Contextual AD Narration with Interleaved Multimodal Sequence

Hanlin Wang[1,3] Zhan Tong[2] Kecheng Zheng[3] Yujun Shen[3] Limin Wang[1†]

[1]State Key Laboratory for Novel Software Technology, Nanjing University
[2]ESAT, KU Leuven [3]Ant Group

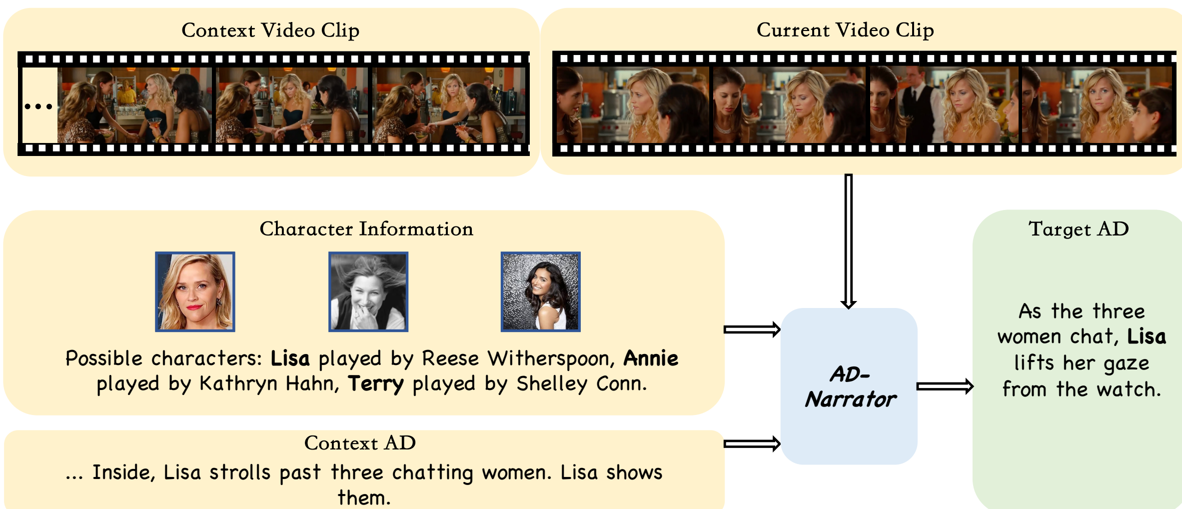

Figure 1. Taking video clip, text, character bank and context information as the inputs, the narrator generates corresponding audio description (AD) for video comprehension. Rather than describe all characters appearing in the video, the narrator should focus on characters that truly contribute to the storyline.

Abstract

The Audio Description (AD) task aims to generate descriptions of visual elements for visually impaired individuals to help them access long-form video content, like movies. With video feature, text, character bank and context information as inputs, the generated ADs are able to correspond to the characters by name and provide reasonable, contextual descriptions to help audience understand the storyline of movie. To achieve this goal, we propose to leverage pre-trained foundation models through a simple and unified framework to generate ADs with interleaved multimodal sequence as input, termed as Uni-AD. To enhance the alignment of features across various modalities with finer granularity, we introduce a simple and lightweight module that maps video features into the textual feature space. Moreover, we also propose a character-refinement module to provide more precise information by identifying the main characters who play more significant roles in the video context. With these unique designs, we further incorporate contextual information and a contrastive loss into our architecture to generate smoother and more contextually appropriate ADs. Experiments on multiple AD datasets show that Uni-AD performs well on AD generation, which demonstrates the effectiveness of our approach. Our code is available at: https://github.com/ant-research/UniAD.

1. Introduction

Audio Description (AD) [1, 14, 39, 46] provides descriptive narration of visual content in videos. Unlike subtitle or transcription, AD focuses more on describing the scene, characters, actions and storyline of the input video. As a rich visual description, AD can effectively supplement the dialogue and provide viewers with a comprehensive

†Corresponding author.

description of the video content, which not only helps the visual impairments better engage with video content [1], but also benefits the sighted individuals in their media comprehension activities [21, 34], such as language learning for kids and sight-free video consuming while driving. Despite that AD is important for video comprehension, particularly for those professionally produced media contents (movies, TV series *etc*.), currently most videos do not have corresponding AD [5], mainly due to the considerable costs of manual annotation and differences in understanding between annotators [16]. Therefore, studying how to generate ADs automatically is quite meaningful and necessary.

With the advances in computer vision and natural language processing, nowadays the research community is paying growing attention to generating ADs automatically, which requires a model to understand multi-modal information and perform contextual reasoning over video storyline [15, 47]. Compared with the conventional video captioning task [9, 11, 19, 30, 37], Audio Description (AD) is not only a scene description of the video clip, but also a narration that includes characters' names and actions to generate a coherent plot description, as shown in Fig. 1. This brings two characteristics of the AD generating task: (i) Multiple modality inputs. Video clip, text, character portraits and names are provided for AD generating. (ii) Rich contextual information. Context video and past AD can be applied to assist the current AD generation.

Previous methods [15, 16] introduce learnable adapters into GPT-2 to generate ADs. However, the amount of parameters in these adapters will rapidly increase with larger LLM [16], thus not conducive to scaling up. Instead, AutoAD-III [17] applied Q-former architecture to bridge the visual space with the language space. Training-free methods like MM-Narrator [47] and AutoAD-Zero [46] directly prompt GPT-4 or LLaMA3 [13] with specialized expert tools for AD generation, which suffers from complex prompt engineering and hallucination problem. In this work, we propose Uni-AD, a simple architecture that takes interleaved multimodal sequence as input to leverage completely open-source LLMs [35, 43] for AD generation by aligning various modality inputs to a unified semantic space. Formulating data as interleaved multimodal sequence makes it convenient to integrate various modality inputs and add contextual information for AD generation. Besides, interleaved multimodal sequence converts visual elements into multiple embedding tokens while maintaining the relative order between data, which ensures embeddings and tokens with the same semantic are naturally close to each other so that more fine-grained feature alignment can be learned spontaneously.

