Bridge-Prompt: Towards Ordinal Action Understanding in Instructional Videos

Muheng Li[1], Lei Chen[1], Yueqi Duan[2], Zhilan Hu[3], Jianjiang Feng[1], Jie Zhou[1], Jiwen Lu[†,1]
[1]Department of Automation, Tsinghua University
[2]Department of Electronic Engineering, Tsinghua University
[3]Media Technology Institute, Huawei Technologies Co., Ltd.

Abstract

Action recognition models have shown a promising capability to classify human actions in short video clips. In a real scenario, multiple correlated human actions commonly occur in particular orders, forming semantically meaningful human activities. Conventional action recognition approaches focus on analyzing single actions. However, they fail to fully reason about the contextual relations between adjacent actions, which provide potential temporal logic for understanding long videos. In this paper, we propose a prompt-based framework, Bridge-Prompt (Br-Prompt), to model the semantics across adjacent actions, so that it simultaneously exploits both out-of-context and contextual information from a series of ordinal actions in instructional videos. More specifically, we reformulate the individual action labels as integrated text prompts for supervision, which bridge the gap between individual action semantics. The generated text prompts are paired with corresponding video clips, and together co-train the text encoder and the video encoder via a contrastive approach. The learned vision encoder has a stronger capability for ordinal-action-related downstream tasks, e.g. action segmentation and human activity recognition. We evaluate the performances of our approach on several video datasets: Georgia Tech Egocentric Activities (GTEA), 50Salads, and the Breakfast dataset. Br-Prompt achieves state-of-the-art on multiple benchmarks. Code is available at: https://github.com/ttlmh/Bridge-Prompt.

1. Introduction

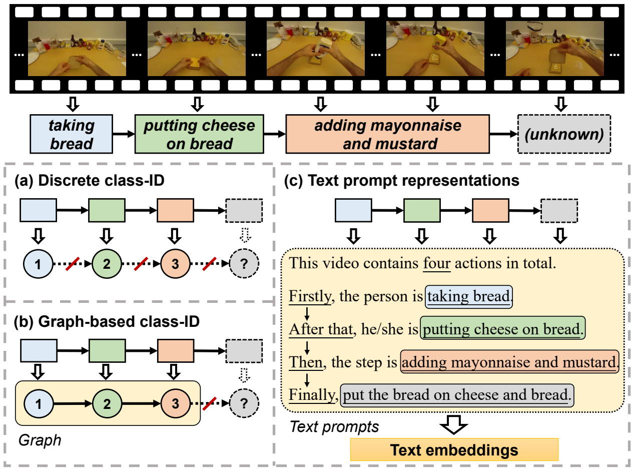
taking bread
putting cheese on bread
adding mayonnaise and mustard
(unknown)
(a) Discrete class-ID
1
2
3
?
(b) Graph-based class-ID
1
2
3
?
Graph
(c) Text prompt representations
This video contains four actions in total.
Firstly, the person is taking bread.
After that, he/she is putting cheese on bread.
Then, the step is adding mayonnaise and mustard.
Finally, put the bread on cheese and bread.
Text prompts
Text embeddings

Figure 1. Comparisons of conventional representations and Bridge-Prompt representations for ordinal actions. The human activity of *making cheese sandwich* contains four actions. Suppose the final action *putting the bread on cheese and bread* is unseen in training set. Conventional approaches in (a) and (b) are unable to depict the intra-semantics and inter-relations of all four actions, while our Bridge-Prompt representations in (c) is able to capture the full semantic information.

Recent years have witnessed the flourish of video analysis. Understanding human actions is the key to analyzing massive amounts of video data, which is conducive to a wide range of applications including video retrieval [8], video captioning [28] and video summarization [2]. Among the many sub-topics in action analysis, action recognition is a basic and core issue, which has made remarkable progress under various well-designed models [3, 5, 11].

Meanwhile, the current research trend of video analysis is experiencing a transition from understanding single-semantics short video clips to longer and more complex videos [38]. The increased attention on instructional video analysis has shown the significance of understanding semantically rich video contents [29, 38, 46]. From the perspective of action analysis, conventional action recognition approaches focus on classifying the single action being performed in a short video clip [5, 36]. In contrast, instructional video analysis methods need to study a series of actions being performed in longer time duration. In order to analyze instructional videos, we do not only need to understand the semantics of individual actions, but are also

[†]Corresponding author.

required to learn the semantic relations between contextual actions. Recently, some works have studied the mutual information between correlated actions in instructional videos using graph-based models [15, 30, 43]. The common approach is to regard each kind of action as a single node on a graph, where the edges between the nodes represent the contextual relations between adjacent actions.

However, the graph-based approaches are transductive, which are limited by the prior knowledge of input nodes and/or edges. Therefore, graph-based approaches are unable to address unknown types of nodes and thus are hard to extend and transfer. Moreover, under the existing framework of action recognition, the current way of depicting human actions is to allocate individual annotations to every single action, where different actions are treated as separate class IDs. This is practicable for recognizing separate actions, yet it is unable to depict the contextual relations between ordinal actions since individual class IDs cannot provide contextual information. The example of (a) and (b) in Figure 1 further illustrates the limitations of conventional class-ID-based approaches.

In this paper, we discover that human language is a powerful tool to depict the ordinal semantics between correlated actions. Human language is able to describe multiple sequentially occurred events based on ordinal numerals and specific sentence patterns. For example, ordinal relations between *taking bottle* and *pouring water* can be described in:“*the person firstly takes (the) bottle, and then pours water (into it)*”. The language naturally bridges the semantics between ordinal actions. In certain circumstances, even the textual descriptions of actions themselves can provide contextual information. For example, the ordinal relationship between actions of *taking bread*, *putting cheese on bread* and *putting bread on cheese and bread* is easy to be deduced literally. Moreover, language can intuitively extrapolate to unknown types of action. Given a new expression *putting bread on bread*, its semantics can be inferred from the expressions of known types of action. Figure 1(c) illustrates the effectiveness of language representations.

