A. Datasheet

These questions were copied from “Datasheets for Datasets” [21].

A.1. Motivation

The questions in this section are primarily intended to encourage dataset creators to clearly articulate their reasons for creating the dataset and to promote transparency about funding interests. The latter may be particularly relevant for datasets created for research purposes.

- **For what purpose was the dataset created?** Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

  Multimodal summarization with multimodal output (MSMO) has emerged as a promising research direction. Nonetheless, numerous limitations exist within existing public MSMO datasets, including insufficient upkeep, data inaccessibility, limited size, and the absence of proper categorization, which pose significant challenges to effective research. To address these challenges and provide a comprehensive dataset for this new direction, we have meticulously curated the **MultiSum** dataset.

- **Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?**

  Our institution (will release the identity later).

- **Who funded the creation of the dataset?** If there is an associated grant, please provide the name of the grantor and the grant name and number.

  Our institution (will release the identity later).

- **Any other comments?**

  No.

A.2. Composition

Dataset creators should read through these questions prior to any data collection and then provide answers once data collection is complete. Most of the questions in this section are intended to provide dataset consumers with the information they need to make informed decisions about using the dataset for their chosen tasks. Some of the questions are designed to elicit information about compliance with the EU’s General Data Protection Regulation (GDPR) or comparable regulations in other jurisdictions.

Questions that apply only to datasets that relate to people are grouped together at the end of the section. We recommend taking a broad interpretation of whether a dataset relates to people. For example, any dataset containing text that was written by people relates to people.

- **What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)?** Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

  Videos, transcripts, keyframes, textual summaries, segmentation boundaries, titles, authors, and thumbnails.

- **How many instances are there in total (of each type, if appropriate)?**

  17 main categories and 170 subcategories, 5,100 videos, in a total of 1229.9 hours.

- **Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?** If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

  It contains all possible instances.

- **What data does each instance consist of?** "Raw" data (e.g., unprocessed text or images) or features? In either case, please provide a description.

  Video, transcripts, keyframes, textual summaries, segmentation boundaries, titles, authors, and thumbnails.

- **Is there a label or target associated with each instance?** If so, please provide a description.

  Segmentation boundaries are labels for video temporal segmentation. Keyframes and textual summaries are labels for MSMO. Thumbnails are the ground-truth for thumbnail generation.

- **Is any information missing from individual instances?** If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

  No.

- **Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)?** If so, please describe how these relationships are made explicit.

  N/A.

- **Are there recommended data splits (e.g., training, development/validation, testing)?** If so, please provide a description of these splits, explaining the rationale behind them.

  Yes, for the train/val/test split, since our dataset is already randomly collected from YouTube, we designate the last 30% of videos within each subcategory (indexed 21-29) as the testing set. The remaining videos are then assigned to the training set (indexed 00-20) in each subcategory.

- **Are there any errors, sources of noise, or redundancies in the dataset?** If so, please provide a description.

  No, but we would expect subjectivity exists in the annotations.

- **Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?** If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a dataset consumer? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

  The Youtube links to the videos are provided in the public version. If some of the data become unavailable, we can provide the features for those videos.

- **Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor–patient confidentiality, data that includes the content of individuals' non-public communications)?** If so, please provide a description.

  No.

- **Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?** If so, please describe why.

  No.

If the dataset does not relate to people, you may skip the remaining questions in this section.

- **Does the dataset identify any subpopulations (e.g., by age, gender)?** If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

  No.

- **Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset?** If so, please describe how.

  No, too many people are involved in the videos.

- **Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals race or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)?** If so, please provide a description.

  No.

A.3. Collection Process

As with the questions in the previous section, dataset creators should read through these questions prior to any data collection to flag potential issues and then provide answers once collection is complete. In addition to the goals outlined in the previous section, the questions in this section are designed to elicit information that may help researchers and practitioners to create alternative datasets with similar characteristics. Again, questions that apply only to datasets that relate to people are grouped together at the end of the section.

- **How was the data associated with each instance acquired?** Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

  Data resources are publicly available online.

- **What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)?** How were these mechanisms or procedures validated?

  We open-sourced our data collection tool.

- **If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?**

  N/A.

- **Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?**

  Students and crowdworkers, and were rewarded with virtual currency.

- **Over what timeframe was the data collected?** Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.

  No specific timeframe is set.

- **Were any ethical review processes conducted (e.g., by an institutional review board)?** If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation.

  Yes. The legal counsel reviewed it.

If the dataset does not relate to people, you may skip the remaining questions in this section.

- **Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?**

  From YouTube.

- **Were the individuals in question notified about the data collection?** If so, please describe (or show with screenshots or other information) how notice was provided, and provide a link or other access point to, or otherwise reproduce, the exact language of the notification itself.

  N/A.

- **Did the individuals in question consent to the collection and use of their data?** If so, please describe (or show with screenshots or other information) how consent was requested and provided, and provide a link or other access point to, or otherwise reproduce, the exact language to which the individuals consented.

  We follow the license from YouTube.

- **If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses?** If so, please provide a description, as well as a link or other access point to the mechanism (if appropriate).

  We follow the license from YouTube.

- **Has an analysis of the potential impact of the dataset and its use on data subjects (e.g., a data protection impact analysis) been conducted?** If so, please provide a description of this analysis, including the outcomes, as well as a link or other access point to any supporting documentation.

  No.

- **Any other comments?**

  No.

A.4. Preprocessing/cleaning/labeling

Dataset creators should read through these questions prior to any preprocessing, cleaning, or labeling and then provide answers once these tasks are complete. The questions in this section are intended to provide dataset consumers with the information they need to determine whether the "raw" data has been processed in ways that are compatible with their chosen tasks. For example, text that has been converted into a "bag-of-words" is not suitable for tasks involving word order.

- **Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)?** If so, please provide a description. If not, you may skip the remaining questions in this section.

  No.

- **Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)?** If so, please provide a link or other access point to the "raw" data.

  The dataset will be available on our website.

- **Is the software that was used to preprocess/clean/label the data available?** If so, please provide a link or other access point.

  Yes, the data collection tool will be available on our website.

- **Any other comments?**

  No.

A.5. Uses

The questions in this section are intended to encourage dataset creators to reflect on the tasks for which the dataset should and should not be used. By explicitly highlighting these tasks, dataset creators can help dataset consumers to make informed decisions, thereby avoiding potential risks or harms.

- **Has the dataset been used for any tasks already?** If so, please provide a description.

  Yes, the dataset has been used for a series of tasks, including video temporal segmentation, video summarization, text summarization, multimodal summarization (MSMO), and thumbnail generation.

- **Is there a repository that links to any or all papers or systems that use the dataset?** If so, please provide a link or other access point.

  Yes, it will be released on our website.

- **What (other) tasks could the dataset be used for?**

  Video related, multimodal related.

- **Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?** For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other risks or harms (e.g., legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?

  No.

- **Are there tasks for which the dataset should not be used?** If so, please provide a description.

  No.

- **Any other comments?**

  No.

A.6. Distribution

Dataset creators should provide answers to these questions prior to distributing the dataset either internally within the entity on behalf of which the dataset was created or externally to third parties.

- **Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?** If so, please provide a description.

  No.

- **How will the dataset will be distributed (e.g., tarball on website, API, GitHub)?** Does the dataset have a digital object identifier (DOI)?

  Self-contained website.