Given that the development of storyline is always character-centered [20, 42], it is necessary for ADs to include character names to describe their expressions, actions and status. Previous method [16] tried to identify all characters appearing in the given video as character information, without considering who are the main story drivers that should be included in AD. For example, in Fig. 1, though there are multiple characters in the scene, the expression change of *Lisa* is the main content thus only her name is involved in the target AD. With such an observation, we in this paper design a character-refinement module to figure out the AD-related characters. After training, this module can be applied to any videos to recognize main characters who contribute to the storyline and provide more accurate character information.

We further investigate the contextual information on our framework by combining past visual contents and ADs into the interleaved multimodal sequence, rather than only concatenating past ADs like previous works [15, 16, 47]. We find that when the input video is similar to the past video clip, the model tends to generate identical ADs. To address this, we introduce a contrastive loss as an auxiliary to avoid repetition and encourage diversity in AD generation.

To sum up, we develop an AD narration system called Uni-AD which achieves finer-grained feature alignment and supports extension to larger LLMs by formulating multiple inputs as interleaved multimodal sequence. To produce more accurate, coherent audio descriptions, we introduce a character-refinement module and incorporate contextual information along with a contrastive loss. Our Uni-AD outperforms previous methods on multiple AD datasets.

2. Related Work

2.1. Audio Descriptions Generation

Audio Description (AD) describes the key visual elements in videos to form coherent storyline narration. With the development of media technology, captioning for videos has emerged as a growing area in the computer vision research community [2, 19, 25]. Nonetheless, the production of Audio Description (AD) for video content remains a relatively untapped area of research. Initial works designed specialized authoring tools [6] and evaluation mechanisms [27, 31] to collect manually annotated ADs. Several annotation platforms like LiveDescribe [6], Rescribe [33] also emerged to facilitate AD generation. Recently, some works have studied how to generate AD at scale automatically with deep learning models. AutoAD-I [15] introduced the task of AD generation for movies and addressed it by prompting GPT-2 with learnable visual prompt vectors. [16] later incorporated an external character bank to provide character information for more accurate AD generation. Researchers further applied Q-former architecture to bridge the visual and language space [17] for this task. Training free methods [46, 47] proposed designs which extract information from inputs with multimodal experts and queries

GPT-4 [32] or VideoLLaMA2 [10] in a few-shot manner. However, these methods suffer from drawbacks like poor scalability, complex prompt engineering and weak modality alignment, which we in this paper address with the interleaved multimodal sequence design and larger LLM.

2.2. Interleaved Sequence for Multimodal Learning

Traditional vision language datasets for multimodal learning are mainly composed of image-text pairs collected from Internet [36, 38]. Text contents in these datasets are mostly short, less descriptive and independent, resulting in relatively poor text embeddings. Recent works like Flamingo [4], BLIP-2 [22] and CM3 [3] presented to conduct learning on the entire multimodal webpages, formulating interleaved images, videos and text as cohesive sequences. Such sequences offer long-form visual-text pairs for modeling and naturally retain the semantic correlation between different modality information, boosting the development of multi-modal learning. Other works [12, 41, 45] conducting pre-training on large amounts of interleaved multi-modal data further demonstrate the importance and effectiveness of this way. Inspired by these works, we cast inputs for audio description as interleaved multimodal sequences, leveraging the semantic relevance to achieve finer-grained modality alignment.

2.3. LLMs for Video Understanding

The recent surge in large language models (LLMs) [7, 32, 35, 43, 49] has inspired the study of video perception and understanding with LLMs. Models like ChatCaptioner [8], VideoChat [23] and MM-Vid [26] integrate visual experts with LLMs to construct multimodal perception systems for video representation, long-term video comprehension and dialog-centric interaction, *etc*. All these works can be divided into three categories: (i) Prompt tuning [8, 29] is a lightweight approach to transfer LLMs to downstream tasks with learnable prompt vectors. (ii) Adapter-based methods [23] typically insert additional trainable parameters into the LLM at different positions to achieve deep modality alignment, but the amount of introduced parameters increases with the size of LLM, making adapter-based methods difficult to scale up. (iii) Querying LLMs in a training free manner [26], which employs visual experts to transform video into text, thereby guiding LLMs in reasoning on specific tasks. Our Uni-AD follows the visual-conditioned prompt tuning manner to extend our approach to larger LLMs with its memory-friendly characteristic to generate better ADs.

3. Methodology

The audio description (AD) task is challenging mainly due the requirement that generated results should include characters' names to depict their expressions, actions to advance the plot and should be contextually coherent within the storyline. To meet this, we present Uni-AD to formulate various inputs for current video clip along with contextual information as interleaved multimodal sequence and combines a character-refinement module. In this section, we will give a detailed description of our approach. First, we provide an overview of Uni-AD in Sec. 3.1. Then we present how to prompt a LLM for AD generation in Sec. 3.2. Next, we show the design and training details of our proposed character-refinement module in Sec. 3.3. Afterwards, we describe the usage of contextual information for AD generation in Sec. 3.4. Finally, in Sec. 3.5 we further introduce the learning object of our framework.

3.1. Method Overview

The overall framework of our Uni-AD is illustrated in Fig. 2, which contains two key stages: visual modality alignment and multimodal prompt generation. Given a video clip and its corresponding character, contextual information, our model first filters the input character information to retain the AD-related individuals. Then we map all visual contents into embeddings, which will be combined with text tokens to create prompt for LLM. Contextual information can also be involved in this sequence. Finally, this interleaved multimodal prompt will be fed to a frozen LLM to generate audio description. Contrastive loss can be applied to avoid generating duplicate ADs.