To this end, we propose a text-based learning method, Bridge-Prompt, for instructional video analysis. Motivated by the recent advances of prompt-based learning approach in Natural Language Processing (NLP) [25] and visual recognition [31], we introduce a three-plus-one-level design of text prompts to analyze the video clips containing a series of ordinal actions. Figure 1 shows the comparisons between conventional and Bridge-Prompt representations of ordinal actions. More specifically, we develop a prompt-based learning framework to jointly co-train the video and text encoders based on a specially designed video-text fusion module, so that we simultaneously exploit out-of-context and contextual action information towards a more comprehensive understanding of instructional videos. Our work digs deeper into the further potential of prompt-based learning approaches towards ordinal action understanding and instructional video analysis. Extensive experimental results on three benchmark datasets illustrate that the Bridge-Prompt-based approaches have achieved promising performances, and reach state-of-the-art on several benchmarks with the help of the prompt-based learning framework.

2. Related Work

Action analysis on instructional videos. Instructional video analysis is an increasingly popular trend in the field of video understanding. A wide variety of instructional video datasets have been proposed in recent years [29,38,45,47]. Instructional videos include profuse semantic information of human activities. The conventional approaches on action recognition [11,26,35,39] mainly focus on the datasets of trimmed video clips containing a single action in each video clip [5, 36]. Based on the existing studies of action recognition, several works have extended the action analysis methods to instructional videos by paying attention to the relations between ordinal actions. GTRM [15] utilizes a graph-based structure to depict the ordinal actions, and analysis is based on Graph Convolutional Networks (GCNs) [20]. GHRM [43] also represents ordinal actions as a graph, while focusing on the long-term action recognition task. Besides, Shao *et al.* [33] proposed the TransParser method for intra- and inter-action understanding via temporal action parsing. Different from the previous solutions, we make use of human language as a powerful semantic tool for analyzing ordinal actions in instructional videos.

Prompt-based learning on computer vision. Prompt-based learning approaches have been extensively studied in NLP [25, 32, 34]. The pioneer language model as GPT-3 [4] has shown its great few-show or zero-shot potential across various tasks. The core of prompt-based learning is to modify the input sample as a prompted version and embed the expected output information as an unfilled slot inside the prompt. CLIP [31] introduces the prompt-based learning approach into the image recognition task by embedding the textual labels of the to-be-recognized objects into descriptive texts, and the classification procedure can be transformed into a video-text matching problem. Following the prompt-based design, ALIGN [19] scales up the vision-language model by training on over one billion noisy image-text pairs and achieves better prompt-based prediction performances than CLIP. CoOp [44] utilizes learnable tokens as textual prompts and gains a promotion on few-shot image classification. CLIP-Adapter [12] combines the adapted features generated by the designed feature adapter with the CLIP feature to fit the few-shot classification. The prompt-based learning approach has not been widely developed on video understanding. ActionCLIP [40] proposes a specially designed prompt-based paradigm for action recog-

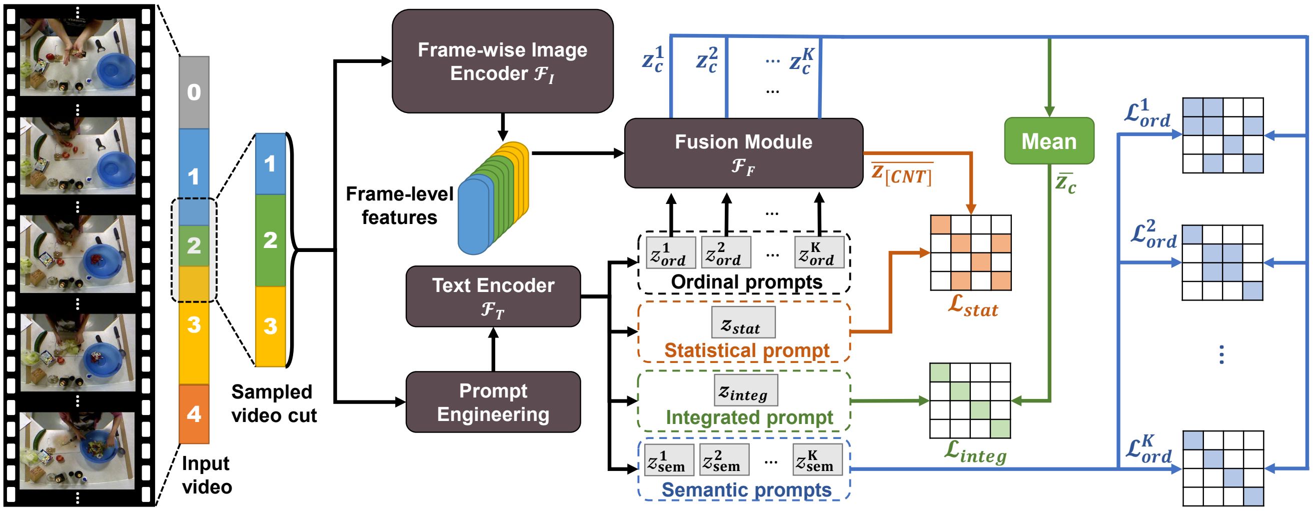

Figure 2. Overview of Bridge-Prompt pipeline. Bridge-Prompt takes the video cuts from minute-long raw inputs. After the special prompt engineering procedure, four types of text prompts are generated. Vision and text information are integrated both in the fusion module and during the video-text contrastive learning process. The proposed pipeline is able to capture the relations between ordinal actions.

nition, but it mainly focuses on recognizing single actions in short video clips. Our proposed Bridge-Prompt aims at analyzing instructional videos, which is more challenging but more conducive to understanding human behaviors.

3. Method

In this section, we introduce the overall pipeline design of Bridge-Prompt. The pipeline of our approach is illustrated in Figure 2.

