VLM: Task-agnostic Video-Language Model Pre-training for Video Understanding

Hu Xu[1], **Gargi Ghosh**[1], **Po-Yao Huang**[12], **Prahal Arora**[1], **Masoumeh Aminzadeh**[1]
Christoph Feichtenhofer[1], **Florian Metze**[1] and **Luke Zettlemoyer**[1]
[1]Facebook AI
[2]Carnegie Mellon University
{huxu,gghosh,berniehuang,prarora,masoumeha,
feichtenhofer,fmetze,lsz}@fb.com

Abstract

We present a simplified, task-agnostic multi-modal pre-training approach that can accept either video or text input, or both for a variety of end tasks. Existing pre-training are task-specific by adopting either a single cross-modal encoder that requires both modalities, limiting their use for retrieval-style end tasks or more complex multitask learning with two unimodal encoders, limiting early cross-modal fusion. We instead introduce new pretraining masking schemes that better mix across modalities (e.g. by forcing masks for text to predict the closest video embeddings) while also maintaining separability (e.g. unimodal predictions are sometimes required, without using all the input). Experimental results show strong performance across a wider range of tasks than any previous methods, often outperforming task-specific pre-training[1].

1 Introduction

We study the challenge of achieving *task-agnostic* pre-training for multimodal video understanding, building on recent unimodal approaches such as pretrained language models for text (Peters et al., 2018; Devlin et al., 2019). Although certain language models are near task-agnostic (Devlin et al., 2019; Lewis et al., 2020) on NLP tasks, being task-agnostic on multi-modal tasks are more challenging due to cross-modal tasks such as text-video retrieval. Existing video-and-language pre-trainings are task-specific, which adopt either (1) a cross-modal single encoder (Sun et al., 2019b,a; Zhu and Yang, 2020) favoring tasks that require cross-modal reasoning (e.g. video captioning), or (2) multiple unimodal encoders/decoders (Miech et al., 2019, 2020; Li et al., 2020b; Luo et al., 2020; Korbar et al., 2020) combining specific tasks that require separately embedding each modality (e.g. video retrieval). We instead show that it is possible to pre-train a task-agnostic model called video-language model (VLM) that can accept text, video, or both as input.

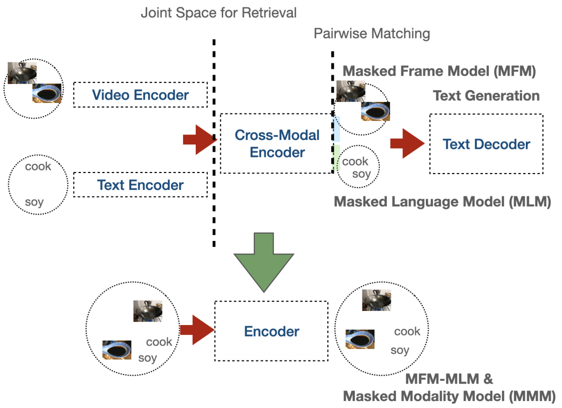


Figure 1: Existing models (upper figure) adopt complex architectures and multiple task-specific training to merge two streams of data to cover a wide range of downstream tasks (such as retrieval or text generation). Our video-language model (VLM) (lower figure) uses a single BERT encoder for task-agnostic pre-training (e.g. only masking tokens, no matching or alignment for specific end tasks) in a joint feature space, while still covering a wide range of tasks (see Figure 3).

As shown in Figure 1, this task-agnostic single encoder approach has several advantages: (1) it reduces the complexity of pre-training with multiple losses and models (e.g. Luo et al. (2020)), and (2) it holds less assumption on being close to end tasks as in retrieval-based pre-training Miech et al. (2020) and is as general as classic LMs, and (3) it encourages feature sharing among modalities when present, without sacrificing separability, and (4) it is more parameter efficient (see Section 5, we achieve strong performance with $\text{BERT}_{\text{BASE}}$ sized models). Table 1 summarizes the design choices of recent models.

Our encoder is a transformer block that combines the existing masked frame model and masked

[1]Code will be released under fairseq.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 4227–4239
August 1–6, 2021. ©2021 Association for Computational Linguistics

language model (MFM-MLM) (Sun et al., 2019a; Li et al., 2020b; Luo et al., 2020) with two new methods to improve the learning of multi-modal fusion. First, we introduce a masking scheme called masked modality model (MMM) that randomly masks a whole modality for a portion of training examples (the rest of the examples goes for traditional MFM-MLM), thereby forcing the encoder to use the tokens from the other modality to produce tokens for the masked modality. We then introduce a single masked token loss to replace two (2) losses on video and text separately for MFM-MLM. Masked token loss uses the embeddings of both video and text tokens to learn joint hidden states for the encoder.

We also show it is possible to fine-tune a single encoder for a wide range of tasks by using task-specific attention masks. Experiments demonstrate that it performs well on a wider range of tasks than previous models, including outperforming task-specific pre-training baselines with unimodal encoders of similar hyper-parameters by more than 2% on retrieval tasks and 1% on video captioning. Note that these results are also achieved with a much smaller model than previous approaches, further demonstrating the improved fusion and sharing across modalities.

In summary, the main contributions of this paper are as follows: (1) we propose to pre-train a task-agnostic encoder for video understanding; (2) we introduce masked modality model (MMM) and masked token loss for cross-modal fusion during pre-training without sacrificing separability; (3) experimental results show that the proposed simple baseline achieves competitive performance with significantly fewer parameters.

2 Related Work

Numerous multimodal task-specific pre-training models are proposed for downstream visual-linguistic tasks. In video and text pre-training, existing research adopts different design choices regarding proxy tasks and neural architectures for end tasks (Luo et al., 2020).

On one hand, VideoBERT (Sun et al., 2019b), Unicoder-VL (Li et al., 2020a), VL-BERT (Su et al., 2020), UNITER (Chen et al., 2020), VLP (Zhou et al., 2018), ActBERT (Zhu and Yang, 2020) adopt a *shared* encoder approach, where the vision and text sequences are concatenated and input to a single Transformer(Vaswani et al., 2017) encoder. Although this approach is simple, it limits the types of downstream tasks to those that input both modalities simultaneously. For example, (Sun et al., 2019b) may not be able to perform joint retrieval tasks and added another decoder for video captioning during fine-tuning. (Zhu and Yang, 2020) uses `[CLS]` token for pairwise metric-learning based retrieval (which is an easier problem but requires a quadratic number of examples and is 50 times slower as reported in (Luo et al., 2020)).