- **When will the dataset be distributed?**

  Will be released soon.

- **Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?** If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

  CC BY-NC-SA License.

- **Have any third parties imposed IP-based or other restrictions on the data associated with the instances?** If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.

  No.

- **Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?** If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.

  No.

- **Any other comments?**

  No.

A.7. Maintenance

As with the questions in the previous section, dataset creators should provide answers to these questions prior to distributing the dataset. The questions in this section are intended to encourage dataset creators to plan for dataset maintenance and communicate this plan to dataset consumers.

- **Who will be supporting/hosting/maintaining the dataset?**

  Our institute.

- **How can the owner/curator/manager of the dataset be contacted (e.g., email address)?**

  We will provide the contact information on the webpage.

- **Is there an erratum?** If so, please provide a link or other access point.

  We will provide the contact information on the webpage.

- **Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?** If so, please describe how often, by whom, and how updates will be communicated to dataset consumers (e.g., mailing list, GitHub)?

  Currently no, but we will post an announcement on the website once there is a new version available.

- **If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were the individuals in question told that their data would be retained for a fixed period of time and then deleted)?** If so, please describe these limits and explain how they will be enforced.

  No.

- **Will older versions of the dataset continue to be supported/hosted/maintained?** If so, please describe how. If not, please describe how its obsolescence will be communicated to dataset consumers.

  Yes. All the released versions will be hosted on the same website.

- **If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?** If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to dataset consumers? If so, please provide a description.

  Under the same license and contact our team.

- **Any other comments?**

  No.

A.8. URL to Website

Will be available soon.

A.9. Responsibility Statement

The authors declare that they bear all responsibility for violations of rights related to this dataset.

A.10. Hosting, licensing, and maintenance plan

The dataset will be properly hosted and maintained through our website. The dataset is under CC BY-NC-SA License.

A.11. Data Format

The released dataset includes (1) the annotation files in the $.json$ format, including video transcripts, video segmentation boundaries, textual summaries for each segment, titles, authors, and timestamps. (2) the keyframe files in the $.png$ format. (3) The thumbnail files in the $.png$ format.

A.12. Long-term preservation

The authors guarantee that the dataset will be properly hosted and maintained through our website for a long time.

B. Categories of MMSum Dataset

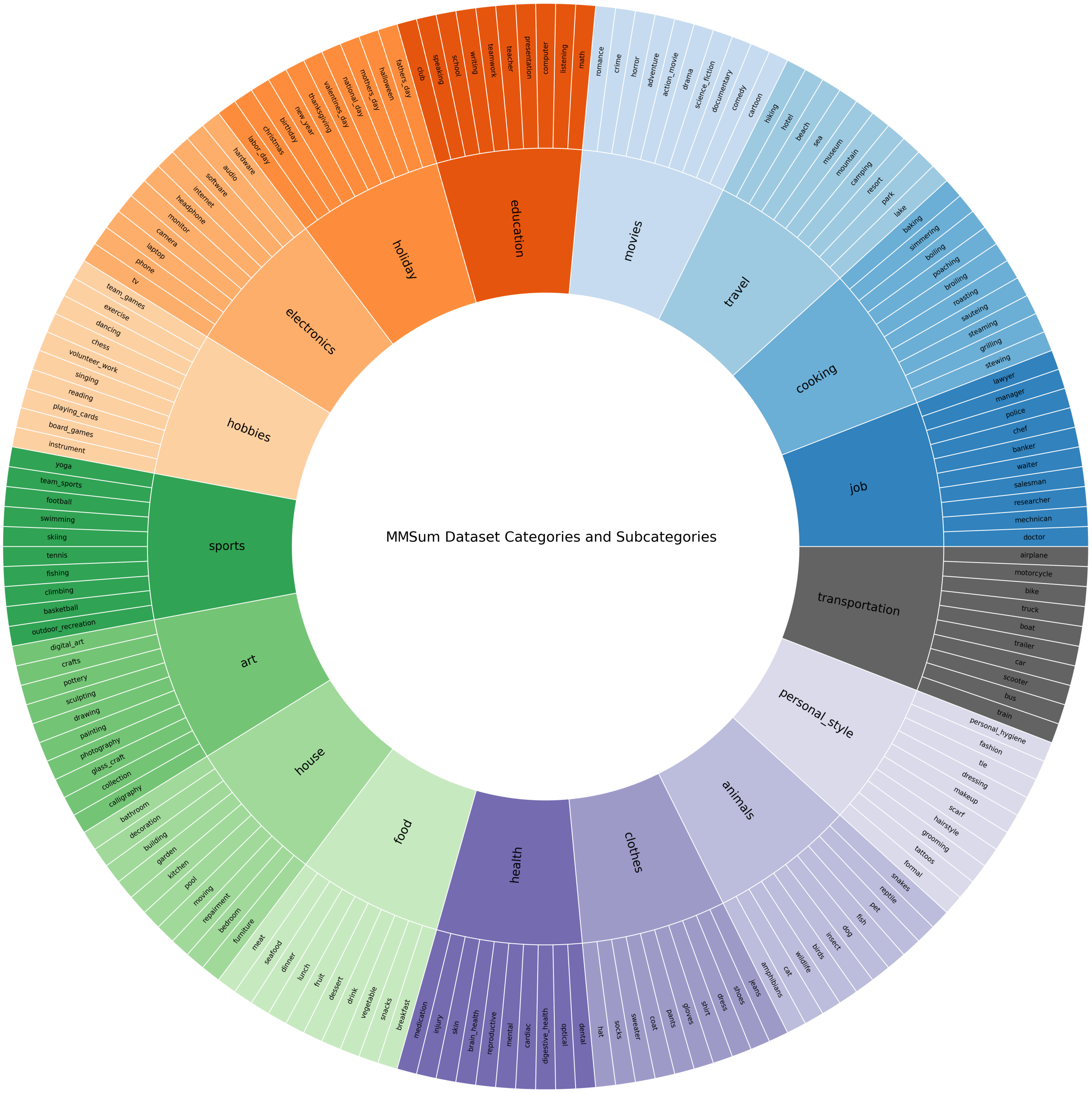
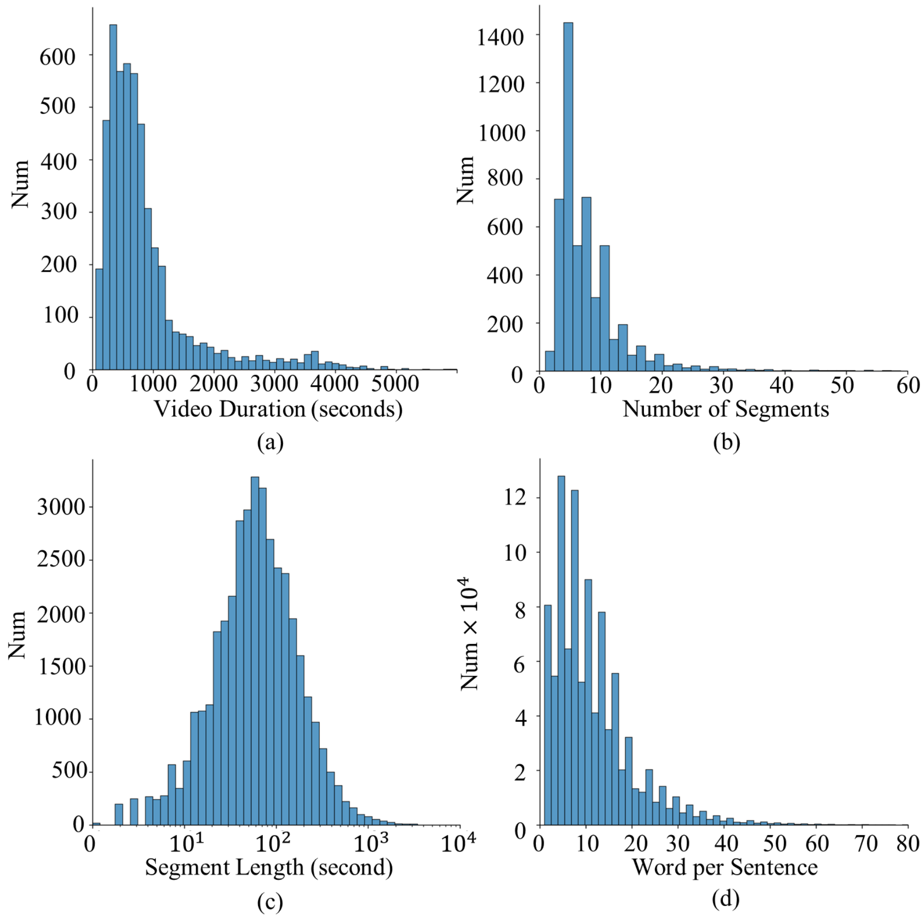