3.1. Prompt Engineering

Prompt engineering refers to the design of an input text template that embeds the expected output strings as fill-in-the-blank formats [4] (*e.g.*, cloze test). The objective of our prompt engineering procedure is to design specific forms of text prompts to describe groups of ordinal actions in instructional videos. Suppose a series of single actions ($A = \{a_1, a_2, ..., a_K\}$) composes a specific kind of human activity. An easier way to design the prompts is to pose a blank-filling problem for every single action. For example, the prompt format as "*the person is* $\underline{\{vp_i\}}$ *right now*" (vp_i refers to the verb-phrase description for action a_i) can be used to abstract the semantics for each separate action of the character. However, since each action is still treated as an independent prompt instance, this strategy is unable to depict the contextual semantics between adjacent ordinal actions. For example, within the human activity of *scrambling egg*, the *stir-frying egg* action can only happen after *cracking egg*. A better form of text prompts towards ordinal action analysis should not only capture the out-of-context semantics of each separate action, but also bridge the gap between contextually related actions, and depict the overall semantics of the series of actions.

To better represent the series of actions in the Bridge-Prompt framework, we propose a three-plus-one-level design of prompt engineering for instructional videos: statistical prompt, ordinal prompt, semantic prompt, and integrated prompt. Considering the input video cut with K consecutive actions:

1) Statistical prompt captures the total count information for the series of actions. We use the format as "*this video clip contains* $\underline{\{num(K)\}}$ *actions in total*". The statistical prompt is denoted as y_{stat}.

2) Ordinal prompt captures the positional information for each action. We use the format as "*this is the* $\underline{\{ord_i\}}$ *action in the video*". The ordinal prompt is denoted as y_{ord}^i. The ordinal prompt set for x is denoted as:

$$\mathcal{Y}_{ord} = [y_{ord}^1, ..., y_{ord}^K] \tag{1}$$

3) Semantic prompt is the core of prompt design, which captures the semantic information of the actions. To integrate both out-of-context and contextual action information, we merge the ordinal information into the semantic prompts to create a multi-prompt format. We use the format as "$\underline{\{ord_i\}}$, *the person is performing the action step of* $\underline{\{vp_i\}}$" for action a_i. The semantic prompt set for x can be denoted as:

$$\mathcal{Y}_{sem} = [y_{sem}^1, ..., y_{sem}^K] \tag{2}$$

3+1) Integrated prompt captures the overall information for video x. The integrated prompt is formed by the integra-

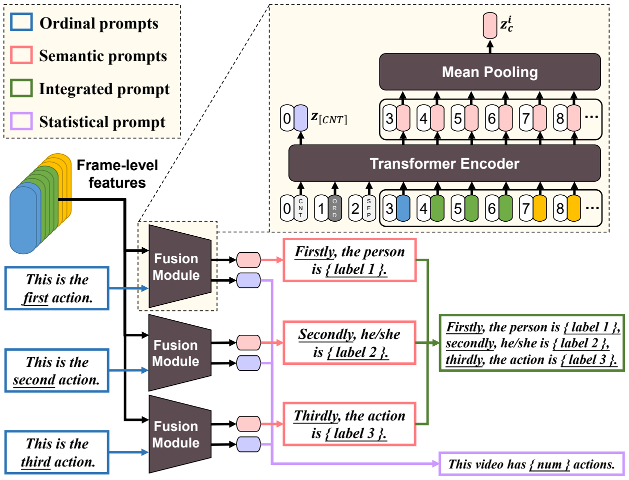
Ordinal prompts
Semantic prompts
Integrated prompt
Statistical prompt
Frame-level features
z_c^i
Mean Pooling
0
$z_{[CNT]}$
3 4 5 6 7 8 ...
Transformer Encoder
0 CNT 1 ORD 2 SEP 3 4 5 6 7 8 ...
This is the first action.
This is the second action.
This is the third action.
Fusion Module
Fusion Module
Fusion Module
Firstly, the person is { label 1 }.
Secondly, he/she is { label 2 }.
Thirdly, the action is { label 3 }.
Firstly, the person is { label 1 }, secondly, he/she is { label 2 }, thirdly, the action is { label 3 }.
This video has { num } actions.

Figure 3. Detailed illustration of prompt formats and fusion encoder. The fusion encoder takes the encoded frame-wise features and the ordinal prompt embeddings as inputs. It employs a learnable count token to analyze the statistical information. We adopt an ordinal-attention manner, meaning that the module only focuses on a single action with respect to a particular ordinal each time. The integrated semantics is extracted by mean-pooling operation.

tion of all semantic prompts $\mathcal{Y}_{sem}$. The integrated prompt y_{integ} can be denoted by:

$$y_{integ} = y_{sem}^1 \oplus y_{sem}^2 \oplus ... \oplus y_{sem}^K \tag{3}$$

where $\oplus$ refers to the string concatenation operation.

3.2. Bridge-Prompt: Framework

Sampling for raw videos. The raw instructional video sample $x_0 \in \mathbb{R}^{L_0 \times 3 \times H \times W}$ contains L_0 RGB frames of size $H \times W$. Usually, L_0 is different for each raw video. Moreover, suppose K_0 actions are contained in x_0, and K_0 is also unequal for different activities. Within each video, the duration of each action is unevenly distributed. We propose a sampling strategy by generating random video cut $x \in \mathbb{R}^{L_c \times 3 \times H \times W}$ from raw videos of a fixed length L_c to extract useful information while improving training efficiency. Each cut x may contain a single action or several successive actions, where K denotes the action count for x. The prompt engineering is conducted on those video cuts to generate the corresponding prompted text pair y. The sampling operation actually limits the temporal reception field of the model to a more localized range. The advantage of such a sampling strategy is to force the Bridge-Prompt model to focus more on the logical connections both within and between locally related actions.

