UniVL, we design the framework considering the masking tokens as one of the tokens. However, we should reasonably view that the masking tokens do not impact sentence generation. We calculate the token significance score with the tokens, excluding the masking ones. When conducting the operation, the number of selected tokens will also vary for each video. We apply the UniVL technique directly, using masking for the spaces that lack. However, since this is an input going into a small generation module, we assume a maximum length of 16, unlike the large generation module that has set the maximum input size to 32.

A.2.2 MPLUG-2

A unique aspect of the mPLUG-2 model is that it does not involve attention between the CLS and video tokens during training. With this in mind, we redefine the formula in the method section to represent the cumulative impact each video token exerts on each text token, not the CLS token. The procedures proceed as described initially in the method section

A.2.3 SWINBERT

SwinBERT distinguishes itself from the UniVL and mPLUG-2 settings by extracting multiple tokens from two frames. In applying MAMS to SwinBERT, we consider two frames to be one and execute the operation. We extract tokens that correspond to these two frames during the token selection process. This characteristic mandates that the variable T' in MAMS be an even number exclusively for SwinBERT. Another noteworthy feature involves applying the learnable attention mask even to the first layer. For the implementation, we compute the formula using the original attention value before integrating it with the learnable attention mask.

A.3 ALGORITHM OF THE GUMBEL SOFTMAX TRICK IN MODULE & TOKEN SELECTOR

As outlined in Algorithm 2, we introduced a tok-T' selection mechanism that employs the Gumbel-softmax operation T' times. Several models in contemporary literature utilize a Gumbel softmax-based trick for top-k selection. The majority of these implementations involve a sampling with a value replacement strategy. While extracting the top-k indices, such a strategy applies the gumbel-softmax operation k times. Due to the probabilistic nature of the gumbel-softmax computation, extracting an exact set of top k indices is not guaranteed. Therefore, diverging from conventional methods of restoring the original values, our implementation, as shown in Line 9 of the algorithm, we replaces the values corresponding to the extracted indices with a small value0. Subsequently, proceed with the Gumbel softmax operation. Nonetheless, we incorporated a while loop to account for instances where precisely T' indices are not extracted, as shown in Lines 11-20 of the algorithm. This ensures the extraction of an exact T' indices.

A.4 ALGORITHM OF MODIFY MODULE SELECTION

Looking at Equation 4, the candidates for (α, β) become $(1, 0)$ and $(0, 1)$ through the operation involving $\tilde{f}$ and λ. ($\tilde{f}$ is the overall significance score of the selected frame, as defined in Section 3.3.1) The modified module learning approach proceeds through an Adaptive Module Selection function utilizing the Gumbel-Softmax function. The principle of the Adaptive Module Selection function is as follows: If the value of $\tilde{f}$ is large, it indicates that the tokens extracted from the selected frames are heavily influenced by the caption, meaning that the data is more suitable for the

Algorithm 1 Get the selected frame index using Gumbel-softmax trick

INPUT: f^s : frame_score
OUTPUT: f^s_{idx} : selected_frame_index
$f^s_{idx} \leftarrow$ all-zeros tensor of same shape as f^s
for _ $\in$ range$(1, T')$ **do**
    one_hot $\leftarrow$ Gumbel-Softmax(f^s)
    mask $\leftarrow (f^s_{idx} + \text{one\_hot}) \leq 1$
    $f^s_{idx} \leftarrow f^s_{idx} + \text{one\_hot} \times \text{mask}$
    indices $\leftarrow$ max index of one_hot
    $f^s[\text{indices}] \leftarrow 0$
end for
while true **do**
    one_hot $\leftarrow$ Gumbel-Softmax(f^s)
    mask $\leftarrow (f^s_{idx} + \text{one\_hot}) \leq 1$
    $f^s_{idx} \leftarrow f^s_{idx} + \text{one\_hot} \times \text{mask}$
    indices $\leftarrow$ max index of one_hot
    $f^s[\text{indices}] \leftarrow 0$
    if sum of $f^s_{idx} \geq T'$ **then**
        break
    end if
end while

Algorithm 2 Get the selected frame index using Gumbel-softmax trick

INPUT: $\tilde{f}$
OUTPUT: $Y : (1,0) or (1,1) or (0,1)$
$X \leftarrow (\tilde{f}, \lambda)$
$Y \leftarrow all\_zeros\_tensor\_of\_same\_shape\_as X$
for _ $\in$ range$(1, 2)$ **do**
    one_hot $\leftarrow$ Gumbel-Softmax(X)
    mask $\leftarrow (Y + \text{one\_hot}) \leq 1$
    $Y \leftarrow Y + \text{one\_hot} \times \text{mask}$
    indices $\leftarrow$ max index of one_hot
end for

small generation module. Taking this into account, we extract the outputs of (α, β) such as $(1, 0)$, $(1, 1)$, and $(0, 1)$ by performing the Gumbel-Softmax operation twice on the $(\tilde{f}, \lambda)$ values. Since the Gumbel-Softmax operation is probabilistic, the probability of obtaining $(\alpha, \beta) = (1, 0)$ will be higher if the S value is large. If the $\tilde{f}$ and λ values are similar, $(1, 1)$ is more likely, and if the λ value is large, $(0, 1)$ is more likely. This operation implies that if S is large, training occurs in the small generation module; if $\tilde{f}$ is small, training is directed toward the large generation module, and training occurs in both modules if the value is ambiguous. The corresponding equation is as follows: $L_{\text{final}} = L_{\text{large}} \cdot \alpha + L_{\text{small}} \cdot \beta,$ where$(\alpha, \beta) = modified_module_selector(\tilde{f})$