3.2. Uni-AD Pipeline

Visual Mapping Network. The visual input for AD generation includes two components: (1) video clip consisting of N frames with timestamp t, denoted as $v_t = \{\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_N\}$; (2) characters' portrait images, denoted as $\{\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_C\}$, where C is the number of related characters in video clip v_t. To produce corresponding AD for v_t, we need to transform visual elements into embeddings with visual mapping network to achieve cross-modal alignment. Inspired by ClipCap [29] and AutoAD-I [15], we here apply a multi-layer transformer encoder with a fixed number of learnable vectors as our mapping network (shown in Fig. 3) based on the following findings: First, video frames are able to interact with each other to model the temporal relation via the attention mechanism, which is essential given that effective interaction between visual tokens is hard to achieve in a frozen LLM. Second, the introduction of learnable vectors allows us to control the length of the visual representation and using more embeddings to preserve visual details. Finally, such a structure requires no addition of adapters inserted into the LLM for modality alignment, making it easier to scale to larger LLMs compared to methods like AutoAD-II [15].

Specifically, we first extract visual features of the input video clip and character images with the pre-trained

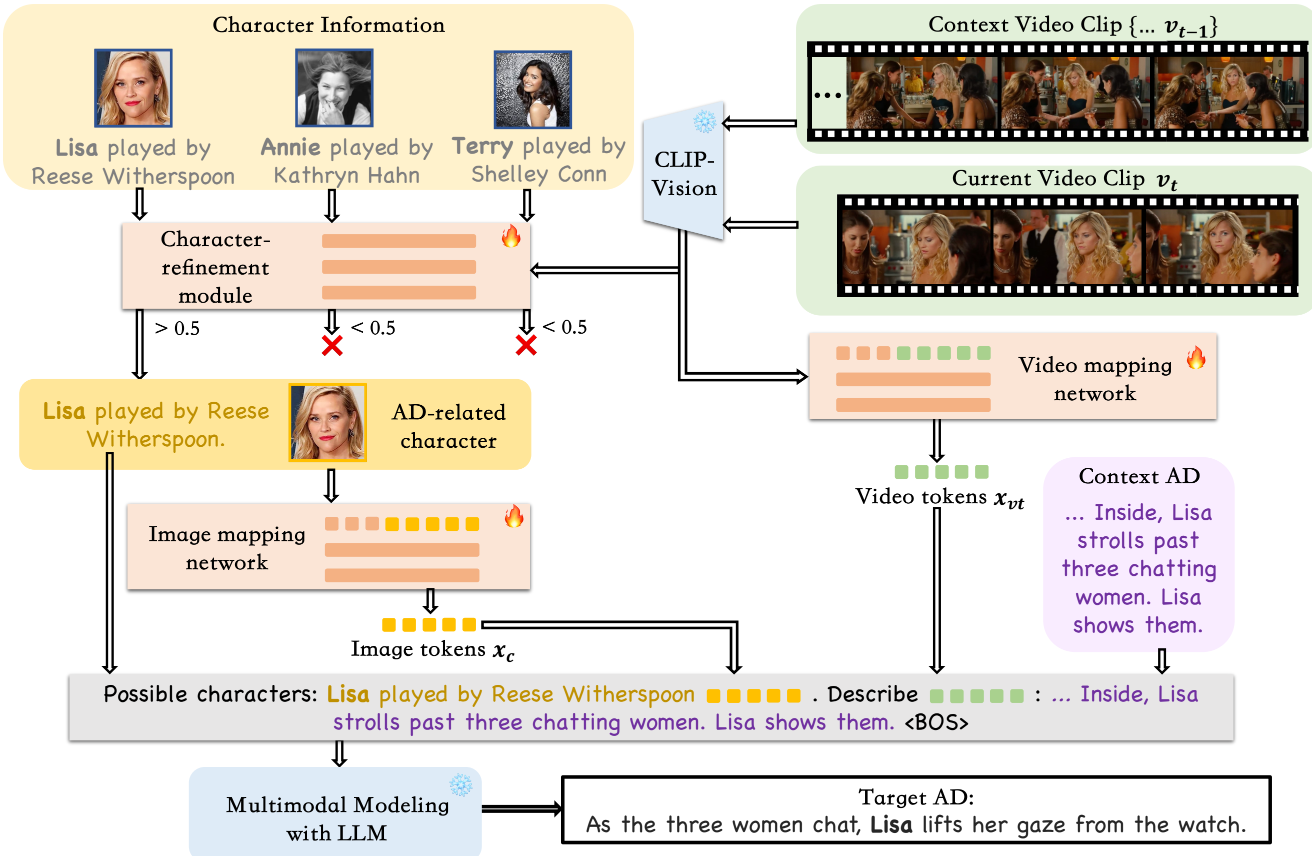

Figure 2. Overall architecture of our proposed Uni-AD. Our model first filters the input character information to retain the AD-related characters. Then all visual contents are mapped into the unified semantic space to form the interleaved multimodal sequence with text and contextual information. Afterwards, we prompt a frozen LLM with this sequence to generate the corresponding AD.

CLIP [36] visual encoder:

$$\begin{aligned} z_{v_t} &= f_{CLIP}(v_t), \\ \{z_1, z_2, ..., z_C\} &= f_{CLIP}(\{\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_C\}). \end{aligned} \tag{1}$$

In order to reduce the impact of difference between an actor's portrait and appearance in films, we follow [16] to adopt exemplar feature as character image information. The exemplar feature is obtained by averaging features of 5 frames that are most similar to the actor's portrait within the same movie. CLIP feature of the i_{th} character z_i is used as signature to compute similarity with movie CLIP features. Afterwards, we convert the visual input into embeddings with our visual mapping network $\mathcal{M}_v$:

$$\begin{aligned} x_{v_t} &= \mathcal{M}_v(Proj(z_{v_t})), \\ \{x_1, x_2, ..., x_C\} &= \mathcal{M}_v(Proj(e_1), ..., Proj(e_C)), \end{aligned} \tag{2}$$

where e_i denotes the i_{th} character's exemplar feature and $Proj$ represents a Linear Layer that transforms the channel number of visual features to match the LLM. Note that exemplar features are mapped via $\mathcal{M}_v$ separately, meaning there is no need for interaction between characters here.