Meanwhile, many existing approaches adopt or add task-specific pre-training to accommodate retrieval and video captioning tasks (e.g. *two-stream* encoders (video and text separately) and text decoders). For example, (Miech et al., 2019, 2020; Rouditchenko et al., 2020; Ging et al., 2020; Gabeur et al., 2020; Alayrac et al., 2020; Patrick et al., 2021; Huang et al., 2021) adopts a retrieval task for pre-training. CBT (Sun et al., 2019a), HERO (Li et al., 2020b), VideoAsMT (Korbar et al., 2020) and UniVL (Luo et al., 2020) adopt multi-task learning (MTL) to learn retrieval tasks on video and text encoders. HERO (Li et al., 2020b) and UniVL (Luo et al., 2020) adopts another cross-encoder to further learn the fusion of different modality. UniVL (Luo et al., 2020) and VideoAsMT (Korbar et al., 2020) add another text decoder for video captioning. Compared with the single-stream input in the shared encoder approach, two-stream encoders typically come with a complex architecture and proxy tasks to cover more end tasks. To the best of our knowledge, none of the existing works target task-agnostic pre-training.

2.1 Image-Text Pre-training

ViLBERT (Lu et al., 2019), LXMERT (Tan and Bansal, 2019) adopt two transformers for image and text encoding separately. VisualBERT (Li et al., 2019), Unicoder-VL (Li et al., 2020a), VL-BERT (Su et al., 2020), UNITER (Chen et al., 2020), Unified VLP (Zhou et al., 2020) use one shared BERT model. These models employ MLM and pairwise image-text matching as pretraining tasks which are effective for downstream multimodal tasks. Our fine-tuning for video captioning is inspired by Unified VLP (Zhou et al., 2020) that adopts attention masks and language model heads of BERT for image-captioning.

2.2 Video-Text Pre-training

VideoBERT (Sun et al., 2019b) and CBT (Sun et al., 2019a) are the first works to explore the capability of pre-training for video-text. Although VideoBERT and CBT pre-train the model on multimodal data, the downstream tasks mainly take video representation for further prediction. ActBERT (Zhu and Yang, 2020) is a weakly-supervised pre-training method. It leverages global action information to catalyze mutual interactions between linguistic texts and local regional objects and introduces a transformer block to encode global actions, local regional objects, and linguistic descriptions. HERO (Li et al., 2020b) encodes multimodal inputs in a hierarchical fashion. Besides, two new pre-training tasks, video-subtitle matching and frame order modeling, are designed to improve representation learning. VideoAsMT (Korbar et al., 2020) and UniVL (Luo et al., 2020) further adopt a BART-style(Lewis et al., 2020) text generation task for downstream tasks such as video captioning and UniVL adopts a EnhancedV training stage to mask all text tokens for better learning of generation.

3 Pre-training

As a reminder, our goal is to train a *task-agnostic* model for various tasks in video-text understanding. This section introduces task-agnostic proxies for pre-training. We first describe two masking schemes as a baseline: masked frame model (MFM) for video frames and masked language model (MLM) for text tokens (Sun et al., 2019a; Li et al., 2020b; Luo et al., 2020). Then we introduce masked modality model (MMM) that encourage to learn the representations of one modality from the other. Lastly, we introduce masked token loss that unifies losses on masked video and text tokens as a single loss function.

3.1 Vector Quantization and BERT

Assume we have a clip (v, t) sampled from a video, where v and t corresponds to video modality and text modality, respectively. Since videos are signals in continuous space, we first extract token embeddings from raw videos. We decode v into frames and then feed them into a (frozen) video encoder $\text{Encoder}_{\text{video}}(\cdot)$ and a trainable MLP layer to obtain *video tokens*:

$$\boldsymbol{x}_v = \text{MLP}(\text{Encoder}_{\text{video}}(\boldsymbol{f}_v)), \qquad (1)$$

where we use a bolded symbol to indicate a sequence and $\boldsymbol{f}_v$ is a sequence of continuous frames from a video. We use S3D (Xie et al., 2018; Miech et al., 2020), which is pre-trained via self-supervised learning on the Howto100M dataset. The MLP layer allows the hidden size of video tokens to be the same as BERT's hidden sizes d: $x_v \in \mathbb{R}^d$. Similarly, vectors for text tokens $\boldsymbol{x}_t$ are obtained via embedding lookup as in BERT.

To simplify multi-modal pre-training, we adopt a single BERT transformer with minimum changes. We first concatenate video tokens $\boldsymbol{x}_v$ and text tokens $\boldsymbol{x}_t$ via the `[SEP]` token so video and text belongs to one corresponding segment of BERT:

$$\boldsymbol{x} = \texttt{[CLS]} \circ \boldsymbol{x}_v \circ \texttt{[SEP]} \circ \boldsymbol{x}_t \circ \texttt{[SEP]}. \qquad (2)$$

We further mask $\boldsymbol{x}$ as $\boldsymbol{x}_{\text{masked}}$ (detailed in the next subsection) and feed the whole sequence into BERT:

$$\boldsymbol{h} = \text{BERT}(\boldsymbol{x}_{\text{masked}}), \qquad (3)$$

where $\boldsymbol{h}$ indicates the hidden states of the last layer of BERT. To encourage learning video/text hidden states in a shared space for the masked token loss (introduced in Section 3.3), we use a *shared* head to predict video/text token embeddings via a linear projection layer:

$$e = \boldsymbol{W}h + b, \qquad (4)$$

where $e \in \mathbb{R}^d$ and $\boldsymbol{W}$ and b are the weights from the prediction heads of BERT. In this way, our model learns a joint embedding space for both video and text tokens from inputs to outputs of BERT. This allows for pre-training a single encoder directly from any existing LMs and the only layer that requires initialization is the MLP layer.

