MULTI-MODAL FEW-SHOT TEMPORAL ACTION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Conventional temporal action detection (TAD) methods rely on supervised learning from many labeled training videos, rendering them unscalable to new classes. Recent approaches to solving this problem include few-shot (FS) and zero-shot (ZS) TAD. The former can adapt a pretrained vision model to a new task represented by as few as a single video per class, whilst the latter synthesizes some semantic description given a new class (*e.g.*, generating the classifier using a pretrained vision-language (ViL) model). In this work, we further introduce a hybrid problem setup, *multi-modality few-shot* (MMFS) TAD, that integrates the respective advantages of FS-TAD and ZS-TAD by accounting for both few-shot support videos (*i.e.*, visual modality) and new class names (*i.e.*, textual modality) in a single formula. To tackle this MMFS-TAD problem, we introduce a novel ***MUlti-modality PromPt mETa-learning*** (MUPPET) method. Our key idea is to construct multi-modal prompts by mapping few-shot support videos to the textual token space of a pretrained ViL model (*e.g.*, CLIP) using a meta-learned adapter-equipped visual semantics tokenizer; This facilitates a joint use of the two input modalities for learning richer representation. To address the large intra-class variation challenge, we further design a query feature regulation scheme. Extensive experiments on ActivityNetv1.3 and THUMOS14 demonstrate that our MUPPET outperforms state-of-the-art FS-TAD, ZS-TAD and alternative methods under a variety of MMFS-TAD settings, often by a large margin.

1 INTRODUCTION

The objective of temporal action detection (TAD) is to predict the temporal duration (*i.e.*, start and end time) and the class label of each action instance in an untrimmed video (Idrees et al., 2017; Caba Heilbron et al., 2015). By supervised learning on many (*e.g.*, hundreds) videos with costly segment-level annotations, conventional TAD methods (Xu et al., 2021; 2020a; Buch et al., 2017; Wang et al., 2017; Zhao et al., 2017; Nag et al., 2022a; 2021a) are hardly scalable in practice. To alleviate this problem, `few-shot` (FS) (Yang et al., 2018; 2020; Zhang et al., 2020; Nag et al., 2021b) and `zero-shot` (ZS) (Zhang et al., 2020; Ju et al., 2022; Nag et al., 2022b) learning based TAD methods have been recently introduced. Specifically, FS-TAD aims to learn a model that can adapt to a new task with as few as a single training video per class (Fig. 1(a)). This is achieved often by meta-learning a TAD model over a distribution of simulated tasks on seen classes.

Instead, ZS-TAD only needs to translate the new class names into some semantic space (*e.g.*, attributes, word embeddings), without any training samples (Fig. 1(b)). Typically, visual feature has been aligned with this semantic space during training so that the model can be directly applied. In particular, the emergence of ever stronger Visual-Language (ViL) models (*e.g.*, CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021)) have surged significantly the research of *zero-shot transfer* across diverse problems. Central in this research line is a favourable ability of *synthesizing the classification weights* from text (*e.g.*, action class name or description) that are semantically aligned with the feature space of an image encoder. For example, Ju et al. (2022) first generate action proposals and then classify them with a pretrained CLIP in two stages. Whilst Nag et al. (2022b) present a single-stage TAD model with parallel action localization and ViL infused action classification for suppressing localization error propagation.

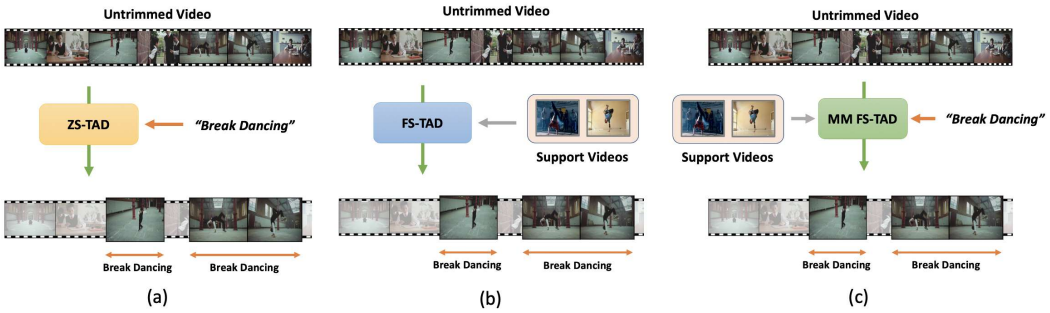

Figure 1: **Illustration of different problem settings**. (a) *Zero-shot temporal action detection* (ZS-TAD) translates new class name (*i.e.*, textual input) into the inference process. (b) *Few-shot temporal action detection* (FS-TAD) can rapidly learn a new class from a few support (training) videos (*i.e.*, visual input). (c) To enjoy the advantages of ZS-TAD and FS-TAD, we introduce *multimodal few-shot temporal action detection* (MMFS-TAD) where both textual and visual modality inputs can be jointly leveraged for detecting new action classes.