Formulating Interleaved Multimodal Prompt. With visual embeddings x_{v_t} and $\{x_1, x_2, ..., x_C\}$, we now combine them with the text query to get our interleaved multimodal prompt. We apply the prompting template in [16] to query the frozen LLM, and our prompt is formulated as:

$\langle$ Possible characters: $char_1$ played by $actor_1$ x_1, $char_2$ played by $actor_2$ x_2, ... Describe x_{v_t}: $\rangle$,

where $char_i$, $actor_i$, x_i, x_{v_t} denote the i_{th} character's name, real actor name, image feature and video embeddings, respectively. Our main thought here is to represent visual content with multiple tokens and preserve the positional relationships between different modalities of information, hoping to achieve a finer-grained alignment.

3.3. Character-Refinement Module

Character information provides names, portraits of active characters to help model generate person-centric descriptions, making it important to be incorporated into AD generation. Previous work achieved this by introducing an external character bank [16]. With an aim to recognize all *active* characters who appear in the given video clip,

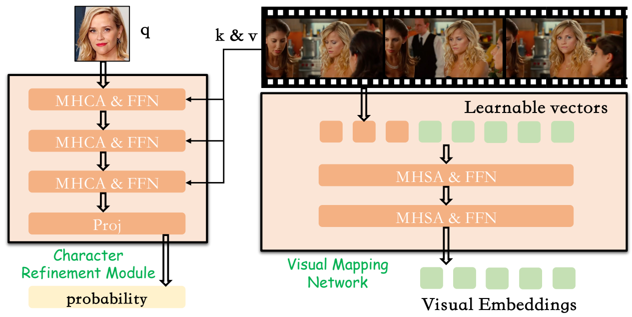

Figure 3. Structure of Visual Mapping Network and Character-Refinement Module.

researchers trained a character recognition module on the annotated MovieNet [18] dataset to predict the active characters given their exemplars and the movie clip. Then the output of this module is used to build the character bank to provide character information for AD generation.

Introducing the external character bank significantly improves the quality of AD generation. However, this approach overlooks the difference between *active* characters and *AD-related* characters. That is, a character who appears in the current video clip may not necessarily be mentioned in the corresponding AD, especially in scenes with multiple characters. ADs usually prioritize the main characters who drive the story forward and do not mention secondary characters to avoid overburdening audiences with too much information. In such cases, taking all characters appearing in the video as character information will confuse the AD narrator and generate descriptions not align with the development of the storyline.

Given observations mentioned above, we adapt our goal to identify the *AD-related* characters in the video clip based on their behavior and mannerisms. Since these characters are likely to appear in the video clip, we design a character-refinement module to perform further identification based on the external character bank provided by AutoAD-II [16]. Our character-refinement module consists of 3 Multi-Head Cross-Attention (MHCA) and FeedFoward Network (FFN) layers (shown in Fig. 3), which takes characters' exemplar features in external character bank as query, video features as key&value and outputs the probability for each character on whether they are AD-related. A projection layer is added to transform the output feature to a probability value. We train this module with a binary classification loss. Training labels are obtained from the AD annotations by retrieving all names that appear in both the movie's cast list and the annotation [16]. After trained, we apply this module to the test movies and treat characters whose probability exceeds 0.5 as AD-related characters.

3.4. Contextual Information Modeling

We in this section show how to use contextual information to generate more coherent ADs with our model. With the design of interleaved multimodal sequence, we can easily incorporate past context ADs and video clips into our prompt for LLM.

Context ADs contain descriptions of preceding story plot that leads up to the current scene, thus can help the model better follow the storyline for narration. We here utilize context AD by directly concatenating our prompt in Sec. 3.2 with the past K ADs $\{\mathcal{T}_{t-K}, \mathcal{T}_{t-K}, ..., \mathcal{T}_{t-1}\}$. In this way, we provide more text conditional information for AD generation. To separate context ADs from current AD, we add a BOS token in our prompt to start AD generation.

Although context ADs can provide the most accurate description of the preceding story plot, the ADs we generate during inference will inevitably differ from the ground truth, leading to an inaccurate guidance. Therefore, we consider to introduce past videos into our Uni-AD as contextual information. Specifically, we take frame features from the past K video clips $\{z_{v_{t-K}}, z_{v_{t-K+1}}, ..., z_{v_{t-1}}\}$ and concatenate them with the current video clip in temporal order. Then these concatenated visual features will be fed to the visual mapping network $\mathcal{M}_v$ for interaction, resulting in the contextual video representation $x_{t\_context}$:

$$x_{t\_context} = \mathcal{M}_v(Proj(z_{v_{t-K}}; z_{v_{t-K+1}}; ...; z_{v_t})), \quad (3)$$

where $[\cdot;\cdot]$ denotes the concatenation operation. Then we can use $x_{t\_context}$ instead of x_{v_t} as video embedding to get our video-context prompt. Fig. 2 shows our prompt with both visual and text contextual information.

In practice, we find that when the input video does not vary much from the past video clip, LLM tends to generate very similar, or even identical ADs. This is unreasonable for the AD generation task, since current AD should carry on the narration from previous content rather than repeating. To address this, we add a contrastive loss to our training process:

$$\begin{aligned} s &= \frac{\sum_n logP_\Theta(a_n|\text{prompt}; a_{<n})}{||\mathcal{T}||}, \\ \mathcal{L}_{ct} &= max(0, s_{last} - s_{current}), \end{aligned} \quad (4)$$

where $\mathcal{T}$ denotes an AD, a_n denotes the n_{th} token in $\mathcal{T}$, $a_{<n}$ denotes tokens preceding a_n in $\mathcal{T}$; prompt denotes our interleaved multimodal sequence fed to LLM; Θ denotes learnable parameters in our visual mapping network; s represents the average likelihood score of the generated AD. Our contrastive loss $\mathcal{L}_{ct}$ is calculated as Eq. (4). Thus our model is constrained to ensure that the score of generating last AD (s_{last}) is always lower than score of the current ground-truth AD ($s_{current}$). In this way, we encourage the model to generate more accurate and non-repetitive ADs.