3.2 MFM-MLM

Inspired by (Sun et al., 2019a; Li et al., 2020b; Luo et al., 2020), we adopt masked frame model (MFM) for videos and masked language model (MLM) for text as a baseline. Note that unlike LMs that typically come with a fixed vocabulary with a special `[MASK]` token, video tokens are innumerable in the continuous space and we mask a video token by setting a video token with all zeros and ask the encoder to recover the video token. via noisy contrastive estimation (NCE):

$$\mathcal{L}_{\text{MFM}} = -\mathbb{E}_{s \sim V} \log \text{NCE}(x_s | \boldsymbol{x}_{\text{masked}}; V'), \qquad (5)$$

where V is all indexes of video tokens and

$$\text{NCE}(x_v | \boldsymbol{x}_{\text{masked}}; V') = \frac{\exp(x_v^T e_v)}{\exp(x_v^T e_v) + \sum_{j \in V'} \exp(x_j^T e_v)}, \qquad (6)$$

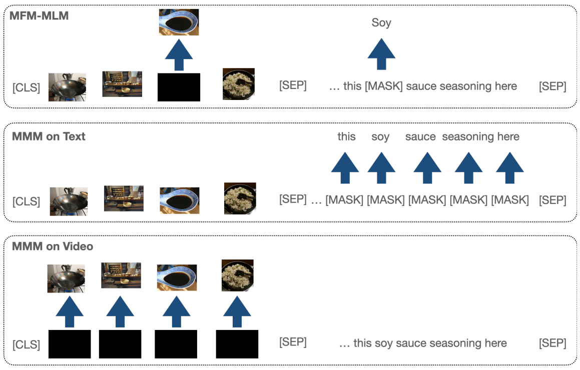

MFM-MLM
Soy
[CLS]
[SEP]
... this [MASK] sauce seasoning here
[SEP]
MMM on Text
this soy sauce seasoning here
[CLS]
[SEP] ... [MASK] [MASK] [MASK] [MASK] [MASK]
[SEP]
MMM on Video
[CLS]
[SEP]
... this soy sauce seasoning here
[SEP]

Figure 2: Task-agnostic pre-training (e.g. w/o task on retrieval-style alignment): MFM-MLM: 50% of training examples are masked as masked frame model (MFM) and masked language model (MLM); the rest 50% examples are masked as masked modality model (MMM) (25% on text as in the second row and 25% on video as in the third row).

where V' indicates all non-masked video tokens within the same batch. The final loss is the sum of both MFM and MLM:

$$\mathcal{L}_{\text{MFM-MLM}} = \mathcal{L}_{\text{MFM}} + \mathcal{L}_{\text{MLM}}, \qquad (7)$$

where $\mathcal{L}_{\text{MLM}}$ is the same as BERT and we omit its details for brevity. We experiment this classic baseline in Section 5.

3.3 MMM and Masked Token Loss

Masked Modality Model We introduce masked modality modal (MMM) that masking either all video or all text tokens out for a given example of video-text clip. This masking scheme complements MFM-MLM (e.g. in our experiments 50% of training examples are masked as MMM and the rest 50% are masked as MFM-MLM). This encourages the encoder to use tokens from one modality to recover the tokens for the other modality. This resolves the issue that an encoder may use nearby tokens from their modality for prediction just because tokens from a single modality are closer As in the lower two (2) sub-figures in Figure 2, we either mask the whole modality of video or text so this modality can be "generated" from the other modality. Our experiments indicate that this is critical for pre-training a single encoder for retrieval tasks.

Masked Token Loss We further introduce masked token loss that unifies loss functions for MFM and MLM. This loss encourages learning a joint token embedding space for video and text and both types of tokens contribute to the prediction of a masked (video or text) token. This also improves the number of contrasted negative embeddings in two separate losses for MFM and MLM.

We define masked token loss $\mathcal{L}_{\text{VLM}}$ as the following:

$$-\mathbb{E}_{s\sim V\cup D}\log \mathrm{NCE}(x_s|\boldsymbol{x}_{\text{masked}}; V'\cup D_{\setminus s}), \qquad (8)$$

where D is the word embeddings over the vocabulary of BERT and $D_{\setminus s}$ excludes token s (if s is a text token). Further, $\mathrm{NCE}(x_s|\boldsymbol{x}_{\text{masked}}; V'\cup D_{\setminus s})$ is defined as:

$$\frac{\exp(x_s^T e_s)}{\exp(x_s^T e_s) + \sum_{j\in V'\cup D_{\setminus s}} \exp(x_j^T e_s)}. \qquad (9)$$

Note that $j \in V'\cup D_{\setminus s}$ can be either a video or text token and one predicted token e_s must be closer to the ground-truth token embedding (either a video token or word embedding) and be away from other embeddings of video/text tokens. We perform an ablation study in Section 5 to show that $\mathcal{L}_{\text{VLM}}$ works better than $\mathcal{L}_{\text{MFM-MLM}}$.

4 Fine-tuning

In this section, we describe how to use different types of attention masks to fine-tune VLM for a variety of tasks, as shown in Figure 3.

4.1 Text-Video Retrieval

One major challenge of pre-training on a single encoder is how to adapt such a model to joint space retrieval without using unimodal encoders for task-specific pre-training on contrastive loss (as in Howto100M (Miech et al., 2019, 2020)). The main reason is that many existing models encode text and video tokens together via self-attention, and one cannot obtain hidden states for text/video alone.

To resolve this, we propose to apply an isolated attention mask with two squared masks that are diagonally placed, as shown in the lower sub-figure of the first box in Figure 3.[2] These two squares disable video and text tokens to attend and see each other, while still allow video and text tokens to use the same self-attention layers for learning representations in the same feature space. Further, note that the first and second `[SEP]` tokens of BERT will

[2] One can further reduce $O(m+n)^2$ complexity to $O(m^2+n^2)$ (m and n are lengths for video and text, respectively) by feeding video/text separately to BERT but we adopt squared masks for simplicity.