Pre-training pipeline. The sampled video cut x with L_c frames $[f_1, ..., f_{L_c}]$ firstly passes through a frame-wise image encoder $\mathcal{F}_I$ to generate the frame-level features $[\mathcal{F}_I(f_1), ..., \mathcal{F}_I(f_{L_c})]$. Meanwhile, according to the prompt rules, a set of textual prompts $\{y_{stat}, \mathcal{Y}_{ord}, \mathcal{Y}_{sem}, y_{integ}\}$ can be generated for x. A text encoder $\mathcal{F}_T$ is introduced to extract the textual prompt embeddings $\{\mathbf{z}_{stat}, \mathbf{Z}_{ord}, \mathbf{Z}_{sem}, \mathbf{z}_{integ}\}$ respectively. The frame-level features are then passed through a fusion encoder $\mathcal{F}_F$ together with ordinal prompt embeddings to extract the clip-level feature $\mathbf{z}_c^i = \mathcal{F}_F(\mathcal{F}_I(f_1), ..., \mathcal{F}_I(f_{L_c}), \mathbf{z}_{ord}^i)$ for the i-th action of x. The design for the fusion module is the key to understanding both intra-action and inter-action information in x. We propose a Transformer-based structure for fusion. The information of ordinal prompt y_{ord}^i is fused into the fusion encoder to provide instructive information. We also embed a count token inside $\mathcal{F}_F$ to collect the quantitative information to be matched with statistical prompt y_{stat}. The details of the fusion approach for Bridge-Prompt pre-training will be discussed in the following sub-section. The clip-level feature is jointly learnt with both semantic prompts $\mathcal{Y}_{sem}$ and integrated prompt y_{integ} under a contrastive vision-text learning pattern.

Fusion module. The fusion encoder extracts the core information from the consecutive frame-level features. In other words, it tries to abstract the series of actions that occur in the input video clip. We utilize an ordinal-attention manner for the fusion module, *i.e.*, each time the fusion module only focuses on the action of a specific location. The ordinal-attention mechanism is implemented by adding the i-th ordinal prompt embeddings $\mathbf{z}_{ord}^i$ to the fusing inputs, which is an early-fusion strategy. We utilize a Transformer-Encoder structure for the fusion module. The input tokens of the fusion encoder include a learnable count token `[CNT]`, $\mathbf{z}_{ord}^i$ as a token `[ORD]`, a split token `[SEP]`, and L_c visual tokens representing frame-level features. `[ORD]` indicates which number of actions the fusion encoder is focusing on. The encoded representations of L_c frame-level features are mean-pooled to represent the clip-level feature. Besides, we added a learnable count token to learn additional quantitative information of actions. The encoded representation $\mathbf{z}_{[CNT]}$ for `[CNT]` will pass through the same contrastive vision-text learning framework with statistical prompt embeddings $\mathbf{z}_{stat}$ as a clip-level feature.

Joint vision-text representation learning. The joint vision-text representation learning maximizes the similarity between the encoded vision features and text features. A video clip x and its text description y can be encoded respectively with a video encoder and a text encoder, generating the clip representation $\mathbf{z}_x$ and the text representation $\mathbf{z}_y$. The similarity between $\mathbf{z}_x$ and $\mathbf{z}_y$ can be defined as their cosine distance:

$$s(\mathbf{z}_x, \mathbf{z}_y) = \frac{\mathbf{z}_x \cdot \mathbf{z}_y}{|\mathbf{z}_x| \, |\mathbf{z}_y|} \tag{4}$$

For a batch of the clip features $\mathcal{Z}_x$ and its corresponding batch of text features $\mathcal{Z}_y$, the batch similarity matrix S is:

$$S(\mathcal{Z}_x, \mathcal{Z}_y) = \begin{bmatrix} s(\mathbf{z}_{x_1}, \mathbf{z}_{y_1}) & \cdots & s(\mathbf{z}_{x_1}, \mathbf{z}_{y_B}) \\ \vdots & \ddots & \vdots \\ s(\mathbf{z}_{x_B}, \mathbf{z}_{y_1}) & \cdots & s(\mathbf{z}_{x_B}, \mathbf{z}_{y_B}) \end{bmatrix} \tag{5}$$

where B is the batch size. A text-wise/clip-wise softmax-normalization function can be applied respectively along rows/columns on $S(\mathcal{Z}_x, \mathcal{Z}_y)$, generating $S_T(\mathcal{Z}_x, \mathcal{Z}_y)$ and $S_V(\mathcal{Z}_x, \mathcal{Z}_y)$. A ground truth batch similarity matrix GT is defined where the similarity score of positive pair equals to 1, while negative pair equals 0. Our objective is to maximize the similarity between S and GT. We define the Kullback–Leibler (KL) divergence for matrices as the multi-modal contrastive loss:

$$D_{KL}(P \| Q) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i=1}^{N} P_{ij} \log \frac{P_{ij}}{Q_{ij}} \tag{6}$$

where P and Q are $N \times N$ matrices. The contrastive loss for video-text pair can be defined as:

$$\mathcal{L} = \frac{1}{2} \left[D_{KL}(S_T \| GT) + D_{KL}(S_V \| GT) \right] \tag{7}$$

Under the Bridge-Prompt framework, there are three parts of video-text contrastive losses in total:

i) $\mathbf{z}_c^i$ fused by the i-th ordinal token with $\mathbf{z}_{sem}^i$ of corresponding ordinal prompt, notated as $\mathcal{L}_{sem}^i$;

ii) mean-pooled $\overline{\mathbf{z}_c}$ fused by all ordinal tokens with $\mathbf{z}_{integ}$, notated as $\mathcal{L}_{integ}$;

iii) mean-pooled $\overline{\mathbf{z}_{[CNT]}}$ with $\mathbf{z}_{stat}$, notated as $\mathcal{L}_{stat}$;

The overall loss objective for the Bridge-Prompt pre-training framework is as follows:

$$\mathcal{L} = \sum_{i=1}^{K} \mathcal{L}_{sem}^i + \lambda_1 \mathcal{L}_{integ} + \lambda_2 \mathcal{L}_{stat} \tag{8}$$

where λ_1 and λ_2 balance the three losses.