3.5. Objective Function

Overall, given a video clip with timestamp t, our goal is to query a frozen LLM for AD generating with our visual-conditioned prompt. The supervision we apply is the commonly used auto-regressive loss function:

$$\mathcal{L}_{auto} = -\sum_{n} \log P_{\Theta}(a_n | \text{prompt}; a_{<n}), \quad (5)$$

where a_n denotes the n_{th} token in the target AD and $prompt$ denotes our interleaved multimodal prompt.

As mentioned in Sec. 3.4, we further introduce a contrastive loss to avoid repetitive AD generation. In this case, our complete loss is: $\mathcal{L}_{\Theta} = \mathcal{L}_{auto} + \mathcal{L}_{ct}$.

4. Experiments

In this section, we evaluate Uni-AD on multiple AD generation benchmarks and show the experiment results. We first introduce our implementation details in Sec. 4.1. Then we compare our performance with state-of-the-art AD generation approaches in Sec. 4.2. We further conduct a detailed ablation study on the impact of character-refinement module and visual mapping network on our model in Sec. 4.3. Next in Sec. 4.4, we confirm the effectiveness of incorporating contextual information into Uni-AD and show how different contextual information affects our model. Finally, we provide qualitative examples of our Uni-AD in Sec. 4.5.

4.1. Implementation Details

Dataset. We follow AutoAD-I [15] to conduct partial-data pre-training on the AudioVault-AD dataset [15]. We train our model on the MAD-v2-Named dataset and evaluate on MAD-eval-Named [15, 40]. For evaluation on CMDAD [17] and TVAD [46], we train our model with the CMDAD [17] training set. We use both classic captioning metrics and newly proposed metrics for evaluation. The former metrics include ROUGE-L [24] (R-L) and CIDEr [44] (C) to measure the quality of our generated ADs versus human-annotated ones. The latter metrics include R@k/N [16], CRITIC [17] and LLM-AD-eval [17]. More information about datasets and metrics is provided in the supplementary material.

Training details. We train our GPT-based model with a batch size of 96 movie clips and the learning rate is 10^{-3}, while our LLaMA-based model is trained with a batch size of 12 movie clips and the learning rate is 5.0×10^{-5}. We use the AdamW [28] optimizer to train our model for 10 epochs, with a cosine-decayed learning rate schedule and linear warm-up. All training is conducted on 8 A100 GPUs. For external character information, we use the prediction results from AutoAD-II [16] as input for our character refine module on MAD-eval-Named dataset and prediction results from AutoAD-Zero [46] as input for character refine module on CMDAD and TVAD. We apply the visual branch of VideoLLaMA [48] as visual mapping network on experiments of CMDAD and TVAD, as in [17].

Table 1. Comparison with the state-of-the-art methods on MAD-eval-Named. The *Context* column denotes whether contextual information is applied. The *V-Feature* column indicates the type of visual expert used for extracting movie frame features.

Methods	Context	LLM & V-Feature	RL↑	C↑	R@5/16↑
ClipCap [29]	✗	GPT-2 & CLIP-B32	8.5	4.4	36.5
AutoAD-I [15]	✗	GPT-2 & CLIP-B32	10.3	12.1	39.8
AutoAD-II [16]	✗	GPT-2 & CLIP-B32	13.1	19.2	51.3
AutoAD-III [17]	✗	LLaMA2 & EVA-CLIP	-	24.0	52.8
AutoAD-Zero [46]	✗	LLaMA3 & VideoLLaMA2	-	22.4	-
Uni-AD(ours)	✗	GPT-2 & CLIP-B32	15.9	24.0	50.5
Uni-AD(ours)	✗	GPT-2 & CLIP-L14	16.4	25.7	51.5
Uni-AD(ours)	✗	LLaMA2 & CLIP-L14	**16.8**	**27.3**	**53.3**
AutoAD-I [15]	✓	GPT-2 & CLIP-B32	11.9	14.3	42.1
AutoAD-II [16]	✓	GPT-2 & CLIP-B32	13.4	19.5	50.8
MM-Narrator [47]	✓	GPT-4 & CLIP-L14	13.4	13.9	49.0
MM-Narrator [47]	✓	GPT-4V & CLIP-L14	12.8	9.8	-
Uni-AD(ours)	✓	LLaMA2 & CLIP-L14	**17.1**	**28.2**	**54.2**

Table 2. Comparison with the state-of-the-art methods on CMDAD and TVAD. The gray row shows the results of AutoAD-III pretrained on the 3.4M HowTo-AD dataset, which is not public.

Method	Dataset	CIDEr↑	CRITIC↑	LLM-AD-eval↑
AutoAD-II [16]	CMDAD	13.5	8.2	2.08
AutoAD-III [17]	CMDAD	21.7	25.2	2.85
AutoAD-III [17]	CMDAD	25.0	32.7	2.92
AutoAD-Zero [46]	CMDAD	17.7	**43.7**	2.83
Uni-AD(Ours)	CMDAD	**21.8**	41.9	**2.92**
AutoAD-III [17]	TVAD	26.1	15.9	2.78
AutoAD-Zero [46]	TVAD	22.6	27.6	**2.94**
Uni-AD(Ours)	TVAD	**26.6**	**28.3**	2.89

4.2. Comparison with state-of-the-art approaches

Multiple models for AD generation are involved in our comparison. Descriptions for these methods are available in the supplementary material. Tab. 1 shows the evaluation results of our Uni-AD and these state-of-the-art methods on the MAD-eval-Named benchmark. For fair comparison, we first evaluate Uni-AD under the same setting (GPT-2 as language decoder and CLIP ViT-B/32 as visual feature) with previous methods [15, 16, 29]. Our Uni-AD outperforms all previous methods by a notable margin, which demonstrates the effectiveness of our model. Afterward, we apply a stronger CLIP ViT-L/14 model as visual encoder to extract movie features, and the results of Uni-AD show growth on all metrics. Utilizing the design of lightweight visual mapping network, we then extend our method to a more powerful LLM LLaMA2-7B, which further enhances the performance (1.6 growth points on *Cider*, 0.4 on *Rouge-L* and 1.8 on *Recall@5/16*).

