DECEMBERT: Learning from Noisy Instructional Videos via Dense Captions and Entropy Minimization

Zineng Tang **Jie Lei** **Mohit Bansal**
Department of Computer Science
University of North Carolina at Chapel Hill
{terran, jielei, mbansal}@cs.unc.edu

Abstract

Leveraging large-scale unlabeled web videos such as instructional videos for pre-training followed by task-specific finetuning has become the *de facto* approach for many video-and-language tasks. However, these instructional videos are very noisy, the accompanying ASR narrations are often incomplete, and can be irrelevant to or temporally misaligned with the visual content, limiting the performance of the models trained on such data. To address these issues, we propose an improved video-and-language pre-training method that first adds automatically-extracted dense region captions from the video frames as auxiliary text input, to provide informative visual cues for learning better video and language associations. Second, to alleviate the temporal misalignment issue, our method incorporates an entropy minimization-based constrained attention loss, to encourage the model to automatically focus on the correct caption from a pool of candidate ASR captions. Our overall approach is named DECEMBERT (Dense Captions and Entropy Minimization). Comprehensive experiments on three video-and-language tasks (text-to-video retrieval, video captioning, and video question answering) across five datasets demonstrate that our approach outperforms previous state-of-the-art methods. Ablation studies on pre-training and downstream tasks show that adding dense captions and constrained attention loss help improve the model performance. Lastly, we also provide attention visualization to show the effect of applying the proposed constrained attention loss.[1]

1 Introduction

Video and language are ubiquitous in the world we live. The ability to understand the interplay of video and language is thus essential for intelligent agents to operate in real-world scenario. Past success in video-and-language has mostly been driven by supervised learning, where models are learned on manually labeled data for a particular task (e.g., text-to-video retrieval). However, manually annotating video and language data is very expensive, hence limiting the scale of such datasets, and consequently also limiting the performance of models trained on the datasets. The self-supervised pre-training then finetuning paradigm offers an easy and generic solution to this dilemma, where models are first pre-trained on large-scale unlabeled data by performing various "proxy tasks", followed by finetuning the pre-trained model on downstream tasks where data is often limited.

Recent advances on language pre-training (Devlin et al., 2019; Liu et al., 2019) demonstrate the effectiveness of this approach, where transformer-based (Vaswani et al., 2017) models pre-trained on large-scale unlabeled text corpus has shown to perform remarkably well across a wide range of natural language tasks (Rajpurkar et al., 2016; Williams et al., 2017; Zellers et al., 2018; Wang et al., 2018). Following this momentum, multimodal pre-training (Tan and Bansal, 2019; Lu et al., 2019; Chen et al., 2020; Su et al., 2019; Cho et al., 2021; Sun et al., 2019; Li et al., 2020c; Zhu and Yang, 2020; Miech et al., 2020; Li et al., 2020b; Lei et al., 2021) on large-scale image-text corpus (Sharma et al., 2018; Chen et al., 2015; Krishna et al., 2017) and video-text corpus (Lei et al., 2018; Miech et al., 2019; Sun et al., 2019) have also shown to outperform existing approaches (Anderson et al., 2018; Yu et al., 2018a; Lei et al., 2020a,b) on vision and language tasks (Antol et al., 2015; Xu et al., 2016; Yu et al., 2018a; Suhr et al., 2019; Zhou et al., 2017; Lei et al., 2020b). The most commonly used "proxy tasks" for multimodal pre-training are masked language modeling (Devlin et al., 2019) (MLM) and cross-modal matching (Tan and Bansal, 2019; Lu et al., 2019; Zhu and Yang, 2020) (e.g., video-text matching), where MLM aims to learn a better language model in the presence of the extra

[1]Code and models: https://github.com/zinengtang/DeCEMBERT

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2415–2426
June 6–11, 2021.

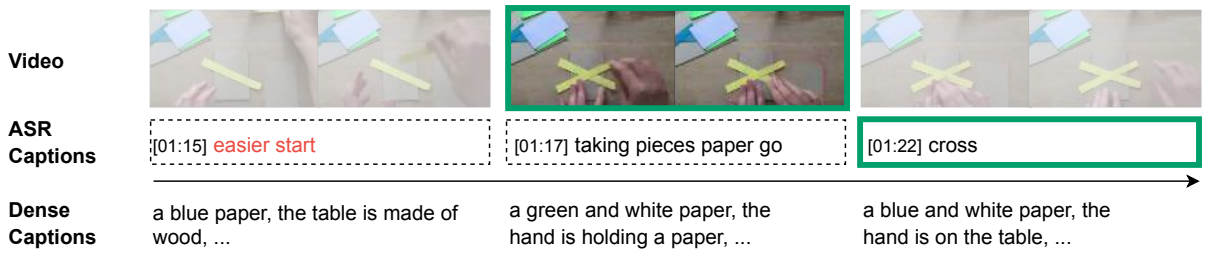

Figure 1: An instructional video example from HowTo100M (Miech et al., 2019). We show three clips and their corresponding ASR captions and dense captions. We use *green box* to indicate correct matched ASR caption for the middle clip. We highlight semantically misaligned ASR caption in *pink*. As can be seen from this example, the ASR captions are often incomplete and unpunctuated, and are semantically or temporally misaligned with their corresponding clips. In contrast, dense captions typically capture key objects, attributes and actions in the clips.

vision modality, and the matching objective encourages better association and alignment between relevant image-text or video-text pairs.