3.3. Prompt-Based Inference

The *"pre-train, prompt, and predict"* paradigm in NLP has suggested that prompt-based design has the superiority of combining the objectives of downstream tasks into the pre-training procedure. The Bridge-Prompt framework has the capability of recognizing a series of actions by solving prompt-based cloze tests as "*this video clip contains* __ *actions in total*" or "__, *the person is performing the action of* __". In practice, we first generate the text features for all relevant ordinal prompts, statistical prompts, and semantic prompts by the pre-trained text encoder. For each test video, we extract the clip-wise features embedded by different ordinal prompts $\mathbf{z}_c^i$ and the average statistical representation $\overline{\mathbf{z}_{[CNT]}}$ using the pre-trained image encoder and fusion encoder. At first, we find the most matched embedding of statistical prompts with $\overline{\mathbf{z}_{[CNT]}}$ to determine the total count of actions. Then, we find the most matched embedding of semantic prompt with each ordinal prompt-embedded clip-wise feature $\mathbf{z}_c^i$ to determine each ordinal action one by one. As for the prompt variants, we vote among all variant formats to get the most matched prompt during inference stage.

4. Experiments

4.1. Datasets

We evaluate our proposed model on three challenging datasets. **50Salads** [37] contains 50 top-view 30-fps instructional videos regarding salad preparation. Totally 19 kinds of actions are contained in all videos. The 5-fold cross-validation is performed for evaluation, and the average results are reported. **Georgia Tech Egocentric Activities (GTEA)** [10] contains 28 egocentric 15-fps instructional videos of daily kitchen activities. Totally 74 classes of actions are summarized from all videos. We use the 4-fold cross-validation to evaluate the performances, and the average results are reported. **Breakfast** [21] contains 1,712 third-person 15-fps videos of breakfast preparation activities. 48 types of different actions are included in all 10 different kinds of breakfast activities. For evaluation, we use the train-split setting as proposed in [16], with 1357 videos for training and 355 videos for testing.

4.2. Implementation Details

Sampling strategy. The video cut sampling strategy is adjusted concerning frame rates and scales of different datasets. In general, we adopt a 16-frame window for each video cut. For GTEA dataset, we adopt multiple downsampling rates as 1, 2 and 4 respectively corresponding to the window striding rates of 2, 1 and 0.5. For 50Salads dataset, we use higher 24 and 32 downsampling rates with window striding rate of 1. For the Breakfast dataset, we a employ downsampling rate of 16 with a window striding rate of 2.

Bridge-Prompt architectures. For the image and text encoders, we follow the setups as CLIP [31] and ActionCLIP [40]. We adopt ViT-B/16 [7] as the image encoder $\mathcal{F}_I$, which is a 12-layer Transformer with input patch sizes of 16. The output representation for `[CLS]` token is regarded as the image feature. The text encoder $\mathcal{F}_T$ is also a 12-layer Transformer with the width of 512 and 8 attention heads. The output representation for `[EOS]` token is regarded as the text feature. The output frame-wise feature of the image encoder is a 768-dimensional vector, which is mapped to a 512-dimensional latent vector to match the embedded text features. For the fusion module $\mathcal{F}_F$, we employ a Transformer-Encoder-based structure to fuse the information of both image and text features. The fusion module contains 6 layers. As for the details of the prompt engineering procedure, we utilize an invariant prompt format for ordinal prompts and statistical prompts. With respect to the semantical prompts (which also contribute to integrated prompts), we adopt 19 variant prompt formats (9 short variant versions for integrated prompts) to describe the action semantics. The average similarity of all variants are computed during the prompt-based inference stage.

Training details. The image encoder and text encoder are together pre-trained on Kinetics-400 [5] by [40] before our training. We adopt AdamW [27] optimizer with the base learning rate of 5×10^{-6} with a 0.2 weight decay. The first 10% of training epochs are set as a warm-up phase, and the learning rate gradually decays down to zero during the remaining epochs under a cosine schedule. The spatial resolution of the input video is 224 × 224. For the loss function, we simply set $\lambda_1 = \lambda_2 = 1$. The model is trained for 50 epochs on GTEA and 50Salads, and 35 epochs on Breakfast. We use the batch size of 12 during training.

4.3. Results on Action Segmentation

The objective of action segmentation is to classify the action that occurs in each frame of a video [22]. Different from action recognition, action segmentation processes videos with multiple action instances. In consequence, action segmentation approaches should not only understand the out-of-context semantics for each separate action, but also be aware of the logical relations between adjacent actions. Several works have been conducted, and have achieved promising segmentation results. Most of the current SOTA approaches on action segmentation utilize the frame-wise I3D [5] features pre-trained on Kinetics extracted by [9], since the videos used for action segmentation are generally long videos that are hard to conduct direct analysis based on raw data. Bridge-Prompt utilizes a video cut-based approach to learn the contextual relations between adjacent actions locally, which is feasible on long videos. Since our approach is not specially designed for end-to-end action segmentation, we mainly adopt the Bridge-Prompt pre-trained image encoders to generate frame-wise features for raw videos. We test the action segmentation results based on current segmentation backbones.

Evaluation metrics. To evaluate the action segmentation results, we adopt several metrics including frame-wise accuracy (Acc), segmental edit distance, and the segmental F1 score at overlapping thresholds {10%, 25%, 50%}, denote by F1@{10,25,50}. The frame-wise accuracy is the most direct and frequently used metric, whereas it is unable to penalize the over-segmentation errors in long-duration actions. The segmental edit distance and segmental F1 score [22, 23] are proposed to handle over-segmentation errors and measure the segmentation quality.