Figure 7. The 17 categories (with 10 subcategories each) of the MMSum dataset, resulting in 170 subcategories in total.

C. Tasks

Our dataset contains sufficient information, making it possible to conduct many downstream tasks, such as video temporal segmentation (VTS), video summarization (VS), text summarization (TS), and multimodal video summarization with multimodal output (MSMO). To make it more clear, we highlight the description of each task and the differences between them.



(a)

(b)

(c)

(d)

Figure 8. The statistics of the MMSum dataset, which show the distribution of (a) video duration; (b) number of segments per video; (c) segment duration; (d) number of words per sentence

Video Temporal Segmentation (VTS) Video temporal segmentation (VTS) is the process of partitioning a video sequence into disjoint sets of consecutive frames that are homogeneous according to some defined criteria. Normally, VTS aims at splitting the whole video into several small segments based on video scene change, which is also related to video shot detection and video transition detection. Multimodal Video Temporal Segmentation (M-VTS) differs from VTS, where textual data (video transcript) is also used as inputs for splitting the input video into small video segments.

Video Summarization (VS) Video Summarization aims at generating a short synopsis that summarizes the video content by selecting the most informative and vital parts. The input only contains visual information and uses computer vision mechanisms to generate summaries.

Text Summarization (TS) Textual summarization takes textual metadata, i.e., documents, articles, tweets, etc, as input, and generates textual summaries, in two directions: abstractive summarization and extractive summarization. Abstractive methods select words based on semantic understanding, and even the words may not appear in the source [89, 97]. Extractive methods attempt to summarize language by selecting a subset of words that retain the most critical points, which weights the essential part of sentences to form the summary [65, 106].

Multimodal Summarization with Multimodal Output (MSMO) MSMO aims to produce both visual and textual summaries for a given video. Different from pure video summarization, MSMO takes both visual and textual information as inputs and outputs both visual and textual summaries.

Table 6. Categories and sub-categories for the MMSum dataset.

Category	Sub-categories	Number
Animals	Dog, Wildlife, Cat, Fish, Birds, Insect, Snakes, Pet, Amphibians, Reptile	30×10 =300
Education	School, Club, Teacher, Speaking, Listening, Writing, Presentation, Math, Computer, Teamwork	30×10 =300
Health	Mental, Injury, Medication, Digestive health, Dental, Optical, Reproductive, Skin, Brain health, Cardiac	30×10 =300
Travel	Museum, Park, Sea, Beach, Mountain, Lake, Hotel, Resort, Camping, Hiking	30×10 =300
Movies	Action movie, Comedy, Romance, Science fiction, Horror, Drama, Cartoon, Documentary, Adventure, Crime	30×10 =300
Cooking	Broiling, Grilling, Roasting, Baking, Sauteing, Boiling, Steaming, Poaching, Simmering, Stewing	30×10 =300
Job	Manager, Researcher, Chef, Police, Lawyer, Salesman, Mechnican, Banker, Doctor, Waiter	30×10 =300
Electronics	laptop, TV, Phone, Software, Internet, Camera, Audio, Headphone, Hardware, Monitor	30×10 =300
Art	Crafts, Photography, Painting, Collection, Drawing, Digital art, sculpting, pottery, glass craft, calligraphy	30×10 =300
Personal Style	Grooming, Fashion, Personal Hygiene, Tattoos, Scarf, Hair Style, Makeup, Dressing, Tie, Formal	30×10 =300
Clothes	Sweater, Jeans, Shirt, Socks, Coat, Pants, Hat, Gloves, Dress, Shoes	30×10 =300
Sports	Outdoor recreation, Team sports, Tennis, Football, Basketball, Climbing, Skiing, Swimming, Fishing, Yoga	30×10 =300
House	Building, Garden, Pool, Bathroom, Bedroom, Kitchen, Repairment, Moving, Decoration, Furniture	30×10 =300
Food	Fruit, Vegetable, Drink, Meat, Seafood, Snacks, Dessert, Breakfast, Lunch, Dinner	30×10 =300
Holiday	Halloween, Christmas, Labor day, Thanksgiving, Valentines day, Mother's day, Birthday, National day, New year, Father's day	30×10 =300
Transportation	Car, Train, Bus, Boat, Bike, Airplane, Motorcycle, Truck, Trailer, Scooter	30×10 =300
Hobbies	Dancing, Singing, Playing cards, Reading, Chess, Board games, Team games, Volunteer work, Instrument, Exercise	30×10 =300
Total	——————————————————————————————-	$17 \times 30 \times 10$=5,100

D. More Details about Our Model

Text Encoder The Transformer encoder [101] is employed to convert the text into a sequence of token embeddings. Inspired by [37, 110], we initialize the encoder's weights using the pre-trained mT5 model [108]. To investigate the impact of task-specific pre-training, we fine-tune mT5 on the text-to-text summarization task, where $X_{enc} = \mathrm{TextEncoder}(X)$.

Video Encoder To capture short-term temporal dependencies, we utilize 3D convolutional networks as in [37]. We partition the video into non-overlapping frame sequences and employ a 3D CNN network for feature extraction. Specifically, we utilize two different feature extractors. Firstly, we utilize the $\mathrm{R}(2+1)\mathrm{D}$ model trained by [22] for video action recognition on weakly-supervised social-media videos. Secondly, we utilize the visual component of the S3D Text-Video model trained in a self-supervised manner by [52] on the HowTo100M dataset [54]. To incorporate long-term temporal dependencies, we process the sequence of video features using a Transformer encoder. This enables us to effectively capture and model the relationships between video frames over an extended duration, where $V_{\text{enc}} = \mathrm{3D-CNN}(V), V_{enc} = \mathrm{VideorEncoder}\left(V_{\text{enc}}\right)$.