Existing video-text pre-training models (Sun et al., 2019; Miech et al., 2020; Zhu and Yang, 2020) are typically trained on large-scale instructional video datasets such as HowTo100M (Miech et al., 2019). The dataset contains 1.2 million videos with 136 million clips that are automatically harvested from YouTube. Each clip is paired with text transcribed from the video narrations via an automatic speech recognition (ASR) system. While the models trained on HowTo100M have shown promising results, they suffer from a few inherent drawbacks from the dataset: (i) Semantic misalignment: the narration words are sometimes irrelevant to the visual content (e.g., credits or other non-visual words, see Figure 1 text highlighted in *pink*), and *vice versa*, i.e., some important visual objects and actions are not described by words. (ii) Temporal misalignment: the videos and the captions are far from perfectly aligned, i.e., people might talk about something before or after they actually demonstrate it. For example, Figure 1 shows the caption "cross" is spoken after the action happened. Miech et al. (2019) reported that around 50% of the clip-caption pairs in HowTo100M suffers from these two misalignments, both of which cause difficulties in optimizing the video-text matching objective. (iii) Furthermore, the ASR captions are generally noisy, incomplete, and unpunctuated (Tilk and Alumäe, 2015) (e.g., in Figure 1, "taking pieces paper go"), which limits the language modeling ability of the systems that trained on such text.

To address the aforementioned issues, we propose to add Dense Captions (Johnson et al., 2016; Yang et al., 2017) as a complementary text input to the ASR captions. Beyond serving as an extra language input for better language modeling, dense captions also describes important object, attribute, and action details regarding several salient regions in the video frames, providing useful signals for video-text matching. In addition to its use in the pre-training stage, these dense captions also provide helpful clues for downstream tasks such as video question answering.

In parallel, to alleviate the temporal misalignment issue, we propose a constrained attention loss that encourages the model to automatically focus on the relevant ASR caption from a pool of continuous caption candidates. Instead of using only a single paired ASR caption for each clip, we also use the captions from its neighboring clips. We expect one of neighboring captions semantically aligns with the clip. To encourage the alignment between the clip and its relevant caption, we employ a "constrained attention loss" that encourages the attention mass from video features to the captions to be distributed mostly in one of the caption, by minimizing the entropy of attention scores.

We evaluate our DECEMBERT (Dense Captions and Entropy Minimization) model on a wide range of video-and-language tasks, including video question answering (Xu et al., 2017), text-to-video retrieval (Xu et al., 2016; Zhou et al., 2017), and video captioning (Xu et al., 2016; Zhou et al., 2017), where our approach outperforms previous state-of-the-art methods. To better understand the underlying factors that contribute to this success, we present comprehensive analyses concerning each of the added components.

To summarize, our contribution is three-fold: (i) We propose incorporating automatically extracted dense captions as an extra text input for video-text pre-training. (ii) We propose an entropy minimization-based constrained attention loss to

encourage the model to dynamically select the best matched captions from a pool of neighboring captions, to alleviate the inherent misalignment between the ASR captions and the videos. (*iii*) Extensive experiments on three video-and-language tasks (text-to-video retrieval, video captioning, and video question answering) across five datasets demonstrate the effectiveness of our approach. Furthermore, we also provide comprehensive ablation study and visualization to quantitatively and qualitatively examine the effect of using dense captions and the proposed constrained attention loss.

2 Related Work

Since the birth of BERT (Devlin et al., 2019), transformer (Vaswani et al., 2017) language pre-training models (Liu et al., 2019; Yang et al., 2019; Lan et al., 2020; Dong et al., 2019; Song et al., 2019; Raffel et al., 2020; Clark et al., 2020) which perform unsupervised pre-training followed by downstream task specific finetuning has became the *de facto* approach for various natural language understanding tasks (Rajpurkar et al., 2016; Williams et al., 2017; Zellers et al., 2018; Wang et al., 2018). Followed by this success, image-and-language pre-training models (Tan and Bansal, 2019; Lu et al., 2019; Chen et al., 2020; Zhou et al., 2020; Li et al., 2020a) and video-and-language pre-training models (Sun et al., 2019; Miech et al., 2019; Zhu and Yang, 2020; Miech et al., 2020; Li et al., 2020b; Luo et al., 2020; Huang et al., 2020; Stroud et al., 2020) have also shown promising results on many vision and language tasks (Antol et al., 2015; Xu et al., 2016; Zhou et al., 2017).

For video-and-language pre-training in particular, most existing work (Sun et al., 2019; Miech et al., 2019; Zhu and Yang, 2020; Miech et al., 2020; Li et al., 2020b; Luo et al., 2020) are trained on large-scale unlabeled instructional videos, such as HowTo100M (Miech et al., 2019) videos. However, as the ASR captions associated with these videos are noisy, i.e., they are often temporally or semantically misaligned with the video content. Miech et al. (2020) propose Multiple Instance Learning Noise Contrastive Learning (MIL-NCE) to address the temporal misalignment issue, but semantic misalignment still remains. Moreover, MIL-NCE requires computing a separate similarity score from the target clip to each of the ASR caption candidates, it does not suitable for the prevailing single-stream transformer pre-training architecture due to linearly increased computation cost.