Comparisons with the state-of-the-art. We compare the segmentation performances based on Bridge-Prompt-encoded frame-wise features with previous state-of-the-art methods. We use the ASFormer [42] as the backbone model for proceeding action segmentation. Bridge-Prompt is used as the pre-training approach to train the frame-wise image encoder (ViT). The output 768-dimensional frame-wise representations are regarded as the training inputs for the action segmentation backbone. In comparison, the previous state-of-the-art approaches use 2048-dimensional I3D features as training inputs. We conduct action segmentation on the GTEA dataset and the 50Salads dataset.

Table 1. Action segmentation results on GTEA dataset.

GTEA	**F1@{10,25,50}**			**Edit**	**Acc**
BCN [41]	88.5	87.1	77.3	84.4	79.8
MS-TCN++ [24]	88.8	85.7	76.0	83.5	80.1
ASRF [18]	89.4	87.8	79.8	83.7	77.3
G2L [13]	89.9	87.3	75.8	84.6	78.5
SSTDA [6]	90.0	89.1	78.0	86.2	79.8
SSTDA+HASR [1]	90.9	88.6	76.4	87.5	78.7
ASFormer (I3D) [42]	90.1	88.8	79.2	84.6	79.7
ASFormer (ViT)	88.5	86.2	77.6	87.1	75.6
Br-Prompt+ASFormer	**94.1**	**92.0**	**83.0**	**91.6**	**81.2**

Table 2. Action segmentation results on 50Salads dataset.

50Salads	**F1@{10,25,50}**			**Edit**	**Acc**
MS-TCN++ [24]	80.7	78.5	70.1	74.3	83.7
BCN [41]	82.3	81.3	74.0	74.3	84.4
SSTDA [6]	83.0	81.5	73.8	75.8	83.2
ASRF [18]	84.9	83.5	77.3	79.3	84.5
ASFormer (I3D) [42]	85.1	83.4	76.0	79.6	85.6
ASFormer+ASRF (I3D)	85.1	85.4	79.3	81.9	85.9
SSTDA+HASR [1]	86.6	85.7	78.5	81.0	83.9
Br-Prompt+ASFormer	**89.2**	**87.8**	**81.3**	**83.8**	**88.1**

Table 1, 2 compare the quantitative results of our approach. Specifically, we predict the 11 verbs of actions in GTEA for fair comparisons, and our method outperforms current state-of-the-art approaches under all five evaluation metrics. For comparison, we also evaluate the performances using raw features of ViT pre-trained by [40], which are inferior to the results using I3D-pre-trained features. However, after the ViT image encoder is further trained by Bridge-Prompt, the performances get obvious boosts. The performance of our approach also precedes previous state-of-the-art results on 50Salads. Figure 4 shows the qualitative illustration of action segmentation on both datasets.

4.4. Results on Long-Term Activity Recognition

A series of ordinal actions in instructional videos generally form a high-level semantics of human activity. The objective of long-term activity recognition is to classify the types of activities in long videos. Recognizing a high-level activity requires understanding the basic relations and temporal evolution of its ordinal sub-actions. Since Bridge-Prompt aims to study the relations between ordinal actions,

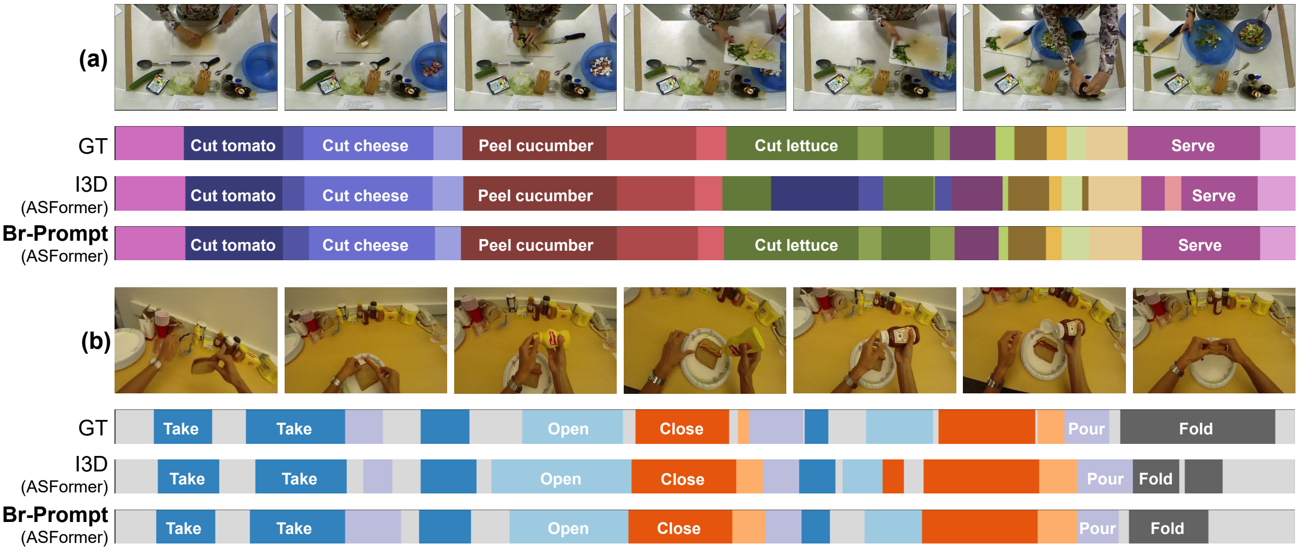

Figure 4. Qualitative results for action segmentation task on (a) 50Salads, and (b) GTEA dataset. Part of the actions are annotated on the color bar. The Br-Prompt pre-trained representation has greater potential on action segmentation task.

Table 3. Human activity recognition results on Breakfast dataset.