Frame Encoder To facilitate the selection of a specific frame as a cover picture, we require frame-level representations [37]. In our experimental setup, we sample one frame per second from the video. For feature extraction, we employ two models: EfficientNet [98] and Vision Transformer (ViT) [18]. Both models were pre-trained on the ImageNet dataset [85] for image classification tasks. To provide contextual information, we process the sequence of frame features using a Transformer encoder, which captures the relationships and dependencies between the frame-level representations, enabling a more comprehensive understanding of the video content. Before applying the Transformer encoder, we ensure that both the video features and frame features have the same dimensions as the hidden states of the text encoder. In the case of a single model, the two sets of features are concatenated together before undergoing the projection step.

$$V_{\text{frame}} = \mathrm{CNN}(\mathrm{Sample}(V)), V_{\text{frame}} = \mathrm{FrameEncoder}\left(V_{\text{frame}}\right) \tag{1}$$

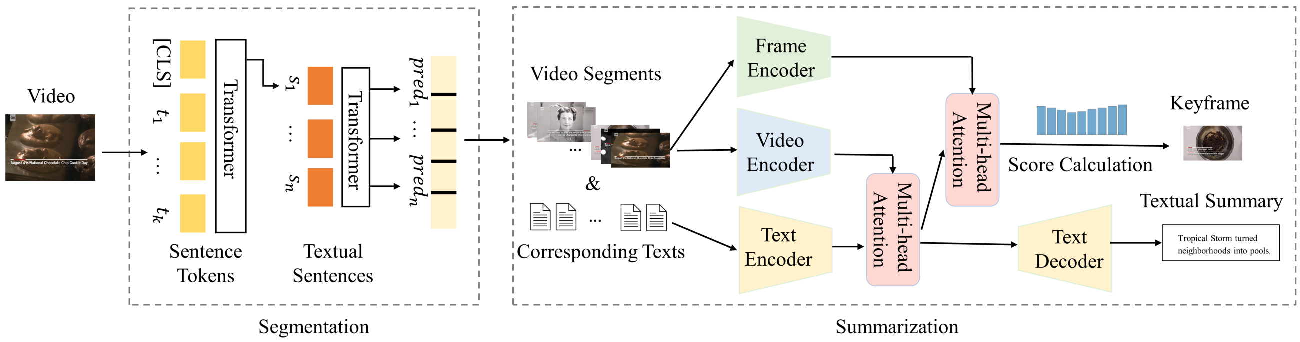


Figure 9. An overview of our model.

Multi-head Attention In line with the study conducted by [37, 110], which explored various methods of integrating visual information into pre-trained generative language models, we adopt the approach of multi-head attention-based fusion. This technique allows us to obtain a vision-guided text representation by incorporating visual information into the model. The fusion process takes place after the last encoder layer, ensuring that both textual and visual inputs are combined effectively to enhance the overall representation.

$$\begin{aligned} &Q = X_{enc}W_q, Q \in \mathbb{R}^{M \times d}, K = V_{enc}W_k, K \in \mathbb{R}^{N' \times d} \\ &V = V_{enc}W_v, V \in \mathbb{R}^{N' \times d}, \widetilde{X}_{enc} = \text{MHA}(Q, K, V), \widetilde{X}_{enc} \in \mathbb{R}^{M \times d} \end{aligned} \tag{2}$$

As recommended by [37, 46], we incorporate the use of the forget gate mechanism (FG) in our model. This mechanism enables the model to filter out low-level cross-modal adaptation information. By utilizing the forget gate, our model can selectively retain and focus on the most relevant and informative features, disregarding less important or noisy information during the cross-modal fusion process. This helps improve the overall performance and robustness of the model in handling multimodal data.

$$\widehat{X}_{\text{enc}} = \text{FG}\left(X_{\text{enc}}, \widetilde{X}_{\text{enc}}\right), \widehat{X}_{\text{enc}} \in \mathbb{R}^{M \times d} \tag{3}$$

To obtain the text+video guided frame representations, we employ the same multi-head attention mechanism. However, in this case, we substitute the input X_{enc} with V_{frame} and V_{enc} with $\widehat{X}enc$. By using the video frame features $Vframe$ and the transformed text representations $\widehat{X}enc$, we generate the guided frame representations $\widehat{V}frame$ through the multi-head attention process. This allows us to effectively incorporate both textual and visual information, guiding the frame-level representations based on the context provided by the text and video.

Text Decoder To generate the textual summary, we employ a standard Transformer decoder, initializing its weights with the mT5 checkpoint. The vision-guided text representation $\widehat{X}_{enc}$ serves as the input to the decoder. During training, we utilize the standard negative log-likelihood loss (NLLLoss) with respect to the target sequence Y. This loss function measures the dissimilarity between the predicted summary generated by the model and the ground truth summary, allowing the model to learn and improve its summary generation capabilities through backpropagation.

$$\widehat{Y} = \text{TransformerDecoder}\left(\widehat{X}_{enc}\right), \mathcal{L}_{\text{text}} = \text{NLLLoss}\left(\widehat{Y}, Y\right) \tag{4}$$

To obtain the labels C for the cover picture (cover frame) selection, we calculate the cosine similarity between the CNN features of the reference cover picture and the candidate frames. In most instances, the similarity values fall within the range of [0, 1], while the remaining negative values are mapped to 0. Previous studies such as [43] and [19] considered the frame with the maximum cosine similarity as the ground truth (denoted as $C_{\max}$), while considering the other frames as negative samples. However, upon analyzing the cosine similarity patterns, we observed that some videos exhibit multiple peaks or consecutive sequences of frames with very similar scores, capturing still scenes. We recognized that this could potentially harm the model's performance, as very similar frames might be labeled as both positive and negative examples. To address this issue, in addition to the binary labels $C_{\max}$, we introduce smooth labels denoted as C_{smooth}. These smooth labels assign to each frame its cosine similarity score with the reference cover picture. By incorporating the smooth labels, we aim to provide a more nuanced and continuous representation of the frame similarities, allowing the model to learn from a broader range of similarity scores during the training process.

In our approach, we utilize a projection matrix to map the text+video guided frame representations $\widehat{V}_{frame}$ to a single dimension. This dimension reduction step allows us to obtain a compact representation of the frame features. Subsequently, we train the model using the binary cross-entropy (CE) loss, where the target labels C can either be $C\max$ or C_{smooth}. To train the entire model in an end-to-end fashion, we minimize the sum of losses $\mathcal{L}$, which includes the negative log-likelihood loss for textual summary generation and the binary cross-entropy loss for cover picture selection. By jointly optimizing these losses, the model learns to generate accurate summaries and make effective cover picture selections based on the input text and video. Please note that $\mathcal{L}$ refers to the combined loss function that encompasses both the negative log-likelihood loss for summary generation and the binary cross-entropy loss for cover picture selection.

$$\widehat{C}=\widehat{V}_{\text{frame}}\, W_p, W_p \in \mathbb{R}^{d \times 1}, \mathcal{L}_{\text{image}}\ = \text{CE}(\widehat{C}, C), \mathcal{L}=\mathcal{L}_{\text{text}}\ +\mathcal{L}_{\text{image}} \tag{5}$$

E. Baseline Implementation Details

E.1. Video Temporal Segmentation

Video Temporal Segmentation Evaluation For VTS, we followed [82] and adopted four common metrics: (1) Average Precision (AP); (2) F1 score; (3) M_{iou}: a weighted sum of the intersection of the union of a detected scene boundary with respect to its distance to the closest ground-truth scene boundary; and (4) Recall@ks: recall at k seconds ($k = \{3, 5, 10\}$), the percentage of annotated scene boundaries which lies within k-second window of the predicted boundary.