Inspired by recent work (Kim and Bansal, 2019; Kim et al., 2020) that uses dense captions (Johnson et al., 2016; Yang et al., 2017) to improve image and video QA models, we propose to add dense captions an as auxiliary text input that provide aligned visual cues to ease the difficulties of learning a video-text matching objective from often temporally and semantically misaligned ASR captions. In addition, we also propose a constrained attention loss, which employs an entropy minimization-based regularization (Tanaka et al., 2018; Yi and Wu, 2019) to the model to encourage higher attention scores from the video to the correct matched caption among a pool of ASR caption candidates.

3 Method

In this section, we describe the details of DECEMBERT, including its architecture, pre-training objectives, dense caption inputs, and the constrained attention loss. Figure 2 shows an overview of DECEMBERT.

Input Representations. Input text (e.g., ASR captions) are tokenized and represented as a sequence of WordPiece (Wu et al., 2016) tokens. We use a trainable word embedding layer to encode the tokens into feature representations. We use appearance and motion features to represent videos. For appearance, we use a resnet152 (He et al., 2016) model pre-trained on ImageNet (Deng et al., 2009) to extract 2D video features at 1FPS. Similarly, for motion, we use a 3D ResNeXt (Xie et al., 2017; Hara et al., 2018; Kataoka et al., 2020) to extract 3D video features at 1FPS. The temporally aligned appearance and motion features are L2-normalized and concatenated together at feature dimension. We then apply a two-layer MLP to map the it to the same dimension as the word embeddings. Next, we add learned positional embedding and token type embedding (Devlin et al., 2019) to the video and text representations to encode the position and token type information. The video and text representations are then concatenated as a single sequence as inputs to a 12-layer transformer encoder for pre-training and downstream task finetuning.

Dense Captions. The original captions from ASR systems might not well describe a video with rich content or can even be irrelevant to the video as discussed in Section 1. Moreover, as ASR cap-

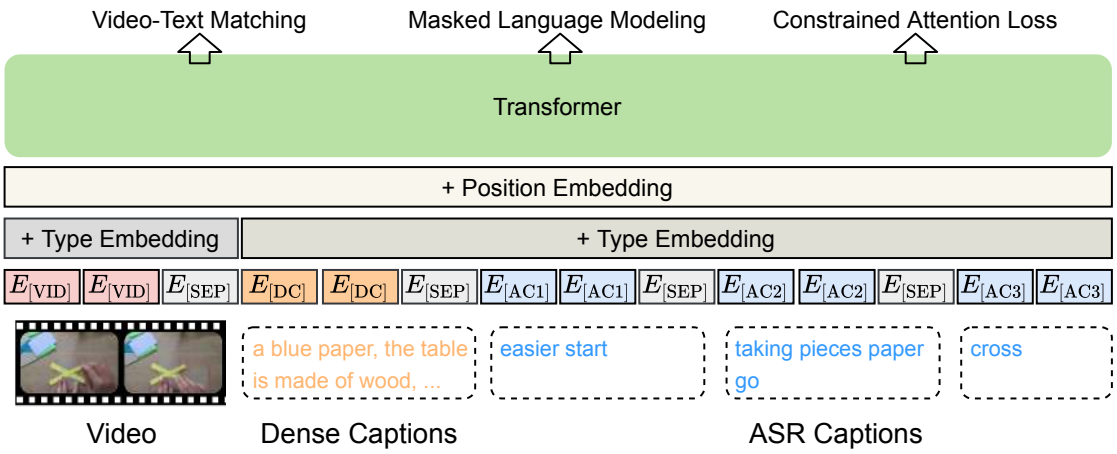

Figure 2: Overview of DECEMBERT architecture. It takes video representations, dense captions and ASR captions as input to its transformer layers, and learn model parameters via video-text matching, masked language modeling. It is also regularized by a constrained attention loss for learning better alignment between the video clips and the ASR captions.

tions are often incomplete and unpunctuated, they might also be sub-optimal for language modeling. Therefore, we use dense captions (Johnson et al., 2016) automatically extracted from an off-the-shelf image dense captioning model (Yang et al., 2017) as additional language input for the model. This dense captioning model is pre-trained on Visual Genome (Krishna et al., 2017) regional captions. To obtain video-level captions, we extract dense captions from frames sampled at every two seconds. There are on average 4.4 dense captions per frame, we sample two of them from each frame at each training step to avoid redundant information and reduce memory and computation cost. Note that the other dense captions might still be sampled in another training step. The sampled dense captions are then concatenated together as video-level captions for training.

These extracted dense captions provide rich and comprehensive information regarding the salient objects, attributes, and actions (see examples in Figure 1 and Figure 2), which helps to optimize a video-text matching objective during pre-training and provide essential visual clues for many downstream tasks such as video question answering. Meanwhile, because the dense captions are text input with diverse semantics, it complements the typically short and incomplete ASR captions as additional resources for better language modeling. We observe in our ablation study that adding dense captions improves both MLM accuracy and video-text matching accuracy, demonstrating the effectiveness of using them as extra inputs.