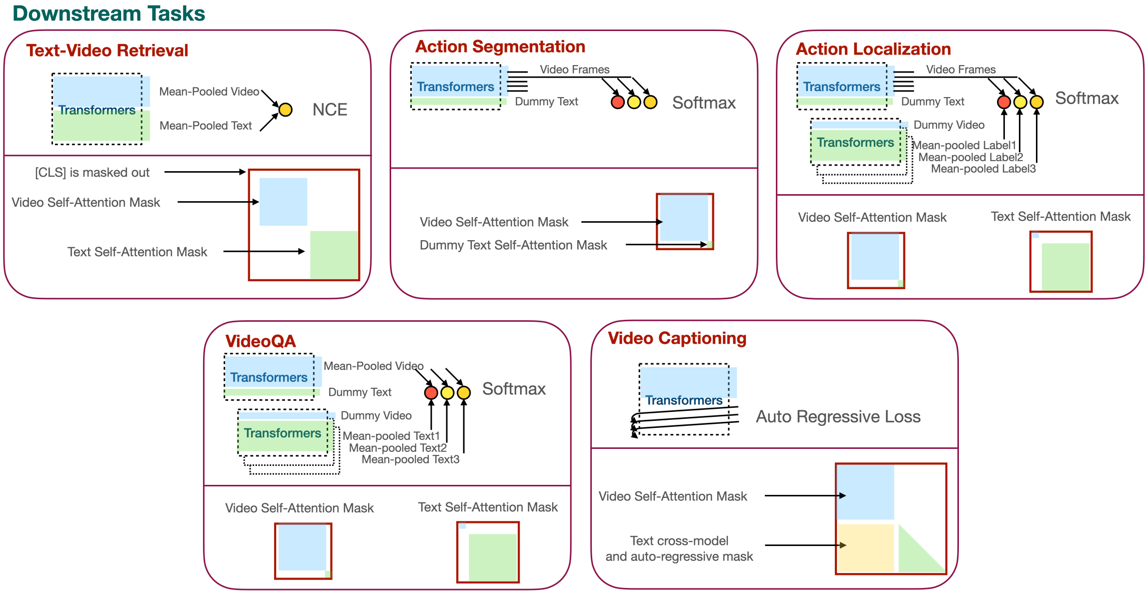


Figure 3: Fine-tuning of downstream tasks: we adopt different types of attention masks for BERT to accommodate downstream tasks that require different modalities: in each box, the upper sub-figure indicates a forward computation; the lower sub-figure indicates squared self-attention mask, where tokens from each row have a weighted sum of columns that are not in white colors.

be used by video and text, respectively, aiming to learn sequence-level features(Clark et al., 2019). The [CLS] is disabled as no need to learn features across video and text. After forwarding, all hidden states of video and text tokens are average pooled, respectively. Then we use a contrastive loss on text-video similarity to discriminate a ground-truth video clip from other video clips in the same batch for a given text clip. During the evaluation, to ensure video and text are isolated (to avoid leaking ground-truth of a similar pair), we split text and video and forward them separately. We report an ablation study in Section 5 showing that the MMM introduced in the previous section is crucial to ensure that the pre-trained hidden states (for video or text) are a good initialization for retrieval tasks.

4.2 Action Segmentation

Action segmentation is to assign each frame of a video with one of the pre-defined labels. This is similar to the named entity recognition (NER) task in NLP but on video frames. We feed in VLM with the whole video, a dummy text token, and an isolated attention mask. Then we add a classification head (with the number of pre-defined labels) on top of the hidden states for each video token in the last layer of VLM.

4.3 Action Step Localization

In action step localization, each video belongs to a task with multiple steps, where each step is described as a short text. Then each frame of a video needs to be aligned with a step in text form. The challenge for applying BERT to action step localization is similar to text-video retrieval: video frames need to be aligned with textual steps in joint space and it is almost impossible for pairwise video/text matching because the number of frame/text pairs is large.

Similar to the text-video retrieval model, we also apply isolated attention masks to video and text. The major difference is that we pass video and text separately to BERT. This is because the video can be several minutes long (more than 100 tokens) but the number of text labels for each video is fixed (e.g. under 10). To keep the format of BERT being consistent for multi-modal inputs, we add a dummy text token for video forwarding and a dummy video token for text, respectively. For a given frame(video token), we compute the distribution of that frame over textual steps via dot products and the softmax function.

4.4 Multiple-choice VideoQA

Multiple-choice VideoQA (Yu et al., 2018) aligns each video with one out of several candidate answers in the text. The major difference between action step localization and multiple-choice VideoQA is that the video hidden state is not on frame-level but sequence-level. We apply isolated attention masks to BERT and forward video and text answers (with dummy tokens), respectively. Then the answer with the maximum similarity with the video is reported. During fine-tuning, we apply contrastive loss on video-text similarity to rank answers.

4.5 Video Captioning

Another big challenge of using a single encoder is how to apply generative tasks (such as video captioning) without pre-training an explicit decoder. We observe that a transformer decoder (Vaswani et al., 2017) has the following major differences from an encoder: (1) an auto-regressive loss that does not allow a text token to see future tokens; (2) a prediction head to generate texts. To resolve (1), one can easily fine-tune the text segment of VLM as auto-regressive loss by passing in shifted tokens and a lower-triangle attention mask to the text segment, as shown in Figure 3. To resolve (2), inspired by (Rothe et al., 2020; Zhou et al., 2020) that uses BERT as a decoder, one can re-use language model heads as prediction heads for generation. Note that this setting has less architecture design than a standard transformer decoder (e.g. no explicit self-attention on text or cross-attention on video). The implicit text decoder inside BERT shares self-attention with the video encoder so to save the total number of parameters.

5 Experiment

5.1 Dataset

5.1.1 Pre-training

We adopt the Howto100M dataset (Miech et al., 2019) for pre-training, which contains instructional videos originally from YouTube via searching keywords from wikihow (`www.wikihow.com`). After filtering the unavailable ones, we get 1.1M videos. We split 4000 videos as the validation set and the rest for pre-training. On average, the duration of each video is about 6.5 minutes with 110 clip-text pairs. After removing repeated texts within overlapped clips from ASR, we get about 7.7+ GB texts of captions, with 2.4 tokens per second on average.