The performance of video temporal segmentation has a great impact on the final performance, so in this section, we compare the performance of VTS with several baselines: Histogram Intersect [40], Moment Invariant [32], Twin Comparison [115], PySceneDetect [9], and LGSS [82].

Histogram Intersect We predict video boundaries at time t when the overlap of color histograms in consecutive frames of the video

$$\frac{\sum_b \min(H_{t,b}, H_{t-1,b})}{\sum H_{t-1}} \geq 0.5 \tag{6}$$

As in the original work [40], we weighted the H, S, and V channels of the base image $0.5, 0.3, 0.2$ when constructing the histogram.

Moment Invariant We predict video boundaries at time t when the distance between the Hu image moments of consecutive frames of the video $dist_{Hu}(I_t, I_{t-1}) \geq 0.3$.

Twin Comparison We define hyperparameters $T_s = 16, T_b = 3750$, such that the algorithm predicts the start of a segment at time t where the difference between consecutive frames $D_{t,t-1} > T_s$, and the end of a segment at t' when $D_{t,t'} > T_b$.

PySceneDetect We run the tool with hyperparameters $adaptive\_threshold = 64, min\_scene\_length = 5$.

LGSS We identify boundaries where the mean difference across channels H, S, and V between consecutive frames of the video $D_{HSV} \geq 20$ [82].

E.2. Video Summarization

For video summarization, we selected the following representative methods as our baselines: Uniform Sampling [33], K-means Clustering [26], Scale Invariant Feature Transform (SIFT) [58], VSUMM [16], and Keyframe Extraction [33].

Uniform Sampling We downsample the videos to 1 frame per second before taking 5 percent of the video frames, evenly spacing them throughout the video to have a uniform sample of key frames.

K-means Clustering We compute the video's histogram per frame and apply K-means to find relevant frames for the summarization process. To extract the required images, images were captured at 1 FPS using the cv2 library.

Scale Invariant Feature Transform (SIFT) We again downsample videos to 1 frame per second, then compute the Euclidean distance between the SIFT feature vectors of adjacent keyframes and select those with a difference greater than some threshold. For the segment-level summarization, we take the maximum, and for the whole-video summarization, we select keyframes whose differences from the previous keyframe are greater than the average.

VSUMM For VSUMM, we use a sampling rate the same as the fps of the video.

Keyframe Extraction Using the video downsampled to 1 fps, we sampled one out of every 3 frames. We then use the differences of adjacent frames (represented as a CNN feature vector) to define scenes in the video. We set the threshold for drawing scene boundaries to 0.65. Using K-means and Euclidean distance, we cluster the keyframes per scene and then remove redundant candidate keyframes from the same scene using a threshold of 0.8. [33].

E.3. Text Summarization

For textual summarization, we selected the following representative models as our baselines: BERT2BERT [100], BART [41] (BART-large-CNN and BART-large-XSUM), Distilbart [91], T5 [80], Pegasus [117], and Longformer Encoder-Decoder (LED) [6].

BERT2BERT Through an encoder-decoder architecture with the auto-regressive generation, we predict summaries from the extracted text at time t. Tokenized length T_m and summary length S_m are bounded as follows: $T_m \leq 512, 2 \leq S_m \leq 15$. Additional parameters include: *truncation* = True, *padding* = ”max-length”, *skip_special_tokens* = True. The pretrained model used can be found in the transformers library under BertTokenizerFast.

BART-large-CNN Using an encoder-encoder framework, the BART-large-CNN model first corrupts text with a noising function, then reconstructs this text with a CNN. Tokenized length T_m and summary length S_m are bounded as follows: $T_m \leq 512, 1 \leq S_m \leq 10$. Additional parameters include: *num_beams* = 2, *clean_up_tokenization_space* = True. The pretrained Facebook model used can be found in the transformers library under BartforConditionalGeneration.

BART-large-XSUM Similar to BART-large-CNN, BART-large-XSUM employs a transformer-based neural machine translation architecture, effective in text generation and comprehension. Tokenized length T_m and summary length S_m are bounded as follows: $T_m \leq 512, 1 \leq S_m \leq 10$. Additional parameters include: *num_beams* = 2, *skip_special_tokens* = True.

Distilbart We use distilbart-cnn-6-6, which copies alternating layers from the BART-large-CNN model and integrates MSE loss from the tinybert model. Tokenized length T_m and summary length S_m are bounded as follows: $T_m \leq 512, 4 \leq S_m \leq 15$. A pretrained model from the transformers library was implemented: ”ml6team/distilbart-tos-summarizer-tosdr”.

T5 T5 integrates supervised and unsupervised tasks in an encoder-decoder framework. We used the ”t5-small” model, having optimized runtime compared to other T5 models. Summary length S_m was bounded as follows: $2 \leq S_m \leq 15$. Additional parameters include: num_beams = 4, $no\_repeat\_ngram\_size$ = 2, $early\_stopping$ = True.

Pegasus Pegasus masks important sentences from the text, combining these into an output sequence to develop an informative summary; we used the “pegasus-xsum” model, being the most fine-tuned. Summary length S_m was bounded as follows: $2 \leq S_m \leq 15$. Additional parameters include: $padding$ = longest, $truncation$ = True.

Longformer Encoder-Decoder (LED) LED employs similar architectures to the BART model; however, it works better on longer input text (over 1024 tokens). We used the “led-large-16384” model; some parameters include: $repetition\_penalty$ = 3.5, $encoder\_no\_repeat\_ngram\_size = 3$, $early\_stopping$ = True, $no\_repeat\_ngram\_size$ = 3. Tokenizer length S_m was bounded as follows: $16 \leq T_m \leq 256$.

F. More Results and Discussions

F.1. Results and Discussion

Supervised training leads to more accurate video temporal segmentation results The performance of video temporal segmentation has a great impact on the final performance, so in this section, we compare the performance of VTS with several baselines: Histogram Intersect [40], Moment Invariant [32], Twin Comparison [115], PySceneDetect [9], and LGSS [82]. The results, displayed in Table 7, indicate that LGSS outperforms the other baselines but falls short when compared to our model. Both our method and LGSS are trained using a supervised approach, which leads to improved performance compared

Table 7. Comparison of video temporal segmentation results.

Model	Average Precision (AP) ↑	F1 ↑	M_{iou} ↑	Recall@3s ↑	Recall@5s ↑	Recall@10s ↑
Histogram Intersect	0.142	0.153	0.221	0.168	0.216	0.296
Moment Invariant	0.081	0.089	0.164	0.101	0.129	0.177
Twin Comparison	0.133	0.140	0.208	0.150	0.193	0.266
PySceneDetect	0.135	0.124	0.211	0.119	0.152	0.199
LGSS	0.243	0.352	0.216	0.163	0.216	0.272
Ours	**0.503**	**0.423**	**0.223**	**0.325**	**0.341**	**0.366**

Table 8. Comparison of video summarization results (whole video setting and segment-level setting).