Pre-Training Objectives. During pre-training, we use masked language modeling (Devlin et al., 2019) (MLM) and cross-modality matching (Tan and Bansal, 2019; Lu et al., 2019; Miech et al., 2019; Zhu and Yang, 2020) (also referred as video-text matching in our context) as our objectives to learn model parameters. For masked language modeling, the goal is to learn better language models conditioned on bidirectional text context and the video. We set a probability of 0.20[2] to replace an input language token with `[MASK]`. When dense captions are used as extra text input, we also perform masked language modeling on them with the same masking probability as the ASR captions.

For video-text matching, with a probability of 0.50, we replace the original ASR captions with randomly sampled captions from other videos or clips as a negative. Of the sampled negative ASR captions, 50% of them are from different videos, while another 50% are from the same video but different clips. Text from the same video clip is likely to have the same theme or similar context, and thus can serve as hard samples to improve the model's ability to do fine-grained matching. We do not designate a `[CLS]` token before the start of input caption, instead we take the mean pooling of the output sequence hidden states to perform binary classification for video-text matching. Empirically, we found this approach works better than using a

[2]Because ASR captions are typically very short and grammatically less rigorous, we use a higher masking probability of 0.20 instead of the commonly used 0.15 as in BERT (Devlin et al., 2019).

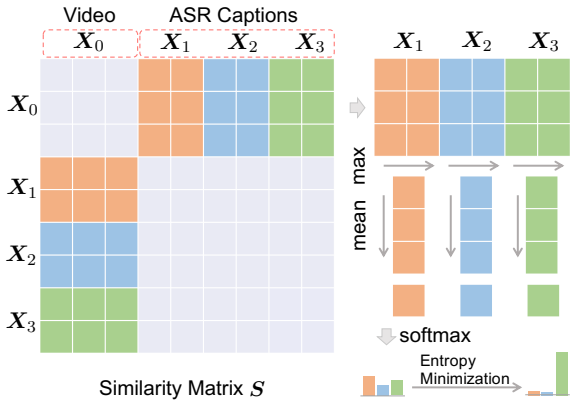

Figure 3: Illustration of applying the proposed entropy minimization-based constrained attention loss. This loss is added to every attention head in the transformer layers. It forces the model to give high attention scores only to one of the candidate ASR captions, i.e., to peak at only one caption rather than being flat because the one-hot distribution has the smallest entropy.

designated `[CLS]` token.

Constrained Attention Loss. The ASR captions are often temporally misaligned with their corresponding clips, simply pre-train a model over these misaligned clip-text pairs may lead to sub-optimal performance. To alleviate this issue, we propose a constrained attention loss that encourages the model to automatically select the best matched ASR caption from a pool of continuous caption candidates. This is achieved by minimizing the entropy of the attentions from the video to the ASR captions. Formally, we denote an input video V as $[c_1, c_2, ..., c_N]$, its corresponding ASR captions are denoted as $[s_1, s_2, ..., s_N]$, where c_i is the i-th clip of V and s_i is the ASR caption of c_i, N is the total number of clips in the video. For a clip c_i, instead of only inputting its associated caption s_i, we also include captions from its two neighboring clips,[3] i.e., s_{i-1} and s_{i+1}. In most cases, the correct matched caption for the clip is from these three captions. We denote $\boldsymbol{X}=[\boldsymbol{X}_{c_i}; \boldsymbol{X}_{s_{i-1}}; \boldsymbol{X}_{s_i}; \boldsymbol{X}_{s_{i+1}}] \in \mathbb{R}^{l\times d}$ as the generalized input sequence to each transformer layer (dense captions are ignored for simplicity), where $\boldsymbol{X}_{c_i}, \boldsymbol{X}_{s_{i-1}}, \boldsymbol{X}_{s_i}, \boldsymbol{X}_{s_{i+1}}$ are the embedding matrices correspond to the input clip and three captions. We further simplify the notations as $\boldsymbol{X}=[\boldsymbol{X}_0; \boldsymbol{X}_1; \boldsymbol{X}_2; \boldsymbol{X}_3]$. A single head self-attention operation in the transformer encoder layers can then be expressed as:

$$\boldsymbol{A} = \text{softmax}(\frac{\boldsymbol{X}\boldsymbol{X}^T}{\sqrt{d}}, \text{dim=1})\boldsymbol{X}, \quad (1)$$

where $\text{softmax}(\cdot, \text{dim=1})$ denotes applying softmax at the second dimension of the input matrix. $\boldsymbol{A}$ is the attention output matrix. When multiple attention heads are used, the formulation is similar. We use $\boldsymbol{S}$ to denote the similarity matrix computed by $\boldsymbol{X}\boldsymbol{X}^T$, it can be expressed using block matrices:

$$\boldsymbol{S}_{q,r} = \boldsymbol{X}_q \boldsymbol{X}_r^T, \quad q, r \in \{0, 1, 2, 3\}. \quad (2)$$

Our goal is to encourage the model to focus on the correct matched caption for an input clip, i.e., the attention mass from the video clip to the correct matched caption should be higher than the others. To achieve this, we first define the maximum response between the video hidden states $\boldsymbol{X}_0$ to the ASR captions hidden states $\boldsymbol{X}_j$ as:

$$\boldsymbol{z}_j = \max(\boldsymbol{S}_{0,j}, \text{dim=1}), j \in \{1, 2, 3\}. \quad (3)$$