5.1.2 Fine-tuning

MSR-VTT (Xu et al., 2016) is a popular dataset for *text-video retrieval* and *VideoQA*. It has open domain video clips, and each training clip has 20 captioning sentences labeled by humans. There are 200K clip-text pairs from 10K videos in 20 categories, including sports, music, etc. Following JSFusion(Yu et al., 2018; Miech et al., 2019), we randomly sampled 1,000 clip-text pairs as test data. We further use the QA test data (Yu et al., 2018) as the dataset for multiple-choice VideoQA.

Youcook2 (Zhou et al., 2017) contains 2,000 cooking videos on 89 recipes with 14K video clips from YouTube. The overall duration is 176 hours (5.26 minutes on average). Each video clip is annotated with one captioning sentence. Follow the split setting in(Miech et al., 2019), we evaluate both text-based video retrieval and multimodal video captioning tasks. We filter the data and make sure there is no overlap between pre-training and evaluation data. After filtering out unavailable ones, we have 9,473 training clip-text pairs from 1222 videos and 3,305 test clip-text pairs from 430 videos.

COIN (Tang et al., 2019) are leveraged to evaluate *action segmentation*. It has 11,827 videos (476 hours) and each video is labeled with 3.91 step segments on average and 46,354 segments in total. There are 778 step labels, plus one background (Outside) label. Since one video can last for several minutes that are much longer than the maximum length of the video segment of VLM. We apply a sliding window with step size 16 and window size 32. During inference, we average the logits for overlapped frames from multiple windows.

CrossTask (Zhukov et al., 2019) is a dataset for action localization that contains 83 different tasks and 4.7k videos. Each task has a set of steps with text descriptions annotated on temporal frames of the video. We use the testing data split via the official code[3], which contains annotated 1690 videos. The rest of the 540 annotated videos are used for weakly supervised training.

5.2 Hyper-parameters

We extract video tokens from video frames using the S3D encoder pre-trained from (Miech et al., 2020). The fps is 30 and we extract one (1) video token per second with the dimension of 512. We apply an MLP to transform such 512 dimensions to the hidden size (768) of $\text{BERT}_{\text{BASE}}$.

Following (Luo et al., 2020), we adopt $\text{BERT}_{\text{BASE}}$ (uncased) as our base model and tuned directly from BERT's weights, so all hyper-parameters are the same as the original BERT. The maximum length of BERT is set as 96, where 32

[3] `https://github.com/DmZhukov/CrossTask`

Model	Paradigm	#params.	#loss	#unimodal/cross en/decoder	Joint Retrieval	Generation
MMT(Gabeur et al., 2020)	task-specific alignment	127.3M	1	2/0/0	yes	no
ActBERT(Zhu and Yang, 2020)	weakly supervised/MTL	n/a (3 typed attentions)	4	0/1(modal-typed attn.)/0	no(pair)	extra decoder
VideoAsMT(Korbar et al., 2020)	weakly supervised/MTL	286M(base)/801M(large)	1	1/1/1	no (gen.)	yes
HERO(Li et al., 2020b)	SSL(w/ sup. video feat.)/MTL	159M	5	1(query)/2/0	no(pair)	extra decoder
UniVL(Luo et al., 2020)	SSL/MTL	260M	5	2/1/1	yes	yes
VLM	SSL/Task-agnostic	**110M**	**1**	0/**1**/0(shared w/ encoder)	yes	yes

Table 1: Comparison of pre-trained models on learning paradigms (SSL means self-supervised learning; MTL means multi-task learning), number of parameters (# params.), number of losses (#loss), number of unimodal/cross-modal encoders/decoders, and whether to support retrieval in joint space(joint retrieval) and text generation. Types and numbers are estimated based on released code or papers: exceptions are in parenthesis (e.g. pair means pairwise matching using `[CLS]`). VLM is extremely simple with fewer parameters and limitations.

Methods	R@1	R@5	R@10	Median R
Random	0.1	0.5	1.0	500
Task-specific Alignment Pre-training				
MMT (Gabeur et al., 2020)	25.8	57.2	69.3	4
Pairwise Matching				
ActBERT(Zhu and Yang, 2020)	8.6	23.4	33.1	36
VideoAsMT(Korbar et al., 2020)	14.7	-	52.8	-
Multi-task Pre-training				
HERO (Li et al., 2020b)	16.80	43.40	57.70	-
UniVL (FT-Joint) (Luo et al., 2020)	20.6	49.1	62.9	6
VLM	28.10	55.50	67.40	4

Table 2: Results of text-video retrieval on MSR-VTT dataset.

Methods	R@1	R@5	R@10	Median R
Random	0.03	0.15	0.3	1675
Task-specific Alignment Pre-training				
Coot(Ging et al., 2020)	16.7	40.2	52.3	9
Pairwise Matching				
ActBERT(Zhu and Yang, 2020)	9.6	26.7	38.0	19
VideoAsMT(Korbar et al., 2020)	11.6	-	43.9	-
Multi-task Pre-training				
UniVL (FT-Joint)(Luo et al., 2020)	22.2	52.2	66.2	5
VLM	27.05	56.88	69.38	4

Table 3: Results of text-based video retrieval on Youcook2 dataset.

tokens are for videos and the rest tokens are for text and special tokens. Remind that texts are 2.4 tokens per second and video tokens are 1 token per second. We form a text clip with a random length in-between 8 and 64 text tokens and collect the corresponding video clip to form a training example. We randomly sample 32 video/text clip pairs from each video and use 8 videos to form a batch of size 256. Each training example has 50% chance for MMM (25% for whole video masking and 25% for whole text masking) and 50% chance on MFM-MLM (with 15% probability of video and text token masking).

We pre-train VLM on 8 NVIDIA Tesla V100 GPUs (each with 32 GB memory) for 15 epochs using fp16 for one (1) day. Following (Liu et al., 2019), we choose Adam (Kingma and Ba, 2014) optimizer with initial learning rate of 5e-5 (with betas as (0.9, 0.98)), 1000 steps of warm-up and a polynomial decay learning rate scheduler. Gradients are clipped with 2.0. All fine-tuning tasks use the same hyper-parameters as pre-training except the number of warm-up steps is 122.