Setting	Model	RMSE ↓	PSNR ↑	SSIM ↑	SRE ↓	Precision ↑	Recall ↑	F1 Score ↑
Whole-video	Uniform	0.479	4.044	0.076	49.808	0.077	0.100	0.049
	K-means	0.348	8.234	0.055	46.438	0.072	0.182	0.103
	SIFT	0.330	8.497	0.046	45.949	0.047	0.125	0.059
	VSUMM	0.279	9.226	0.053	44.862	0.054	0.259	0.088
	Ours	**0.112**	**25.280**	**0.697**	**23.550**	**0.320**	**0.290**	**0.321**
Segment-level	Uniform	0.237	6.307	0.085	42.495	0.186	0.179	0.105
	K-means	0.167	10.123	0.144	46.533	0.123	0.172	0.143
	SIFT	0.114	10.816	0.178	41.634	0.079	0.079	0.079
	VSUMM	0.122	18.818	0.258	41.601	0.160	0.207	0.171
	Ours	**0.091**	**36.370**	**0.698**	**23.430**	**0.333**	**0.275**	**0.255**

Table 9. Comparison of textual summarization results (whole video setting and segment-level setting).

Setting	Model	BLEU-1 ↑	ROUGE-1 ↑	ROUGE-2 ↑	ROUGE-L ↑	METEOR ↑	CIDEr ↑	SPICE ↑	BertScore ↑
Whole-video	BERT2BERT [100]	22.59	3.75	0.45	3.41	5.65	1.76	2.91	71.12
	BART-large-CNN [41]	29.17	3.19	0.51	3.04	2.99	2.28	11.27	68.84
	BART-large-XSUM [41]	30.91	3.83	0.57	3.59	3.99	2.56	3.71	69.56
	Distilbart [91]	26.46	3.87	3.87	0.47	3.59	2.25	4.16	69.37
	T5 [80]	25.39	3.51	0.43	3.21	4.51	1.97	5.66	70.38
	Pegasus [117]	26.73	3.75	0.52	3.40	4.52	2.38	7.82	68.92
	LED [6]	26.47	3.81	0.25	3.51	3.45	1.78	6.72	68.45
	Ours	**32.61**	**9.41**	**2.86**	**9.15**	4.01	4.01	10.11	**74.46**
Segment-level	BERT2BERT [100]	13.58	4.70	1.95	4.53	28.59	11.73	10.13	71.76
	BART-large-CNN [41]	22.79	6.45	2.46	6.32	26.21	20.64	10.13	71.44
	BART-large-XSUM [41]	20.89	7.31	2.77	7.13	29.36	20.90	10.20	71.42
	Distilbart [91]	14.77	1.95	0.15	1.87	23.52	11.83	10.53	66.46
	T5 [80]	16.48	6.17	3.03	5.99	28.22	20.96	10.35	71.95
	Pegasus [117]	16.17	3.41	0.96	3.29	29.82	17.26	10.39	67.81
	LED [6]	16.03	3.80	0.60	3.64	29.81	15.85	10.99	68.46
	Ours	**23.36**	**13.61**	**4.58**	**13.24**	**30.01**	**21.06**	10.28	**85.19**

to unsupervised baselines. Moreover, our approach incorporates attention mechanisms, potentially contributing to better results.

Supervised methods outperform unsupervised methods on video summarization In our video summarization study, we have chosen the following methods as our baseline comparisons: Uniform Sampling [33], K-means Clustering [26], Scale Invariant Feature Transform (SIFT) [58], and VSUMM [16]. The results, presented in Table 8, are under various evaluation metrics. For RMSE and SRE, lower values indicate better performance, whereas for the remaining metrics, higher values are desirable. From Table 8, we can observe that VSUMM showcases the strongest performance among the baseline methods, yet it still falls short compared to our proposed method. But we can conclude that supervised methods outperform unsupervised methods.

Pretrained large language models can still do well in text summarization In the context of textual summarization, we have considered a set of representative models as our baseline comparisons: BERT2BERT [100], BART [41] (including BART-large-CNN and BART-large-XSUM), Distilbart [91], T5 [80], Pegasus [117], and Longformer Encoder-Decoder (LED) [6]. The performance of these models is summarized in Table 9. Among the baselines, T5, BART-large-XSUM, BART-large-CNN, and BERT2BERT exhibit superior performance, with T5 demonstrating relatively better results across various text evaluation metrics. It is worth noting that the ROUGE score may not effectively capture performance differences

Table 10. Comparison of MSMO results.

Methods	Text					Video				
	BLEU ↑	METEOR ↑	CIDEr ↑	SPICE ↑	BertScore ↑	PSNR ↑	SSIM ↑	Precision ↑	Recall ↑	F1 Score ↑
LGSS + VSUMM + T5	27.35	24.32	3.94	5.57	62.77	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BART-large-XSUM	24.83	24.12	4.37	8.86	39.20	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BERT2BERT	13.26	34.83	3.68	9.23	64.34	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BART-large-CNN	24.93	32.61	4.18	11.84	64.44	16.234	0.198	0.143	0.152	0.147
Ours	**33.36**	30.31	4.06	10.28	**85.19**	**36.370**	**0.298**	0.133	**0.275**	**0.155**

compared to other evaluation metrics, which tend to provide more meaningful variations in performance.

MSMO results may depend on segmentation results and summarization methods In the field of MSMO, we encountered limitations in accessing the codebases of existing works such as [10, 19, 20, 30, 113, 130]. Therefore, we independently implemented several baselines to evaluate their performance on the MMSum dataset. For this purpose, we utilized LGSS as the segmentation backbone, VSUMM as the video summarizer, and selected text summarizers that exhibited the best performance in text summarization. The results are presented in Table 10. Based on the findings, it is evident that the aforementioned combination approaches still fall short in comparison to our proposed method. This also indicates that the accuracy of temporal segmentation is crucial prior to generating summaries, highlighting it as a critical step and task preceding MSMO.

G. More Related Work

Video Temporal Segmentation aims at splitting the video into segments based on predefined rules, which is a fundamental step in video analysis. Previous work either formed a classification problem to detect the segment boundaries in a supervised manner [1, 71, 92, 93, 126] or in the unsupervised way [23, 95]. Temporal segmentation of actions in videos is also widely explored in previous works [38, 39, 87, 103, 104, 124]. Video shot boundary detection and scene detection tasks are also relevant and has been explored in many previous studies [11, 27, 28, 82, 116], which aim at finding the visual change or scene boundaries.