For a single example, we define its constrained attention loss as:

$$u_j = \frac{\exp(\bar{\boldsymbol{z}_j})}{\sum_{k=1}^{3} \exp(\bar{\boldsymbol{z}_k})}, \quad (4)$$

$$\mathcal{L}_e = -\sum_{j=1}^{3} u_j \log(u_j). \quad (5)$$

This loss formulation is based on entropy minimization (Tanaka et al., 2018; Yi and Wu, 2019), it forces the model to assign high attention scores only to one of the ASR captions, i.e., to peak at only one caption rather than being flat because the one-hot distribution has the smallest entropy. Figrue 3 shows an overview of applying the constrained attention loss. During pre-training, we add this loss to each of the attention heads across all layers, we add these losses along with the MLM loss and video-text matching loss for joint optimization. Meanwhile, as the similarity matrix $\boldsymbol{S}$ is a symmetric matrix, the entropy minimization objective also encourages the correct matched ASR caption to have higher similarity to the video, while forcing the mismatched captions to put more attention on the other ASR captions rather than the video.

4 Experiments

In this section, we compare our model with state-of-the-art methods on three video-and-language downstream tasks (e.g., video captioning, text-to-video

[3] While our approach works for arbitrary number of neighbors, we use two neighbors to illustrate the idea for simplicity. In fact, we found that, of 100 randomly sampled videos, using two neighbors already covers 95% of the videos with at least one positive matched ASR caption.

retrieval, and video question answering) across five datasets. We then present a comprehensive ablation study, where we show that each of our proposed components help improve the pre-training task performance and downstream task performance. Lastly, we also provide an attention visualization example to demonstrate the effect of applying our proposed constrained attention loss.

4.1 Datasets and Tasks

Pre-training. We use HowTo100M (Miech et al., 2019) for pre-training. It contains 1.22 million YouTube instructional videos that cover 23.6K instruction tasks (e.g., *making peanut butter, pruning a tree*). Each video is associated with an English narration automatically transcribed by an Automatic Speech Recognition (ASR) system. On average, each video has 110 clip-caption pairs, with an average duration of 4 seconds per clip and 4 words per caption. We reserve 10K videos for validation, and use the rest of the videos for pre-training.

Video Captioning. We evaluate video captioning on MSRVTT (Xu et al., 2016) and YouCook2 (Zhou et al., 2017) datasets. The task is to generate a text description (a single sentence or a paragraph of multiple sentences) for a given video. (i) **MSRVTT** contains 10K YouTube videos with 20 descriptions per video. The videos in MSRVTT are typically 10-30 seconds long, with an average length of 14.8 seconds. Its contains 6.5K videos in the train set, 497 videos in the val set, and 3K videos in the test set. (ii) **YouCook2** is a cooking video dataset harvested from YouTube. It contains 2K videos from 89 recipes with a total length of 176 hours. Each video is annotated with temporal timestamps that indicate event segments (clips), a textual description is provided for each segment. In total, there are 14K video segments.

Text-to-Video Retrieval. We evaluate text-to-video retrieval on MSRVTT and YouCook2 datasets, where the goal is to retrieve a relevant video from a gallery of videos given a text query. (i) **MSRVTT** is the same dataset as the captioning task. We follow previous work (Yu et al., 2018b; Miech et al., 2019) to use the 7k train+val videos for training and report results on the 1K test set sampled by Yu et al. (2018b). (ii) **YouCook2** is the same dataset as the captioning task. We evaluate our model on the clip retrieval task as in previous work (Miech et al., 2019; Zhu and Yang, 2020).

Method	B@4	M	R	C
SibNet (Liu et al., 2020b)	40.9	27.5	60.2	47.5
OA-BTG (Zhang and Peng, 2019)	41.4	28.2	-	46.9
GRU-EVE (Aafaq et al., 2019)	38.3	28.4	60.7	48.1
MGSA (Chen and Jiang, 2019)	42.4	27.6	-	47.5
POS+CG (Wang et al., 2019)	42.0	28.2	61.6	48.7
POS+VCT (Hou et al., 2019)	42.3	29.7	62.8	49.1
ORG-TRL (Zhang et al., 2020)	43.6	28.8	62.1	50.9
DECEMBERT	**45.2**	**29.7**	**64.7**	**52.3**

Table 1: Video captioning results on MSRVTT *test* set. We report BLEU@4 (B@4), METEOR (M), Rouge-L (R), CIDEr-D (C).

Video Question Answering. We evaluate video question answering (QA) performance on the **MSRVTT-QA** (Xu et al., 2017) dataset. It contains 243K open-ended questions constructed based on the videos and captions in MSRVTT.