5.3 Model Comparison

We first investigate the design choices of VLM compared to other transformer-based multimodal pre-training baselines. As shown in Table 1, we collect training paradigms, model sizes, etc. of these models (estimated based on their source codes or papers). VLM is significantly smaller than other models since it is just a BERT$_{\text{BASE}}$ (uncased), while it is still fully self-supervised, task-agnostic (e.g. no training on retrieval or auto-regressive style tasks) and supports joint retrieval and text generation.

5.4 Quantitative Analysis

We investigate the performance of VLM on fine-tuning tasks with very basic setups (e.g. no augmented features, large LMs, optimized losses for particular tasks). Note that it could be hard for fair comparisons between task-agnostic and task-specific approaches. We list other baselines by type and our goal is a simple baseline for task-agnostic pre-training as better initialization of strongly performed fine-tuning models.

Text-video Retrieval We use MSR-VTT and Youcook2 to evaluate the performance on text-video retrieval. The results are shown in Table 2 and 3, respectively. VLM achieves good performance on these two datasets, indicating that the MMM and isolated self-attention mask can be used together for joint retrieval. Ablation study shows that using an isolated self-attention mask alone does not yield good performance, indicating MMM is very important to learn features for alignment. Note that our pre-training is task-agnostic but still outperforms baselines with retrieval style pre-training.

Action Segmentation We report the results of action segmentation on COIN dataset in Table 4.

Method	Frame Accuracy
NN-Viterbi (Richard et al., 2018)	21.17
VGG (Simonyan and Zisserman, 2014)	25.79
TCFPN-ISBA (Ding and Xu, 2018)	34.30
CBT (Sun et al., 2019a)	53.90
MIL-NCE (Miech et al., 2020)	61.00
ActBERT (Zhu and Yang, 2020)	56.95
VLM	68.39

Table 4: Action segmentation on COIN dataset.

Methods	Average Recall
Joint Alignment	
Alayrac (Alayrac et al., 2016)	13.3
Zhukov (Zhukov et al., 2019)	22.4
Supervised (Zhukov et al., 2019)	31.6
HowTo100M (Miech et al., 2019)	33.6
MIL-NCE (Miech et al., 2020)	40.5
UniVL (Luo et al., 2020)	42.0
Pairwise Matching	
ActBERT (Zhu and Yang, 2020)	41.4
VLM (task-agnostic, zero-shot)	28.5
VLM (supervised on 540 videos)	46.5

Table 5: Action step localization results on CrossTask.

Method	Accuracy
Joint Retrieval	
JSFusion(Yu et al., 2018)	83.4
Pairwise Matching	
ActBERT(Zhu and Yang, 2020)	85.7
VLM	91.64

Table 6: Video question answering (multiple-choices) evaluated on MSR-VTT.

Methods	B-3	B-4	M	R-L	CIDEr
Extra Decoder					
VideoBERT (Sun et al., 2019b)	6.80	4.04	11.01	27.50	0.49
CBT (Sun et al., 2019a)	-	5.12	12.97	30.44	0.64
ActBERT (?)	8.66	5.41	13.30	30.56	0.65
Coot(Ging et al., 2020)	17.62	11.09	19.34	37.63	-
w/ Pre-trained Decoder					
VideoAsMT (Korbar et al., 2020)	-	5.3	13.4	-	-
UniVL (Luo et al., 2020)	16.46	11.17	17.57	40.09	1.27
VLM	17.78	12.27	18.22	41.51	1.3869

Table 7: Video captioning results on Youcook2 dataset.

VLM outperforms other baselines indicating its good token-level video representations. Note that this task only tests the hidden states of the video indicating the unimodal encoding capability of VLM is not compromised.

Action Step Localization We setup two (2) evaluations for the CrossTask dataset. First, we evaluate the zero-shot transfer of VLM. Note that existing studies evaluate Crosstask with retrieval/alignment style pre-training, where the aligned hidden states are directly used for action step localization. Our task-agnostic pre-training derives an even harder problem: applying hidden states learned from proxy tasks on video frame/text alignment for action step localization without explicitly training on alignment. We simply use the hidden states from the last layer of VLM for video/text representation and directly compute the similarities between video frames and text descriptions. Surprisingly, the performance is better than some baselines and closer to one supervised method. This indicates masked token loss together with MMM can learn certain video-text alignments in joint space. Second, we use just 540 videos for weakly supervised training and we get a much better result.

Video Question Answering We use MSR-VTT QA to evaluate multiple-choice question answering. Recall that this task essentially tests video-text similarity. The performance of VLM is better than ActBERT, which leverages pairwise matching for each video/answer pair.

Video Captioning We lastly evaluate VLM on video captioning with autoregressive attention mask with other baselines that have an explicit text decoder. As shown in Table 7, our "compact" decoder using BERT's LM heads is surprisingly good at video captioning compared to other fine-tuning baselines with external decoders (e.g. Coot). This indicates that it is possible to remove an explicit decoder and sharing weights between video and text tokens.

5.4.1 Ablation Study

We use Youcook2 as the base task for the ablation study on text-retrieval and video captioning. We are interested in the following study: (1) percentage of examples for MMM (w/ MMM x%); (2) minimum length of text tokens, where the length of video will be determined by the start/end timestamps of text tokens; (3) performance of $\mathcal{L}_{\text{VLM}}$ (Equation 8). The results are shown in Table 8 and Table 9.

Effects of MMM Without MMM (w/ MMM 0%, or MFM-MLM 100%), the performance significantly dropped. This indicates that a naive adoption of traditional MFM-MLM masking may not learn joint video/text representations well, as indicated by both retrieval and captioning task. We suspect a masked token is more likely predicted from tokens of the same modality. We further try MMM with different probabilities (30% or 70%) and 50% is the best.

Minimum Length of Texts The length of a clip can be important for retrieval tasks (Miech et al., 2020). We ran VLM on longer (at least 16 text tokens) video/text pairs. The performance is slightly dropped, indicating pre-training on longer clips may not cover fine-tuning tasks with short clips.

Effects of Masked Token Loss We notice that using multi-task style loss $\mathcal{L}_{\text{MFM-MLM}}$ may reduce the performance. This indicates learning a masked token from both video/text tokens can help.

