Prompting Visual-Language Models for Efficient Video Understanding

Author: Chen Ju et al.
Presenters: Myles, Wei, Taixi, Jeff

Rebuttal

1. Simplicity

Ours:

- Simple pipeline
- Easily adapted to different tasks

Theirs:

- Complicated
- Spatial adaptation, temporal adaptation, …

2. Faster training, lower cost

Ours:

- Only optimises several prompt vectors and two Transformer layers
- Everything can be done with one RTX 3090 GPU

Individuals can train easily

Theirs:

- Much more parameters to tune
- Training cost more

… X56

Only for rich people

3. Image Backbone

Image backbone is not an issue but might be an advantage:

- Image data is much easier to get in real world than video data.
- Video transformer model is data-hungry. Image model can achieve a better performance in a lot of real-world application with limited data.

4. Prompt Learning vs. Fine-tuning

Ours:

- One unified powerful video model in one day.
- A video GPT?

Theirs:

- Fine-tune for specific application.
- Limited impact.
- Hundreds of different models.
- Zero-shot?

AIM: Adapting Image Models for Efficient Video Action Recognition

Taojiannan Yang[1], Yi Zhu[2], Yusheng Xie[2], Aston Zhang[2], Chen Chen[1], Mu Li[2] [1]University of Central Florida [2]Amazon Web Services

Presented By:

Annie, Shoubin, Nick, Junjie

Comparison: AIM v.s. PromptCLIP

AIM 😍

👁️ image model for visual tasks, flexiable to a large number of visual models

🤩 wow! reusing spatial layers as temporal layers can work?!

👁️ simple, efficient, and **unified,** more likely to be a new foundation video modeling scheme

👍 Better result but less input view, less/similar parameter!

PromptCLIP 🤔

💬 image-language model for visual tasks, need an extra text model

🤖 everything logical and make sense, but less excited

🤦‍♂️ A + B + C

😐 cool performance on a wide range tasks

Comparison: AIM v.s. PromptCLIP

Table 2: Comparison to state-of-the-art on Kinetics-400. Views = #frames × #temporal × #spatial.

Methods	Pretrain	GFLOPs	Param (M)	Tunable Param (M)	Top-1	Top-5	Views
MViT-B (Fan et al., 2021)	-	4095	37	37	81.2	95.1	64×3×3
UniFormer-B (Li et al., 2021)	IN-1K	3108	50	50	83.0	95.4	32×4×3
TimeSformer-L (Bertasius et al., 2021)	IN-21K	7140	121	121	80.7	94.7	64×1×3
ViViT-L/16×2 FE (Arnab et al., 2021)	IN-21K	3980	311	311	80.6	92.7	32×1×1
VideoSwin-L (Liu et al., 2022)	IN-21K	7248	197	197	83.1	95.9	32×4×3
MViTv2-L (312 ↑) (Li et al., 2022)	IN-21K	42420	218	218	86.1	97.0	32×3×5
MTV-L (Yan et al., 2022)	JFT	18050	876	876	84.3	96.3	32×4×3
TokenLearner-L/10 (Ryoo et al., 2021)	JFT	48912	450	450	85.4	96.3	64×4×3
PromptCLIP A7 (Ju et al., 2021)	CLIP	-	-	-	76.8	93.5	16×5×1
ActionCLIP (Wang et al., 2021a)	CLIP	16890	142	142	83.8	97.1	32×10×3
X-CLIP-L/14 (Ni et al., 2022)	CLIP	7890	420	420	87.1	97.6	8×4×3
EVL ViT-L/14 (Lin et al., 2022)	CLIP	8088	368	59	87.3	-	32×3×1
AIM ViT-B/16	CLIP	606	97	11	83.9	96.3	8×3×1
AIM ViT-B/16	CLIP	1214	97	11	84.5	96.6	16×3×1
AIM ViT-B/16	CLIP	2428	97	11	84.7	96.7	32×3×1
AIM ViT-L/14	CLIP	2802	341	38	86.8	97.2	8×3×1
AIM ViT-L/14	CLIP	5604	341	38	87.3	97.6	16×3×1
AIM ViT-L/14	CLIP	11208	341	38	**87.5**	**97.7**	32×3×1

Comparison: AIM v.s. PromptCLIP

PromptCLIP

For all the following action recognition experiments, we inherit the best practice from the ablation studies, *i.e.* prepend / append 16 prompt vectors to category names, and only use two Transformer layers (5M parameters) for temporal modeling, for its best trade-off on performance and computational cost.

AIM

Table 7: Effect of position of Adapters. Skip means adding Adapters every two blocks.

Position	Tunable Param (M)	Top-1
Bottom 6	5.6	80.7
Top 6	5.6	83.3
Skip	5.6	83.2
All	11	**83.9**