4.2 Implementation Details

We use the BERT-base (Devlin et al., 2019) architecture as our transformer encoder, with hidden size 768 and 12 transformer layers. The entire model contains 115M parameters. The maximum length of video features is set to 100 for both pre-training and downstream tasks. We use Adam optimizer (Kingma and Ba, 2014) to optimize the model, with an learning rate of 1e-4, β_1=0.9, β_2=0.98, L2 weight decay of 0.01. For pre-training, we train the model for 20 epochs until convergence. Dense captions in different frames are potentially repeated if the contiguous frames have similar objects. This is expected as some videos have smooth shooting that stays at one angle for an extended time. We filter those dense captions to avoid redundancy. For downstream tasks, we finetune from the same pre-trained weights and use the same training and optimization settings as pre-training. We conduct all the experiments using NVIDIA GeForce GTX 1080Ti GPUs and Intel(R) Xeon(R) CPU E5-2630 v4. During pre-training, the model's inference speed under this infrastructure with one GPU is 5 samples per second.

4.3 Comparison to State-of-the-Art

We present our results on three downstream tasks across five datasets, and compare the results against the state-of-the-art methods. All the downstream results are obtained by fine-tuning the same pre-trained model that is pre-trained with dense captions and constrained attention loss.

Method	B@4	M	R	C
MTrans	7.62	15.65	32.18	32.26
MART	8.00	15.90	35.74	35.74
MART+COOT	9.44	18.17	34.32	46.06
MTrans+COOT+MIL-NCE PT	11.05	19.79	37.51	55.57
MART+COOT+MIL-NCE PT	11.30	19.85	37.94	57.24
DECEMBERT	**11.92**	**20.01**	**40.22**	**58.02**

Table 2: Video captioning results on YouCook2 *val* set. Model references: MTrans (Zhou et al., 2018), MART (Lei et al., 2020a), COOT (Ging et al., 2020), and MIL-NCE (Miech et al., 2020). *PT* indicates models with pre-training on HowTo100M.

Method	R@1	R@5	R@10	MdR
HERO (Li et al., 2020b) w/ ASR, PT	20.5	47.6	60.9	-
JSFusion (Yu et al., 2018b)	10.2	31.2	43.2	13.0
HowTo (Miech et al., 2019)	12.1	35.0	48.0	12.0
HowTo (Miech et al., 2019) PT	14.9	40.2	52.8	**9.0**
Univilm (Luo et al., 2020) PT	15.4	39.5	52.3	**9.0**
ActBERT (Zhu and Yang, 2020) PT	16.3	42.8	56.9	10.0
HERO (Li et al., 2020b) PT	16.8	43.4	57.7	-
DECEMBERT	**17.5**	**44.3**	**58.6**	**9.0**

Table 3: Text-to-video retrieval results on MSRVTT 1k test set (Yu et al., 2018b). PT indicates models with pre-training on HowTo100M (or on HowTo100M+TV shows (Lei et al., 2018; Liu et al., 2020a) for HERO). We gray out models that used extra ASR features for a fair comparison.

Method	R@1	R@5	R@10	MdR
HGLMM	4.6	14.3	21.6	75.0
HowTo	4.2	13.7	21.5	65.0
HowTo PT	8.2	24.5	35.3	24.0
COOT	5.9	16.7	24.8	49.7
COOT+MIL-NCE PT	16.7	40.2	52.3	**9.0**
DECEMBERT	**17.0**	**43.8**	**59.8**	**9.0**

Table 4: Text-to-video retrieval results on YouCook2 *val* set. PT indicates models with pre-training on HowTo100M. Model references: HGLMM (Klein et al., 2015), HowTo (Miech et al., 2019), COOT (Ging et al., 2020), MIL-NCE (Miech et al., 2020)

Video Captioning. We follow Vaswani et al. (2017) to train auto-regressive captioning models, by only allowing the text tokens to attend to tokens that precede them at training. During inference time, we use beam search with beam size 5 to generate captions. For MSR-VTT, we evaluate captioning performance at sentence level. For YouCook2, we follow previous work (Lei et al., 2020a; Ging et al., 2020) to evaluate performance at paragraph-level, where single segment captions are concatenated as a paragraph for evaluation. We use standard metrics BLEU@4 (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), Rouge-L (Lin, 2004), and CIDEr-D (Vedantam et al., 2015) to report performance. Table 1 shows the comparison on MSRVTT, our DECEMBERT model achieves significant performance gain over previous state-of-the-art. Notably, DECEMBERT outperforms ORG-TRL (Zhang et al., 2020) by 1.6% BLEU@4, 2.6% Rouge-L, and 1.4% CIDEr-D, even though ORG-TRL uses a set of strong visual features (appearance, motion, and object) together with a sophisticated graph encoder network and external language model supervision. Table 2 shows the results on YouCook2 captioning task. Overall, DECEMBERT outperforms previous methods across all metrics. Compared to the strong baseline method MART+COOT+NIL-NCE (Lei et al., 2020a; Ging et al., 2020; Miech et al., 2020) PT, that uses HowTo100M videos for pre-training followed by a designated hierarchical modeling training, our approach still shows better performance with a reasonable margin. This shows the effectiveness of our pre-training strategy.