VLM	R@1	R@5	R@10	Median R
w/ MMM 50%	27.05	56.88	69.38	4.0
w/ MMM 0%	15.12	39.47	52.81	9.0
w/ MMM 30%	25.30	54.80	68.96	4.0
w/ MMM 70%	25.17	54.98	69.11	4.0
w/ min. 16 text tokens	25.84	54.43	68.29	5.0
w/ $\mathcal{L}_{\text{MFM-MLM}}$	26.93	55.92	69.86	4.0

Table 8: Ablation study of VLM for text-based video retrieval on Youcook2.

VLM	B-3	B-4	M	R-L	CIDEr
w/ MMM 50%	17.78	12.27	18.22	41.51	1.3869
w/ MMM 0%	15.47	10.54	16.49	38.83	1.2163
w/ MMM 30%	16.57	11.30	17.55	40.76	1.3215
w/ MMM 70%	16.94	11.68	17.67	41.24	1.3739
w/ min. 16 text tokens	17.25	12.00	17.67	40.62	1.3076
w/ $\mathcal{L}_{\text{MFM-MLM}}$	16.66	11.53	17.34	40.36	1.3224

Table 9: Ablation study of VLM for video captioning on Youcook2 dataset.

5.5 Qualitative Analysis

5.5.1 Error Analysis

Text-video retrieval. We use MSR-VTT as the dataset for error analysis on text-video retrieval, as shown in Table 10 of Appendix. We pair the query text with the text of the top-1 ranked video to show 100 errors in ranking since video tokens are harder to present. We observe the following types of errors in video understanding: (1) objects sometimes are hard to recognize such as dog or cat; (2) attributes of objects may be hard to match the text, e.g. gender, ages, etc. (3) subtle differences of actions; (4) specific videos for a general query or vice versa, e.g. people vs basketball player. We believe the last type may not be errors but hard for existing annotations or evaluations to separate.

Video Captioning. We further examine the generated text from video captioning. Note that our video captioning has no support from ASR or transcript so the video is the only source to generate text content and errors of video understanding can easily be reflected in the text. From Table 11 of Appendix, we notice that one major type of error is from objects of similar shapes and colors, e.g. onion rings vs shrimp.

5.5.2 Visualization

. We observe that video tokens take the majority of space while text tokens are rather clustered together. This is probably because videos from the physical world are more diverse and sparse than text from a fixed vocabulary.

We plot the self-attention of VLM layers within and in-between each modality, as in Figure 4 of Appendix. We observe the following patterns from all 144 attention heads:

- Unlike LMs, there are no recurrent (shifted) position-wise patterns for video tokens;
- Self-attentions in the 1st layer are more diverse than later layers. This suggests that existing video encoders might be too deep for transformers;
- Some attention heads show patterns of cross-modal mapping in-between video and text (e.g. sub-figure (a));
- Word-level cross-modal co-reference: video tokens with *pouring soy sauce* refers to the text token of "soy" (e.g. sub-figure (b));

6 Conclusions

We presented a task-agnostic pre-training with new masking schemes that enable the training of a single masked language model that can accept either video or text input, or both. We showed that this simple VLM model can be effectively tuned for a broad range of downstream tasks, such as text-video retrieval and video captioning via different types of attention masks. Experimental results show that the proposed methods maintain competitive performance while requiring a significantly smaller number of parameters than competing methods.

Acknowledgments

We thank Huaishao Luo (author of UniVL(Luo et al., 2020)), Mandela Patrick (author of Support-Set(Patrick et al., 2021)) and Luowei Zhou (author of Youcook(Zhou et al., 2017)) for supports of baseline setup.

References

Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. 2016. Unsupervised learning from narrated instruction videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4575–4583.

Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. 2020. Self-supervised multimodal versatile networks. *arXiv preprint arXiv:2006.16228*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120. Springer.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Ding and Chenliang Xu. 2018. Weakly-supervised action segmentation with iterative soft boundary assignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6508–6516.

Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal transformer for video retrieval. In *European Conference on Computer Vision (ECCV)*, volume 5. Springer.

Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. 2020. Coot: Cooperative hierarchical transformer for video-text representation learning. *arXiv preprint arXiv:2011.00597*.

Po-Yao Huang, Mandela Patrick, Junjie Hu, Graham Neubig, Florian Metze, and Alexander G. Hauptmann. 2021. Multilingual multimodal pretraining for zero-shot cross-lingual transfer of vision-language models. *CoRR*, abs/2103.08849.

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

Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. 2020. Video understanding as machine translation. *arXiv preprint arXiv:2006.07203*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065, Online. Association for Computational Linguistics.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. In *Arxiv*.

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 *Advances in Neural Information Processing Systems*, volume 32, pages 13–23. Curran Associates, Inc.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889.

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 *Proceedings of the IEEE international conference on computer vision*, pages 2630–2640.

Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F. Henriques, and Andrea Vedaldi. 2021. Support-set bottlenecks for video-text representation learning. In *International Conference on Learning Representations*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alexander Richard, Hilde Kuehne, Ahsan Iqbal, and Juergen Gall. 2018. Neuralnetwork-viterbi: A framework for weakly supervised video learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7386–7395.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Andrew Rouditchenko, Angie Boggust, David Harwath, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Rogerio Feris, Brian Kingsbury, Michael Picheny, Antonio Torralba, et al. 2020. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. Vl-bert: Pretraining of generic visual-linguistic representations. In *International Conference on Learning Representations*.

Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. 2019a. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 3(5).

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019b. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7464–7473.

Hao Tan and Mohit Bansal. 2019. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China. Association for Computational Linguistics.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. 2019. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1207–1216.

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 *Advances in neural information processing systems*, pages 5998–6008.

Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. 2018. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 305–321.

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

Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 471–487.

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*, pages 13041–13049.

Luowei Zhou, Chenliang Xu, and Jason J Corso. 2017. Towards automatic learning of procedures from web instructional videos. *arXiv preprint arXiv:1703.09788*.

Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. 2018. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748.

Linchao Zhu and Yi Yang. 2020. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8746–8755.