Next, we conduct comparison with AD generation approaches under the contextual setting. Uni-AD achieves the state-of-the-art performance (17.1 on *Rouge-L*, 28.2 on *Cider* and 54.2 on *Recall@5/16*) by incorporating context

Table 3. Study the effectiveness of character-refinement module without pre-training or contextual information. *Char.?* shows whether this module is applied. * means our re-implemention.

Methods	Char.?	RL↑	C↑	R@5/16↑
AutoAD-II (GPT-2-B32) [16]	✗	14.7*	19.0*	46.0*
	✓	15.1	21.0	47.8
	gt	19.4	33.8	68.0
Uni-AD (GPT-2-B32)	✗	14.0	18.5	45.7
	✓	15.7	23.7	49.4
	gt	19.7	36.1	69.2
Uni-AD (GPT-2-L14)	✗	14.0	19.7	47.2
	✓	16.0	25.5	50.3
	gt	20.3	37.8	70.4
Uni-AD (LLaMA2-L14)	✗	15.3	22.6	**53.1**
	✓	**16.5**	**25.9**	52.5
	gt	21.2	40.0	72.0

video features and contrastive learning. We further evaluate Uni-AD on CMDAD and TVAD and obtain competitive results, shown in Tab. 2. These outstanding performances show the effectiveness and flexibility of our model.

4.3. Ablation Study

Study on character-refinement module. By recognizing the main characters contributed to the storyline, the character-refinement module provides more precise character information for AD generation. Evaluation results in Tab. 3 show the general performance improvements character-refinement module brings under different settings, confirming the universality of this module.

We further use characters involved in AD annotations as character information to conduct experiments (*gt* rows in Tab. 3). The gap between character-refinement results and *gt* results also show that the current character-refinement performance (0.41 on *Precision*, 0.77 on *Recall*, evaluated on MAD-eval-Named dataset) is far from sufficient and there is still significant room for improvement.

Study on Visual Mapping Network. To verify whether our Uni-AD can achieve finer-grained feature alignment, we in this section study how the visual mapping network affects AD generation. Our main comparison target is the Flamingo-style method AutoAD-II. Specifically, we adjust the number of latent vectors output by the visual mapping network, which also corresponds to the number of visual tokens fed to the LLM, to observe the impact on AD generation. The evaluation results are shown in Tab. 4. It can be seen that as the number of latent vectors increases, the performance of Uni-AD improves under both settings (GPT-CLIP-B32 and LLaMA-CLIP-L14), while the result of AutoAD-II method remains basically unchanged. This confirms our hypothesis that AutoAD-II, which achieves cross-modal alignment by concatenating characters' portraits with video frames followed by perceiver resampler and gated cross-attention modules, tends to extract global feature of the visual contents. In contrast, our Uni-AD is capable of retaining more visual details by increasing the number of latent vectors, thereby achieving finer-grained feature alignment and better AD generation results.

Table 4. Study the impact of visual mapping network on AD generation without pre-training or Contextual information. Character-refinement module is applied by all methods. *#Latent* denotes the number of learnable vectors in the visual mapping network.

Methods	#Latent	RL↑	C↑	R@5/16↑
AutoAD-II [16] (GPT-2-B32)	1	15.1	20.5	47.5
	5	15.1	20.8	48.0
	10	15.1	21.0	47.8
	30	15.1	20.1	47.4
Uni-AD (GPT-2-B32)	1	13.7	19.0	45.4
	5	15.3	22.8	47.9
	10	15.4	22.4	49.0
	30	15.7	23.7	49.4
Uni-AD (LLaMA-L14)	1	16.2	24.9	51.5
	5	16.3	24.9	52.0
	10	16.4	25.0	52.0
	30	**16.5**	**25.9**	52.5
	60	16.3	25.3	**53.7**

Table 5. Study on contextual information. We conduct experiments with our AudioVault pre-trained LLaMA-CLIP-L14 model.

Context-V	C-Loss	RL↑	C↑	R@5/16↑
0	✗	16.8	27.3	53.3
1	✗	16.8	27.5	54.7
3	✗	17.0	27.4	54.2
0	✓	16.9	27.3	53.7
1	✓	**17.1**	**28.2**	54.2
3	✓	16.9	27.3	**54.9**

4.4. Integrating Contextual Information

For contextual information, we first study the effectiveness of integrating context video and contrastive loss into AD generation. Results in Tab. 5 suggest that adding context video clips and contrastive loss can both enhance model's performance. Among these results, incorporating the most recent context video along with contrastive loss is the best, since current AD is most relevant to its preceding content.

Then we conduct experiments to introduce past ADs as additional contextual information under *recurrent* setting (using predicted past ADs) and *oracle* setting (using ground-truth past ADs). As shown in Tab. 6, the *recurrent* setting leads to a decrease in model performance. We attribute this to the discrepancy between the predicted ADs and the ground truth, which provides inaccurate information for AD generation during inference, thus further enlarges the gap between training and testing. For *oracle* setting, the performance of our model improves with more context ADs, indicating that context AD is the most significant influencing factor for AD generation.