Text-to-video Retrieval. We train text-to-video retrieval models similar to the way we perform video-text matching, where we sample a negative caption 50% of the time. We use average recall at K (R@K) and median rank (MdR) to report performance on the retrieval tasks. We show MSRVTT text-to-video retrieval in Table 3. Overall, our approach achieves the best performance. Compared to the pre-trained models HowTo (Miech et al., 2019), ActBERT (Zhu and Yang, 2020), and HERO (Li et al., 2020b), DECEMBERT achieves strong performance with a reasonable margin. It outperforms HERO by 0.7% R1, note that HERO is pre-trained with extra TV show videos (Lei et al., 2018; Liu et al., 2020a) in addition to the HowTo100M videos that we use. Moreover, DECEMBERT is also quite competitive compared to the HERO w/ ASR model that uses additional ASR features during finetuning. For YouCook2 text-to-video retrieval results shown in Table 4, our approach also show better performance compared to the pre-trained models HowTo and COOT+MIL-NCE. Notably, it outperforms previous state-of-the-art COOT+MIL-NCE by 7.5% R@10.

Video Question Answering. We use a two-layer MLP followed by a softmax layer for open-ended question answering, where we optimize the probability of choosing the correct answer from a large pool of candidate answers. We report accuracy to

Method	Accuracy
ST-VQA (Jang et al., 2017)	30.9
Co-Memory (Gao et al., 2018)	32.0
AMU (Xu et al., 2017)	32.5
Heterogeneous Memory (Fan et al., 2019)	33.0
HCRN (Le et al., 2020)	35.6
DECEMBERT	**37.4**

Table 5: Video question answering results on MSRVTT-QA *test* set.

measure the QA performance. We show MSRVTT-QA results in Table 5 where our approach outperform all the baseline methods by a large margin. Compared to HCRN (Le et al., 2020) which employs a complicated hierarchical reasoning module, our approach achieves 1.8% performance gain, achieving a new state-of-the-art for the task.

4.4 Analysis

Ablation Study. We present ablation study on our pre-training strategies, on both the pre-training tasks and the MSRVTT captioning downstream task. We report ablation results on our 10K hold-out HowTo100M videos for pre-training tasks, i.e., masked language modeling (MLM) accuracy and video-text matching accuracy. Because we use MLM for both dense captions and the original ASR captions, we report their accuracy separately. The results are shown in Table 6. To understand how the pre-training strategies affect the downstream performance, we also perform downstream finetuning from pre-trained models using these different pre-training strategies. The results are shown in Table 7. Compared to the basic model that uses only a single paired ASR caption with each clip for training, we observe the the variant that takes three ASR captions achieves significantly higher accuracy in MLM and video-text matching. Adding dense captions and constrained attention loss further improve the performance. Overall, the same trend also holds true for the downstream performance on MSRVTT captioning and QA tasks. The best captioning and QA models are finetuned from the model pre-trained using both the dense captions and the constrained attention loss. Compared to the basic model with only MLM and video-text matching, our best models achieve a significant performance gain: e.g., 3.3% BLEU@4, 3.1% CIDEr-D for captioning, and 2.3% Accuracy for QA.

Qualitative Results During pre-training, we apply our proposed constrained attention loss to every

Pre-training Method	MLM Acc		Matching Acc
	ASR	Dense	
MLM & Matching	21.95	-	61.63
+ Neighboring ASR Cap.	48.42	-	78.90
+ Dense Captions	49.78	84.06	80.02
+ Constrained Loss	50.66	84.46	80.32

Table 6: Ablation results on HowTo100M (Miech et al., 2019) hold-out *val* set. Each row adds an extra component to the row above it.

Pre-training Method	Captioning				QA
	B@4	M	R	C	Acc
MLM & Matching	41.3	27.6	60.3	48.1	35.7
+ Neighboring ASR Cap.	41.6	28.0	60.5	47.8	35.8
+ Dense Captions	43.5	29.6	63.9	49.4	36.7
+ Constrained Loss	44.6	29.9	64.0	51.2	37.0

Table 7: Ablation results on MSRVTT captioning and MSRVTT QA tasks, both on *val* set. Each row adds an extra component to the row above it.

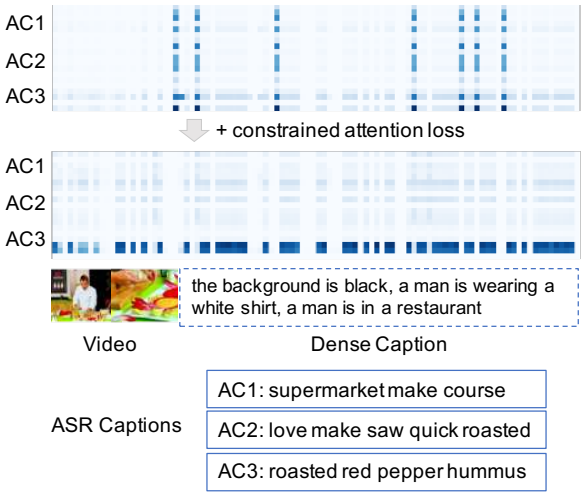

Figure 4: Attention visualization for models with and without constrained attention loss. After adding constrained attention, the attention mass concentrated to the ASR caption (e.g., *AC3*) that best matches the video content and the dense captions. These attention maps are taken from an attention head of the 10-th layer of the transformer model.

attention heads across all layers. In Figure 4, we compare the attention maps from models with or without the proposed constrained attention loss during pre-training. As we found the attention weight distributions (not absolute values) on different layers look similar to each other, we randomly chose the 10-th layer to showcase the effect of adding constrained attention loss. We observe that after adding constrained attention loss as a regularization, the attention mass concentrated to the best-

matched ASR caption rather than distributed to all the captions.