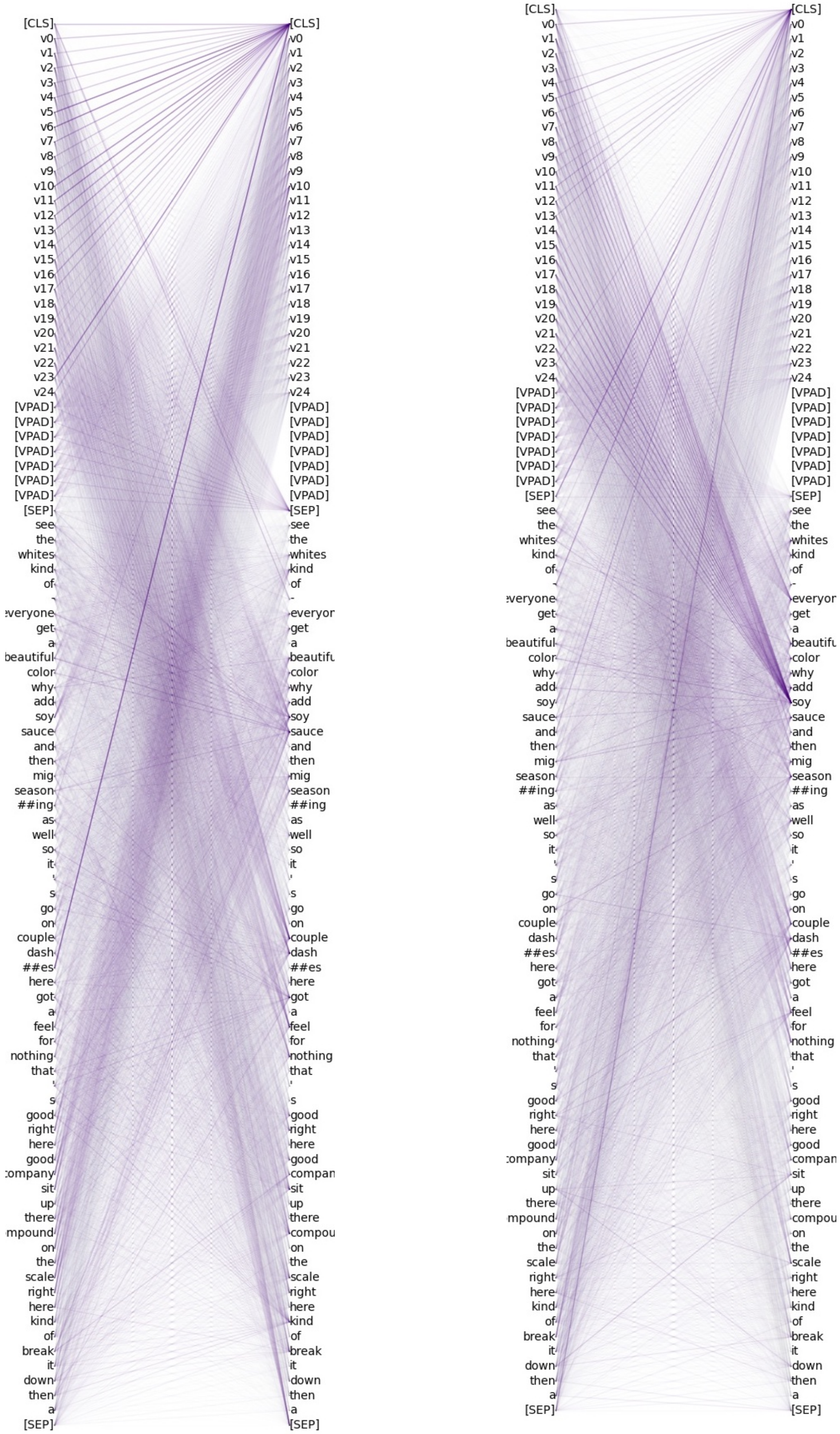
Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. 2019. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545.

Query	Text of Top-1 video
Objects (26%)	
cartoon show for kids	**pokemon video game** play
little pet shop **cat** getting a bath and washed with little brush	several **dogs** playing dead
Attributes of Objects (6%)	
a little **boy** singing in front of judges and crowd	a **woman** singing on the voice
a woman is mixing **food** in a mixing bowl	a man is stirring **something** in a pot
Action (6%)	
a person is **connecting something** to system	a man **looks at** the **battery** of a computer
a boy **plays** grand theft auto 5	a narrator **explains where to find** a rare vehicle in grand theft auto
a man is **giving a review** on a vehicle	a person is **discussing** a car
a **naked child** runs through a field	the **girl** shows the **boys** her medal in this cartoon
a man is singing and standing in the road	a man in sunglasses and a blue shirt beat boxes
Specific vs General (62%)	
some cartoon characters are moving around an area	a cartoon girl and animal jumping on body of male guy girl image still shown displaying on screen
baseball player hits ball	**people** are playing baseball
the man in the video is showing a brief viewing of how the movie is starting	scrolling the the menu of movieclips with different movie trailers
a **student** explains to his **teacher** about the sheep of another student	there is a **guy** talking to his **father**
a video about different sports	a woman talks about horse racing

Table 10: Error analysis for text-video retrieval of MSR-VTT on 100 errors: we group errors in four (4) categories: objects, attributes of objects, actions, and specific vs general. Specific videos for general queries (or vice versa) sometimes may not be errors but hard to evaluate.

Hypothesis	Reference
add the lamb to the **pan**	add the lamb to the **pot**
add the cilantro ~~cilantro~~ and lime juice to the pot	cut the cilantro and lime
add the **onions** to a pot of water	add **flour** to the pot and stir
dip the **onion rings** into the batter	dip the **shrimp** in the batter
add water to the bowl and mix	pour water into the **flour mixture** and mix
remove the **mussels** from the pot	once the **shrimps** are defrosted drain the water
add the sauce to the **pan** and stir	add the sauce to the **wok** and stir
add lemon juice to the pan and stir	add **rice vinegar** and lemon juice to the pan and stir
add the beef to the pan and stir	add the diced beef meat to it and roast it

Table 11: Error analysis for video captioning on Youcook2: VLM tends to make mistakes in recognizing objects of similar shapes and colors to generate the wrong text.



(a) Layer 1, Head 1

(b) Layer 1, Head 5

Figure 4: Self-attention for video HfIeQ9pzL5U from 4:03 to 4:28: darker color indicates higher weights; v0-v24 are video tokens of 25 seconds.