4.5. Qualitative Results

We show our qualitative AD generation results on the MAD-eval dataset in Fig. 4. Specifically, we conduct analy-

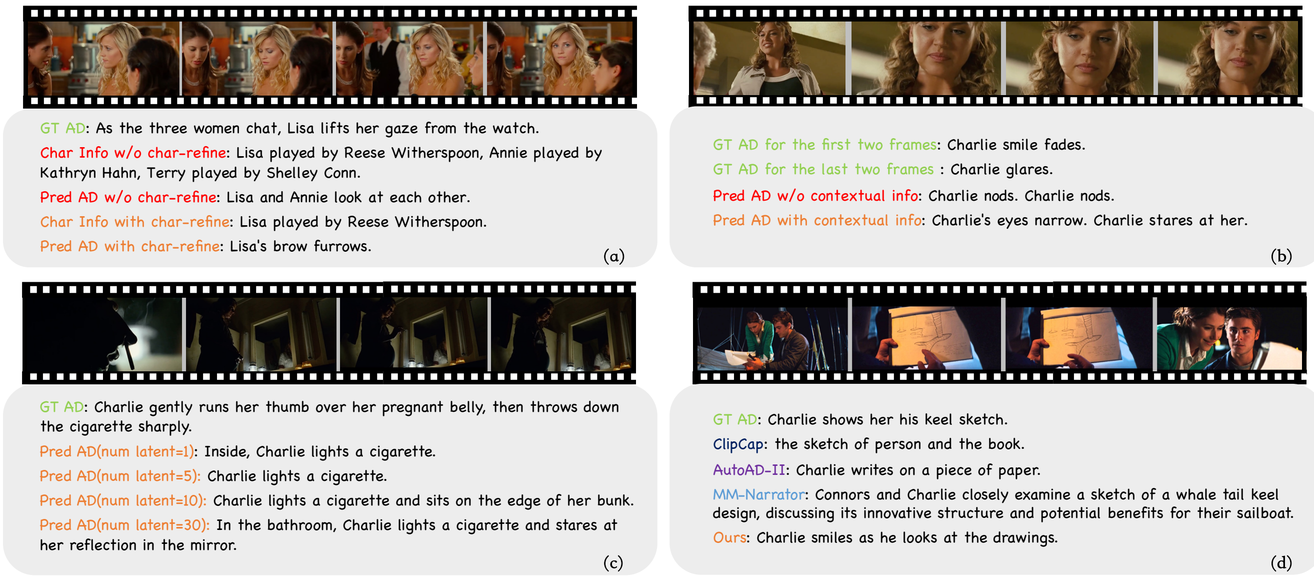

Figure 4. Qualitative analysis on character-refinement module, contextual information, number of learnable vectors and comparison with other approaches. Movies are selected from (a): How Do You Know(2010), (bc): Legion(2010), (d): Charlie St. Cloud (2010).

Table 6. Study the effectiveness of context AD. *Context-AD* denotes the number of past context ADs. *V&C-Loss* indicates whether past videos and contrastive loss are applied.

Context-AD	V&C-Loss	RL↑	C↑	R@5/16↑
1(recurrent)	✗	16.4	25.8	51.9
	✓	16.4	25.8	52.3
3(recurrent)	✗	15.6	23.1	50.9
	✓	16.1	24.0	52.2
1(oracle)	✗	17.7	31.3	55.3
	✓	17.9	31.7	56.5
3(oracle)	✗	18.6	34.9	55.9
	✓	18.6	34.8	55.6

sis on character-refinement module, contextual information and the number of learnable vectors in visual mapping network. Results show that: (1) The character-refinement module can recognize the AD-related people and provide more precise character information for AD generation. For example in (a), AD-narrator without this module mistakes the main character as *Lisa* and *Annie*, thus generates AD deviated from the ground truth. (2) Incorporating contextual information and the contrastive loss can effectively avoid repeated AD generation and get more coherent results. In sample (b) where the contents of two consecutive movie clips vary little, model without contextual information just generates two identical ADs, while descriptions with progressive relationship are generated when contextual information is available. (3) More visual tokens retain more visual details. In sample (c), the corresponding AD becomes more detailed as the number of learnable vectors increases. Finally, we compare our generated ADs with other methods in sample (d). The description generated by Uni-AD is more consistent with the video content than results of AutoAD-II and ClipCap. We find that result of MM-Narrator, which generates AD by prompting GPT-4, contains a lot of details that don't actually exist. We speculate that this is because GPT-4 is trained on longer texts with a large number of detailed descriptions, which results in a serious hallucination problem.

5. Conclusion

In this work, we present a simple and unified framework called Uni-AD for Audio Description (AD) generation task by prompting pre-trained LLMs with interleaved multi-modal sequence as input. Compared with previous work, our Uni-AD is able to leverage more precise character information provided by the character-refinement module and fully utilize rich contextual information to generated ADs. Uni-AD achieves the state-of-the-art performance on multiple AD generation benchmarks. We also conduct comprehensive ablation studies to validate the effectiveness of different components, which demonstrates that fine-grained feature alignment, precise character information, and contextual data can benefit AD generation. We hope our work could facilitate research in this community.

Acknowledgements: This work was supported by the National Key R&D Program of China (No. 2022ZD0160900), ERC AdG Project (No. 101021347), Ant Group Research Intern Program, Jiangsu Frontier Technology Research and Development Program (No. BF2024076), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] American council of the blind. https://adp.acb.org/. 1, 2

[2] Nayyer Aafaq, Ajmal Mian, Wei Liu, Syed Zulqarnain Gilani, and Mubarak Shah. Video description: A survey of methods, datasets, and evaluation metrics. *ACM Comput. Surv.*, 52(6):115:1–115:37, 2020. 2

[3] Armen Aghajanyan, Bernie Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, and Luke Zettlemoyer. CM3: A causal masked multimodal model of the internet. *CoRR*, abs/2201.07520, 2022. 3

[4] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 3

[5] Ava Bartolome and Shuo Niu. A literature review of video-sharing platform research in HCI. In *CHI*, pages 790:1–790:20. ACM, 2023. 2

[6] Carmen J. Branje and Deborah I. Fels. Livedescribe: Can amateur describers create high-quality audio description? *Journal of Visual Impairment & Blindness*, 106(3):154–165, 2012. 2

[7] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020. 3

[8] Jun Chen, Deyao Zhu, Kilichbek Haydarov, Xiang Li, and Mohamed Elhoseiny. Video chatcaptioner: Towards enriched spatiotemporal descriptions. *CoRR*, abs/2304.04227, 2023. 3