5 Conclusion

In this work, we propose DECEMBERT as an improved pre-training method for learning from noisy, unlabeled instructional videos. Specifically, we propose adding automatically-extracted frame-level dense captions as an auxiliary text input for learning better video and language associations. We also propose a constrained attention loss that forces the model to automatically focus on the best-matched caption from a pool of misalignment caption candidates via entropy minimization. Comprehensive experiments on three popular video and language tasks (i.e., text-to-video retrieval, video captioning, and video question answering) across five datasets demonstrate the effectiveness of DECEMBERT compared to existing approaches. We also provide detailed ablation study and visualization to quantitatively and qualitatively examine the impact of our added components.

Acknowledgements

We thank the reviewers for their helpful feedback. This research is supported by DARPA MCS Grant #N66001-19-2-4031, DARPA KAIROS Grant #FA8750-19-2-1004, ARO-YIP Award #W911NF-18-1-0336, and Google Focused Research Award. The views contained in this article are those of the authors and not of the funding agency.

References

Nayyer Aafaq, Naveed Akhtar, Wei Liu, Syed Zulqarnain Gilani, and Ajmal Mian. 2019. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *CVPR*.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *ICCV*.

Shaoxiang Chen and Yu-Gang Jiang. 2019. Motion guided spatial attention for video captioning. In *AAAI*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Learning universal image-text representations. In *ECCV*.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. *arXiv preprint arXiv:2102.02779*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *NeurIPS*.

Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. 2019. Heterogeneous memory enhanced multimodal attention model for video question answering. In *CVPR*.

Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. 2018. Motion-appearance co-memory networks for video question answering. In *CVPR*.

Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. 2020. Coot: Cooperative hierarchical transformer for video-text representation learning. In *NeurIPS*.

Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

Jingyi Hou, Xinxiao Wu, Wentian Zhao, Jiebo Luo, and Yunde Jia. 2019. Joint syntax representation learning and visual cue translation for video captioning. In *ICCV*.

Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. 2020. Multimodal pretraining for dense video captioning. *arXiv preprint arXiv:2011.11760*.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. 2017. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*.

Hirokatsu Kataoka, Tenga Wakamiya, Kensho Hara, and Yutaka Satoh. 2020. Would mega-scale datasets further enhance spatiotemporal 3d cnns? *arXiv preprint arXiv:2004.04968*.

Hyounghun Kim and Mohit Bansal. 2019. Improving visual question answering by referring to generated paragraph captions. In *ACL*.

Hyounghun Kim, Zineng Tang, and Mohit Bansal. 2020. Dense-caption matching and frame-selection gating for temporal localization in videoqa. In *ACL*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. 2015. Associating neural word embeddings with deep image representations using fisher vectors. In *CVPR*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *ICLR*.

Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. 2020. Hierarchical conditional relation networks for video question answering. In *CVPR*.

Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*.

Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara L Berg, and Mohit Bansal. 2020a. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *ACL*.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. 2018. Tvqa: Localized, compositional video question answering. In *EMNLP*.

Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. 2020b. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *ECCV*.

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. 2020a. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*, pages 11336–11344.

Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020b. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020c. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*.

Jingzhou Liu, Wenhu Chen, Yu Cheng, Zhe Gan, Licheng Yu, Yiming Yang, and Jingjing Liu. 2020a. Violin: A large-scale dataset for video-and-language inference. In *CVPR*.

Sheng Liu, Zhou Ren, and Junsong Yuan. 2020b. Sibnet: Sibling convolutional encoder for video captioning. *TPAMI*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.

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

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *ICML*.

Jonathan C Stroud, David A Ross, Chen Sun, Jia Deng, Rahul Sukthankar, and Cordelia Schmid. 2020. Learning video representations from textual web supervision. *arXiv preprint arXiv:2007.14937*.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. A corpus for reasoning about natural language grounded in photographs. In *ACL*.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *ICCV*.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*.

Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2018. Joint optimization framework for learning with noisy labels. In *CVPR*.

Ottokar Tilk and Tanel Alumäe. 2015. Lstm for punctuation restoration in speech transcripts. In *Interspeech*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. 2019. Controllable video captioning with pos sequence guidance based on gated fusion network. In *ICCV*.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *CVPR*.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. 2017. Video question answering via gradually refined attention over appearance and motion. In *ACM MM*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*.

Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. 2017. Dense captioning with joint inference and visual context. In *CVPR*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

Kun Yi and Jianxin Wu. 2019. Probabilistic end-to-end noise correction for learning with noisy labels. In *CVPR*.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018a. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*.

Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018b. A joint sequence fusion model for video question answering and retrieval. In *ECCV*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *EMNLP*.

Junchao Zhang and Yuxin Peng. 2019. Object-aware aggregation with bidirectional temporal graph for video captioning. In *CVPR*.

Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. 2020. Object relational graph with teacher-recommended learning for video captioning. In *CVPR*.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *AAAI*.

Luowei Zhou, Chenliang Xu, and Jason J Corso. 2017. Towards automatic learning of procedures from web instructional videos. In *AAAI*.

Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. 2018. End-to-end dense video captioning with masked transformer. In *CVPR*.

Linchao Zhu and Yi Yang. 2020. Actbert: Learning global-local video-text representations. In *CVPR*.