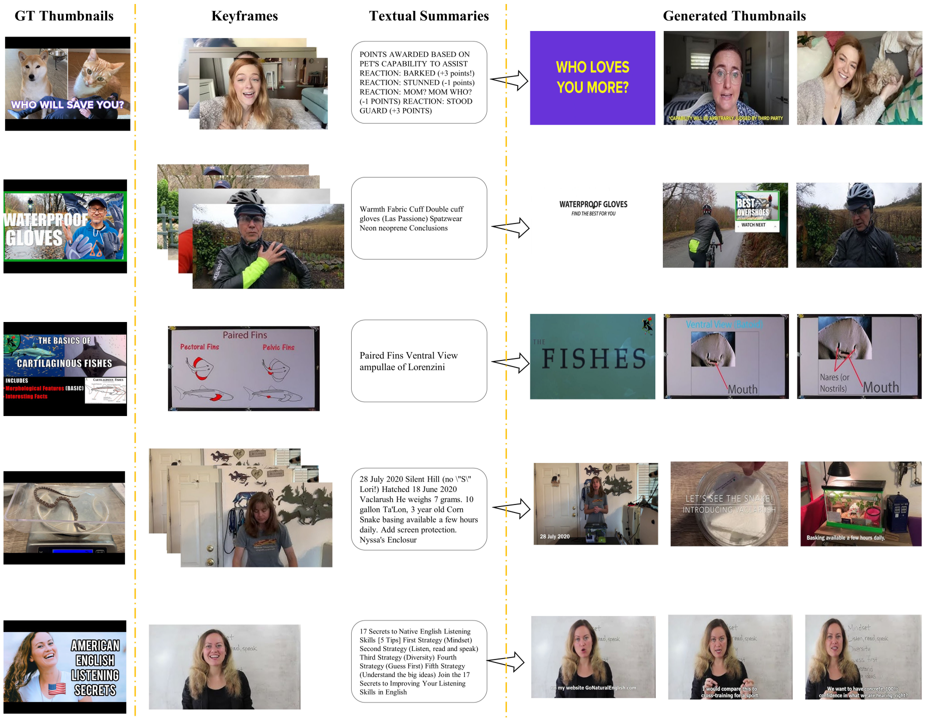
Video Summarization aims at extracting key moments that summarize the video content by selecting the most informative and vital parts, which lie in two directions: unimodal and multimodal approaches. Unimodal methods only use the visual modality, while multimodal methods exploit the available textual metadata and learn semantic or category-driven summarization. The summary usually contains a set of representative video keyframes that have been stitched in chronological order [2]. Traditional video summarization methods only use visual information, while recently, some category-driven or supervised approaches were proposed to generate video summaries with video-level labels [25, 63, 94, 107, 109, 127, 128].

Text Summarization takes textual metadata, i.e., documents, articles, tweets, etc, as input, and generates textual summaries in two directions: abstractive or extractive summarization. Abstractive methods select words based on semantic understanding, even the words may not appear in the source [89, 97]. Extractive methods attempt to summarize language by selecting a subset of words that retain the most critical points, which weights the essential part of sentences to form the summary [65, 106]. Recently, the fine-tuning approaches have improved the quality of generated summaries based on pre-trained language models in a wide range of tasks [48, 121].

Multimodal Learning has advanced quickly in recent years with appealing applications in different fields, i.e., embodied learning [7, 31, 34, 56, 75], multimedia image/video and language understanding [72, 78, 84, 132], and healthcare [24, 47, 73, 74, 76]. Thanks to the larger datasets [70, 79, 88, 111] and larger transformer models [8, 12, 15, 44, 114], many powerful multimodal models have been developed and shown great capability.

H. More Thumbnail Results

In the following Figures 10, 11, and 12, we show more comparisons of our generated thumbnails with Ground-Truth (GT) thumbnails provided by the authors of the video. We can find that our generated thumbnails can be very informative. Besides, we also provided some not-good examples in Figures 13, showing the potential of this new task and lots of room for improvement.



POINTS AWARDED BASED ON PET'S CAPABILITY TO ASSIST REACTION: BARKED (+3 points!) REACTION: STUNNED (-1 points) REACTION: MOM? MOM WHO? (-1 POINTS) REACTION: STOOD GUARD (+3 POINTS)

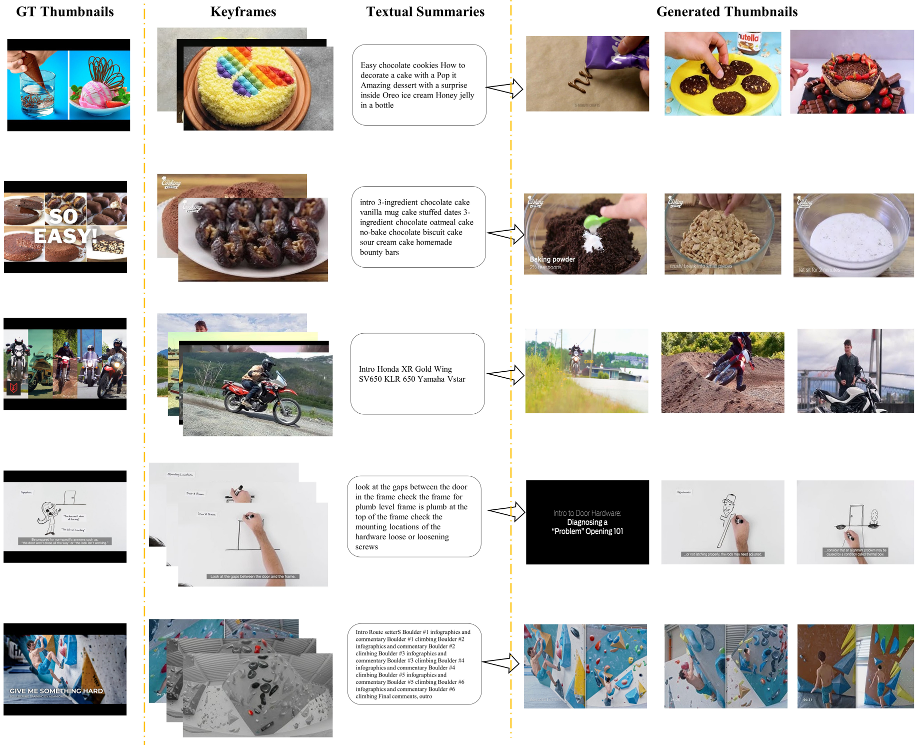
Warmth Fabric Cuff Double cuff gloves (Las Passione) Spatzwear Neon neoprene Conclusions

Paired Fins Ventral View ampullae of Lorenzini

28 July 2020 Silent Hill (no \"S\" Lori!) Hatched 18 June 2020 Vaclarush He weighs 7 grams. 10 gallon Ta'Lon, 3 year old Corn Snake basing available a few hours daily. Add screen protection. Nyssa's Enclosur

17 Secrets to Native English Listening Skills [5 Tips] First Strategy (Mindset) Second Strategy (Listen, read and speak) Third Strategy (Diversity) Fourth Strategy (Guess First) Fifth Strategy (Understand the big ideas) Join the 17 Secrets to Improving Your Listening Skills in English

Figure 10. More comparison of GT thumbnails and our generated ones [1/3].



Easy chocolate cookies How to decorate a cake with a Pop it Amazing dessert with a surprise inside Oreo ice cream Honey jelly in a bottle

intro 3-ingredient chocolate cake vanilla mug cake stuffed dates 3-ingredient chocolate oatmeal cake no-bake chocolate biscuit cake sour cream cake homemade bounty bars

Intro Honda XR Gold Wing SV650 KLR 650 Yamaha Vstar

look at the gaps between the door in the frame check the frame for plumb level frame is plumb at the top of the frame check the mounting locations of the hardware loose or loosening screws

Intro Route setterS Boulder #1 infographics and commentary Boulder #1 climbing Boulder #2 infographics and commentary Boulder #2 climbing Boulder #3 infographics and commentary Boulder #3 climbing Boulder #4 infographics and commentary Boulder #4 climbing Boulder #5 infographics and commentary Boulder #5 climbing Boulder #6 infographics and commentary Boulder #6 climbing Final comments, outro

Figure 11. More comparison of GT thumbnails and our generated ones [2/3].