[9] Shaoxiang Chen and Yu-Gang Jiang. Towards bridging event captioner and sentence localizer for weakly supervised dense event captioning. In *CVPR*, pages 8425–8435. Computer Vision Foundation / IEEE, 2021. 2

[10] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *CoRR*, abs/2406.07476, 2024. 3

[11] Chaorui Deng, Shizhe Chen, Da Chen, Yuan He, and Qi Wu. Sketch, ground, and refine: Top-down dense video captioning. In *CVPR*, pages 234–243. Computer Vision Foundation / IEEE, 2021. 2

[12] Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, Xiangwen Kong, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Dreamllm: Synergistic multimodal comprehension and creation. *CoRR*, abs/2309.11499, 2023. 3

[13] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao,

Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The llama 3 herd of models, 2024. 2

[14] Louise Fryer. *An introduction to audio description: A practical guide*. Routledge, 2016. 1

[15] Tengda Han, Max Bain, Arsha Nagrani, Gül Varol, Weidi Xie, and Andrew Zisserman. Autoad: Movie description in context. In *CVPR*, pages 18930–18940. IEEE, 2023. 2, 3, 6

[16] Tengda Han, Max Bain, Arsha Nagrani, Gül Varol, Weidi Xie, and Andrew Zisserman. Autoad II: the sequel - who, when, and what in movie audio description. In *ICCV*, pages 13599–13609. IEEE, 2023. 2, 4, 5, 6, 7

[17] Tengda Han, Max Bain, Arsha Nagrani, Gül Varol, Weidi Xie, and Andrew Zisserman. Autoad III: the prequel - back to the pixels. In *CVPR*, pages 18164–18174. IEEE, 2024. 2, 6

[18] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *ECCV (4)*, pages 709–727. Springer, 2020. 5

[19] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715. IEEE Computer Society, 2017. 2

[20] Anna Kukleva, Makarand Tapaswi, and Ivan Laptev. Learning interactions and relationships between movie characters. In *CVPR*, pages 9846–9855. Computer Vision Foundation / IEEE, 2020. 2

[21] Elisa Lewis. Deep dive: How audio description benefits everyone, 2021. Accessed on: 2023-11-13. 2
[22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICLR*. JMLR.org, 2023. 3
[23] Kunchang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *CoRR*, abs/2305.06355, 2023. 3
[24] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 6
[25] 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 *CVPR*, pages 17928–17937. IEEE, 2022. 2
[26] Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, Ce Liu, and Lijuan Wang. MM-VID: advancing video understanding with gpt-4v(ision). *CoRR*, abs/2310.19773, 2023. 3
[27] Xingyu Bruce Liu, Ruolin Wang, Dingzeyu Li, Xiang Anthony Chen, and Amy Pavel. Crossa11y: Identifying video accessibility issues via cross-modal grounding. In *UIST*, pages 43:1–43:14. ACM, 2022. 2
[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR (Poster)*. OpenReview.net, 2019. 6
[29] Ron Mokady, Amir Hertz, and Amit H. Bermano. Clipcap: CLIP prefix for image captioning. *CoRR*, abs/2111.09734, 2021. 3, 6
[30] Jonghwan Mun, Linjie Yang, Zhou Ren, Ning Xu, and Bohyung Han. Streamlined dense video captioning. In *CVPR*, pages 6588–6597. Computer Vision Foundation / IEEE, 2019. 2
[31] Rosiana Natalie, Ebrima Jarjue, Hernisa Kacorri, and Kotaro Hara. Viscene: A collaborative authoring tool for scene descriptions in videos. In *ASSETS*, pages 87:1–87:4. ACM, 2020. 2
[32] OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. 3
[33] Amy Pavel, Gabriel Reyes, and Jeffrey P. Bigham. Rescribe: Authoring and automatically editing audio descriptions. In *UIST*, pages 747–759. ACM, 2020. 2
[34] Elisa Perego. Gains and losses of watching audio described films for sighted viewers. *Target*, 28(3):424–444, 2016. 2
[35] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2, 3
[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 3, 4
[37] Tanzila Rahman, Bicheng Xu, and Leonid Sigal. Watch, listen and tell: Multi-modal weakly supervised dense event captioning. In *ICCV*, pages 8907–8916. IEEE, 2019. 2
[38] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL (1)*, pages 2556–2565. Association for Computational Linguistics, 2018. 3
[39] Joel Snyder. Audio description: The visual made verbal. In *International Congress Series*, pages 935–939. Elsevier, 2005. 1
[40] Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba Heilbron, Chen Zhao, Silvio Giancola, and Bernard Ghanem. MAD: A scalable dataset for language grounding in videos from movie audio descriptions. In *CVPR*, pages 5016–5025. IEEE, 2022. 6
[41] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. *CoRR*, abs/2307.05222, 2023. 3
[42] Makarand Tapaswi, Marc T. Law, and Sanja Fidler. Video face clustering with unknown number of clusters. In *ICCV*, pages 5026–5035. IEEE, 2019. 2
[43] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288, 2023. 2, 3
[44] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575. IEEE Computer Society, 2015. 6
[45] Alex Jinpeng Wang, Linjie Li, Kevin Qinghong Lin, Jianfeng Wang, Kevin Lin, Zhengyuan Yang, Lijuan Wang, and Mike Zheng Shou. COSMO: contrastive streamlined multimodal model with interleaved pre-training. *CoRR*, abs/2401.00849, 2024. 3
[46] Junyu Xie, Tengda Han, Max Bain, Arsha Nagrani, Gül Varol, Weidi Xie, and Andrew Zisserman. Autoad-zero: A training-free framework for zero-shot audio description. *CoRR*, abs/2407.15850, 2024. 1, 2, 6
[47] Chaoyi Zhang, Kevin Lin, Zhengyuan Yang, Jianfeng Wang, Linjie Li, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. Mm-narrator: Narrating long-form videos with multimodal in-context learning. *CoRR*, abs/2311.17435, 2023. 2, 6

[48] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 6

[49] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona T. Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: open pre-trained transformer language models. *CoRR*, abs/2205.01068, 2022. 3