Method	Acc(%)
Kinetics pre-trained I3D	
I3D [5]	58.61
ActionVLAD [14]	65.48
Timeception [16]	67.07
VideoGraph [17]	69.45
GHRM [43]	75.49
Breakfast fine-tuned	
I3D (fine-tuned) [43]	74.83
Br-Prompt (fine-tuned)	**80.00**

it is also capable of long-term activity recognition. To adapt our framework for long-term action recognition, we first pre-train the frame-level encoder based on the Bridge-Prompt framework, and extract the frame-wise features for each video. Then, we uniformly sample 64 segments in each video with 8 frames per segment as in [16]. We use a simple Transformer-Encoder as a fusion module to respectively integrate segment-wise frames and different segments to generate video-wise representations. Then the human activities are predicted using prompt-based inferences.

Comparison with the state-of-the-art. The performances are evaluated on the Breakfast dataset as in Table 3. The performance of Bridge-Prompt fine-tuned features precedes I3D fine-tuned features. Since Bridge-Prompt is not a specially designed architecture for activity recognition, our straightforward prompt-based recognition approach may be inferior to more complicated recognition backbones based on fine-tuned I3D (*e.g.* GHRM [43]). The performance can be further improved by combining Bridge-Prompt representations with other high-level backbones.

Table 4. Comparisons of different fusion strategies for Bridge-Prompt by action segmentation results on GTEA dataset (split #1).

Fusion strategy	F1@{10,25,50}			Edit	Acc
(a) Vision-only	90.3	87.4	76.5	86.2	81.0
(b) Pos-embedding i.	89.1	86.2	77.5	84.8	80.0
(b) Pos-embedding ii.	88.7	87.3	76.4	84.0	79.5
(c) Weights for avg.	**91.8**	88.1	79.1	86.5	**83.7**
(d) Early-fusion	91.0	**89.6**	**82.1**	**88.7**	81.2

4.5. Ablation Studies

We perform several ablation studies on the GTEA dataset. Several adjustments have been conducted to evaluate the influence of different settings.

Fusion approaches. We have studied more kinds of fusion strategies to integrate statistical or ordinal information into frame-wise features. They are listed as follows:

(a) Vision-only fusion. Within the vision-only fusion, only the frame-wise features are regarded as inputs of the fusion Transformer. The output clip-wise features are contrastively learned together with statistical prompts, semantical prompts, and integrated prompts.

(b) Ordinal prompt fused as positional embedding. The ordinal prompt embedding can be linearly projected as an embedded vector with its length equal to the clip length. Then it is added to the input frame-wise features as part of the positional embedding after a mapping operation. There are two ways of mapping: i. repeating the embedded vector along the width dimension; ii. computing the outer product between the embedded vector and ordinal prompt embedding. The output clip-wise features are contrastively learned together with all formats of text prompts as (a).

Table 5. Comparisons of different loss choices for Bridge-Prompt by action segmentation results on GTEA dataset (split #1).

Loss components	F1@{10,25,50}			Edit	Acc
$\mathcal{L}_{sem}$	87.4	82.5	70.6	81.9	79.5
$\mathcal{L}_{sem}+\mathcal{L}_{integ}$	88.6	83.6	77.1	83.3	**81.2**
$\mathcal{L}_{sem}+\mathcal{L}_{integ}+\mathcal{L}_{stat}$	**91.0**	**89.6**	**82.1**	**88.7**	**81.2**

(c) Ordinal prompt fused as weights of average. The ordinal prompt embedding can be linearly projected as a weight vector with its length equal to the clip length. Then it is served as the weights of pooling operation for the input frame-wise features. The output weights are punished by an L2 loss function to avoid acquiring impulse-shape weights. The output clip-wise features are contrastively learned together with all formats of text prompts as (a) and (b).

(d) Early-fused ordinal prompts with a learnable count token. This is the fusion strategy adopted in our framework.

The action segmentation performances of different fusion strategies for Bridge-Prompt are evaluated on GTEA (split #1). Table 4 shows the quantitative results, which indicates that the fusion module is significant for improving the learning effectiveness of Bridge-Prompt. By merging ordinal information into the fusion module, the learned representations possess the focused information for each ordinal action. The fusion strategy (b) and (c) are more direct ways to integrate ordinal prompts, however, the ordinal prompt embeddings are not cross-attentioned with vision features. Specifically, the strategy (b) and (c) learn the information like “where may the first action be in any 16-frame video clip?”, while (d) focuses on “where is the first action among all the actions in this video?”. The location for each ordinal action also depends on other adjacent actions, which makes the early-fusion way more convincible.

Choice for loss functions. In our design, we consider three main components in the loss function: semantics, integrated semantics, and statistics. We perform ablation experiments to test the effectiveness of all three loss components. Table 5 shows the quantitative results, which indicates that all three losses make positive contributions to the final performance. It is reasonable since all the three text components are combined to depict both contextual and out-of-context semantics for a series of ordinal actions.

Transferability studies. Text is a flexible and extensible form of supervision. Different from class IDs, knowledge in texts can be transferred to unseen forms of script based on the generalization ability of pre-trained language models. To verify the transferability of Bridge-Prompt, we conduct a test on the prompt-based ordinal action inferences. For humans, action knowledge can be transferred between similar activities. As an example, a person can possibly learn how to *make tea* if he/she knows how to *make coffee*, since the sub-actions of the two activities are highly similar. For a class ID-based model, it is unable to transfer the knowledge between similar activities without manual interventions. Under prompt-based inferences, it is as simple as replacing the filling-in words in prompts. To quantitatively explain the transfer effects, we conduct experiments by training the framework on one human activity and evaluating the prompt inference accuracy on another one. The results are displayed in Table 6, which indicate that Bridge-Prompt has a promising zero-shot transferability.

Table 6. Prompt-based inference accuracies on GTEA. (coffee2tea refers to transferring the knowledge of *making coffee* to *making tea*, and so forth; AKL refers to training with all-knowing labels.)