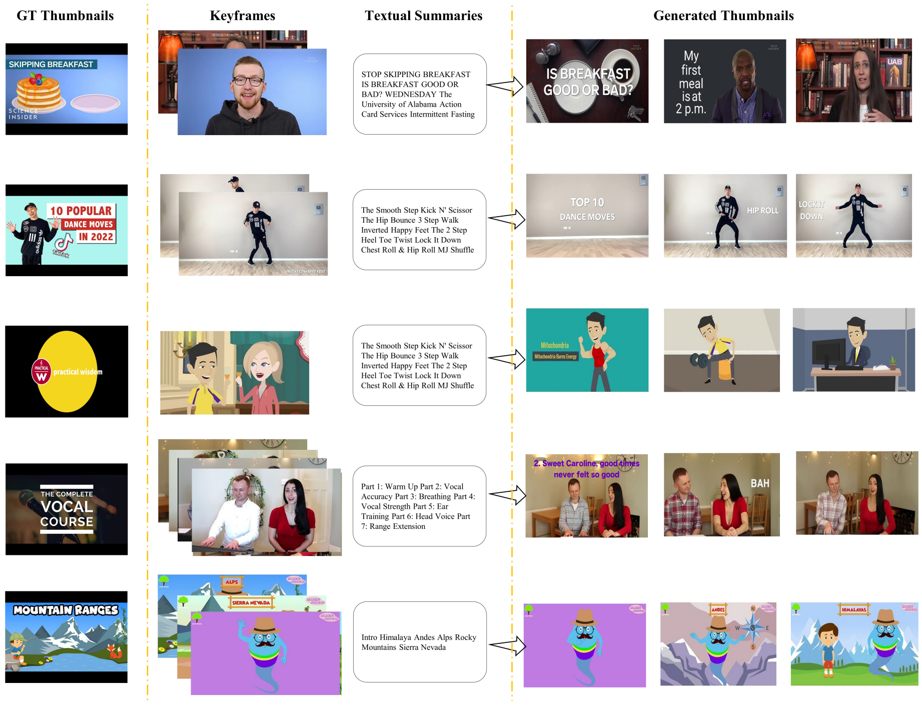

GT Thumbnails
Keyframes
Textual Summaries
Generated Thumbnails
SKIPPING BREAKFAST
STOP SKIPPING BREAKFAST IS BREAKFAST GOOD OR BAD? WEDNESDAY The University of Alabama Action Card Services Intermittent Fasting
IS BREAKFAST GOOD OR BAD?
My first meal is at 2 p.m.
10 POPULAR DANCE MOVES IN 2022
The Smooth Step Kick N' Scissor The Hip Bounce 3 Step Walk Inverted Happy Feet The 2 Step Heel Toe Twist Lock It Down Chest Roll & Hip Roll MJ Shuffle
TOP 10 DANCE MOVES
HIP ROLL
LOCKIT DOWN
practical wisdom
The Smooth Step Kick N' Scissor The Hip Bounce 3 Step Walk Inverted Happy Feet The 2 Step Heel Toe Twist Lock It Down Chest Roll & Hip Roll MJ Shuffle
Mitochondria
THE COMPLETE VOCAL COURSE
Part 1: Warm Up Part 2: Vocal Accuracy Part 3: Breathing Part 4: Vocal Strength Part 5: Ear Training Part 6: Head Voice Part 7: Range Extension
2. Sweet Caroline, good times never felt so good
BAH
MOUNTAIN RANGES
ALPS
SIERRA NEVADA
Intro Himalaya Andes Alps Rocky Mountains Sierra Nevada
ANDES
HIMALAYAS

Figure 12. More comparison of GT thumbnails and our generated ones [3/3].

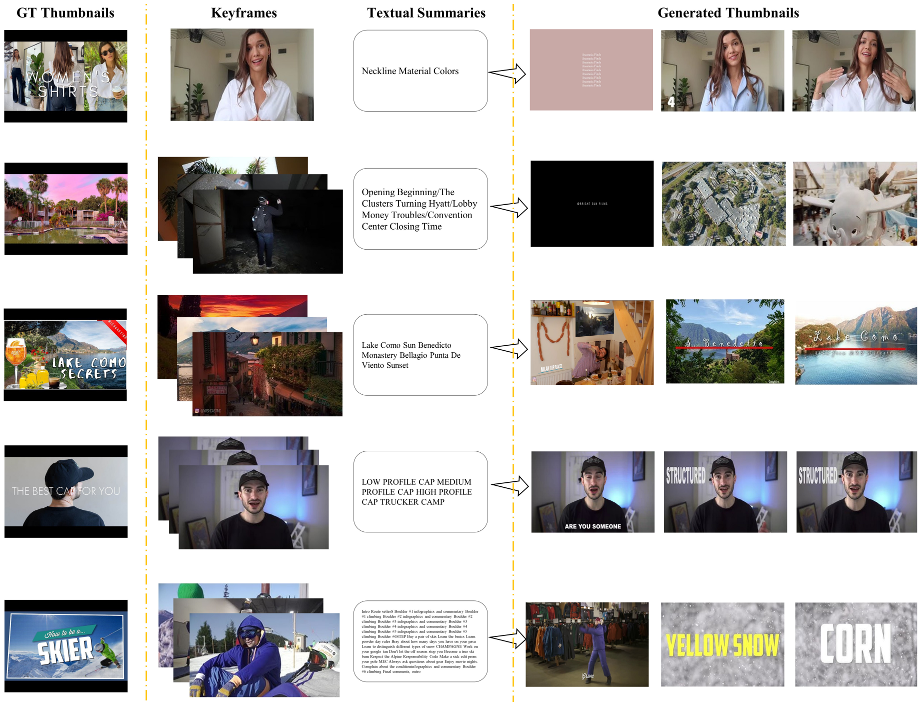

GT Thumbnails
Keyframes
Textual Summaries
Generated Thumbnails
WOMEN'S SHIRTS
Neckline Material Colors
Opening Beginning/The Clusters Turning Hyatt/Lobby Money Troubles/Convention Center Closing Time
LAKE COMO SECRETS
Lake Como Sun Benedicto Monastery Bellagio Punta De Viento Sunset
S. Benedetto
Lake Como
THE BEST CAP FOR YOU
LOW PROFILE CAP MEDIUM PROFILE CAP HIGH PROFILE CAP TRUCKER CAMP
ARE YOU SOMEONE
STRUCTURED
STRUCTURED
How to be a... SKIER
Intro Route setterS Boulder #1 infographics and commentary Boulder #1 climbing Boulder #2 infographics and commentary Boulder #2 climbing Boulder #3 infographics and commentary Boulder #3 climbing Boulder #4 infographics and commentary Boulder #4 climbing Boulder #5 infographics and commentary Boulder #5 climbing Boulder #6STEP Buy a pair of skis Learn the basics Learn powder day rules Bray about how many days you have on your pasu Learn to distinguish different types of snow CHAMPAGNE Work on your google tan Don't let the off season stop you Become a true ski bum Respect the Alpine Respomsibility Code Make a sick edit prom your pole MEC Always ask questions about gear Enjoy movie nights. Complain about the conditioninfographics and commentary Boulder #6 climbing Final comments, outro
YELLOW SNOW
CORN

Figure 13. More comparison of GT thumbnails and some “bad” generated ones.