Trans-type	coffee2 tea	cofhoney2 tea	hotdog2 pealate	peanut2 pealate	overall (AKL)
top-1 Acc(%)	38.8	41.7	15.5	24.6	54.5
top-5 Acc(%)	74.4	81.3	45.1	54.8	90.9

5. Conclusion and Discussion

In this paper, we have focused on the issue of ordinal action analysis in instructional videos. We proposed a prompt-based learning framework, Bridge-Prompt, which models the semantic relations across ordinal actions. To capture both out-of-context and contextual information of ordinal actions, text prompts are designed to integrate statistical, ordinal, and semantic information. Further experiments are conducted on two downstream tasks including action segmentation and long-term action recognition. The results have demonstrated that Bridge-Prompt has strong capability in the analysis of ordinal actions.

Limitations. Language can abstract the semantics from raw tedious videos. Although it is appealing to conduct large-scale vision-language pre-training on massive instructional video datasets such as HowTo100M [29], we are limited by the computing resources. Fortunately, we find that the manual label is a more accurate and concise form of semantic abstraction. With the help of pre-trained language models, we are able to learn the semantics of ordinal actions in a more efficient and accurate way based on text supervision.

Social impact. Despite the adaptiveness and convenience of the prompt-based approach to collaborate with vision models, it also means that fake labels are easier to create. To protect the vision-language model from potential attacks, label-filtering mechanisms and model self-inspections should be considered in practical applications.

Acknowledgments This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 62125603 and Grant U1813218, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI), and in part by the Postdoctoral Innovative Talent Support Program of China under Grant BX2021160.

References

[1] Hyemin Ahn and Dongheui Lee. Refining action segmentation with hierarchical video representations. In *ICCV*, pages 16302–16310, 2021. 6

[2] Evlampios E. Apostolidis, Eleni Adamantidou, Alexandros I. Metsai, Vasileios Mezaris, and Ioannis Patras. Video summarization using deep neural networks: A survey. *Proc. IEEE*, 109(11):1838–1863, 2021. 1

[3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, pages 6836–6846, 2021. 1

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, pages 1877–1901, 2020. 2, 3

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 1, 2, 6, 7

[6] Min-Hung Chen, Baopu Li, Yingze Bao, Ghassan AlRegib, and Zsolt Kira. Action segmentation with joint self-supervised temporal domain adaptation. In *CVPR*, pages 9454–9463, 2020. 6

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 5

[8] Maksim Dzabraev, Maksim Kalashnikov, Stepan Komkov, and Aleksandr Petiushko. Mdmmt: Multidomain multimodal transformer for video retrieval. In *CVPRW*, pages 3354–3363, 2021. 1

[9] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *ICCV*, pages 3575–3584, 2019. 6

[10] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR*, pages 3281–3288, 2011. 5

[11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *CVPR*, pages 6202–6211, 2019. 1, 2

[12] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 2

[13] Shang-Hua Gao, Qi Han, Zhong-Yu Li, Pai Peng, Liang Wang, and Ming-Ming Cheng. Global2local: Efficient structure search for video action segmentation. In *CVPR*, pages 16805–16814, 2021. 6

[14] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *CVPR*, pages 971–980, 2017. 7

[15] Yifei Huang, Yusuke Sugano, and Yoichi Sato. Improving action segmentation via graph-based temporal reasoning. In *CVPR*, pages 14024–14034, 2020. 2

[16] Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. Timeception for complex action recognition. In *CVPR*, pages 254–263, 2019. 5, 7

[17] Noureldien Hussein, Efstratios Gavves, and Arnold WM Smeulders. Videograph: Recognizing minutes-long human activities in videos. In *ICCVW*, 2019. 7

[18] Yuchi Ishikawa, Seito Kasai, Yoshimitsu Aoki, and Hirokatsu Kataoka. Alleviating over-segmentation errors by detecting action boundaries. In *WACV*, pages 2322–2331, 2021. 6

[19] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, pages 4904–4916, 2021. 2

[20] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017. 2

[21] Hilde Kuehne, Ali Bilgin Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, pages 780–787, 2014. 5

[22] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *CVPR*, pages 156–165, 2017. 6

[23] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, pages 36–52, 2016. 6

[24] Shi-Jie Li, Yazan AbuFarha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE TPAMI*, pages 1–1, 2020. 6

[25] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 2

[26] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 2

[27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[28] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 1

[29] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640, 2019. 1, 2, 8

[30] Megha Nawhal and Greg Mori. Activity graph transformer for temporal action localization. *arXiv preprint arXiv:2101.08540*, 2021. 2

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 2, 5

[32] Timo Schick and Hinrich Schütze. Exploiting cloze-questions for few-shot text classification and natural language inference. In *EACL*, pages 255–269, 2021. 2

[33] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Intra-and inter-action understanding via temporal action parsing. In *CVPR*, pages 730–739, 2020. 2

[34] Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *EMNLP*, pages 4222–4235, 2020. 2

[35] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, pages 568–576, 2014. 2

[36] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 1, 2

[37] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *ACM*, pages 729–738, 2013. 5

[38] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *CVPR*, pages 1207–1216, 2019. 1, 2

[39] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2

[40] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021. 2, 5, 6

[41] Zhenzhi Wang, Ziteng Gao, Limin Wang, Zhifeng Li, and Gangshan Wu. Boundary-aware cascade networks for temporal action segmentation. In *ECCV*, pages 34–51, 2020. 6

[42] Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. *arXiv preprint arXiv:2110.08568*, 2021. 6

[43] Jiaming Zhou, Kun-Yu Lin, Haoxin Li, and Wei-Shi Zheng. Graph-based high-order relation modeling for long-term action recognition. In *ICCV*, pages 8984–8993, 2021. 2, 7

[44] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134*, 2021. 2

[45] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, pages 7590–7598, 2018.

[46] D Zhukov, J. B. Alayrac, R. G. Cinbis, D Fouhey, I. Laptev, and J. Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, pages 3537–3545, 2019. 1

[47] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, pages 3537–3545, 2019. 2