In the literature, FS-TAD and ZS-TAD both are still under-studied *independently*, partly due to different nature of their key challenges. However, we instead focus on their intrinsic advantages. In FS-TAD, support videos offer the model the most detailed and fine-grained visual details, and can be mapped to the same embedding space as the test/query videos without cross-modal alignment challenges. In contrast, ZS-TAD is favored by the ability of textual-visual modality alignment, only needing to take the new class names as input whilst not considering any visual training samples. Motivated by their respective strengths, we propose a new hybrid problem setup, namely `multimodal few-shot temporal action detection` (MMFS-TAD), characterized by learning from both support videos (*i.e.*, visual modality) and class names (*i.e.*, textual modality) in a single formula for stronger generalization.

To address the proposed MMFS-TAD problem, we introduce a novel *Multi-Modality Prompt Meta-Learning* (**MUPPET**) method. The key objective is to efficiently fuse few-shot visual examples and high-level class semantic text information. To that end, grounded on a pre-trained vision-language (ViL) model (*e.g.*, CLIP), we integrate meta-learning with learning-to-prompt in a unified TAD framework. This is made possible by introducing a *multimodal prompt learning module* that maps the support videos of a novel task to the textual token space of the ViL model using a meta-learned adapter-equipped visual semantics tokenizer. With the ViL's text encoder, our multimodal prompt can be then transformed to multimodal class prototypes for action detection. To tackle the large intra-class challenge due to limited support samples, we further design a query feature regulation strategy by meta-learning a masking representation from the support sets and attentive conditioning.

We summarize our **contributions** as follows. **(1)** We propose the multimodal few-shot temporal action detection (MMFS-TAD) problem, with combined advantages of conventional FS-TAD and ZS-TAD settings. **(2)** To solve this new problem, we introduce a novel *Multi-Modality Prompt Meta-Learning* (MUPPET) method that integrates meta-learning and learning-to-prompt in a single formulation. It can be easily plugged into existing TAD architectures. **(3)** To better relate query videos with limited support samples, we design a query feature regulation scheme based on meta-learning a masking representation from the support sets and attentive conditioning. **(4)** We conduct extensive experiments on ActivityNet-v1.3 and THUMOS14 to validate the superiority of our MUPPET over state-of-the-art FS-TAD and ZS-TAD methods.

2 Related Works

Temporal action detection Substantial progress has been achieved in TAD. Inspired by object detection (Ren et al., 2016), R-C3D (Xu et al., 2017) uses anchor boxes in the pipeline of proposal generation and classification. Similarly, TURN (Gao et al., 2017) aggregates local features to represent snippet features for temporal boundary regression and classification. SSN (Zhao et al., 2017) decomposes an action instance into start:course:end and employs structured temporal pyramid pool-

ing for proposal generation. BSN (Lin et al., 2018) generates proposals with high start and end probabilities by modeling the start, end and actionness at each time. Later, BMN (Lin et al., 2019) improves the actionness via generating a boundary-matching confidence map. For better proposal generation, G-TAD (Xu et al., 2020a) learns semantic and temporal context via graph convolutional networks. CSA (Sridhar et al., 2021) enriches the proposal temporal context via attention transfer. Unlike most previous models adopting a sequential localization and classification pipeline, TAGS (Nag et al., 2022a) introduces a different design with parallel localization and classification based on a notion of global segmentation masking. All the above methods are supervised with reliance on large training data, and thus less scalable.

Few-shot temporal action detection By fast adaptation of a model to any given new class with few training samples, few-shot learning (FSL) provides a solution for scalability (Vinyals et al., 2016; Sung et al., 2018; Snell et al., 2017). FSL is often realized by meta-learning which simulates new tasks with novel classes represented by only a handful of labeled samples. FSL has been introduced to TAD. Yang et al. (2018) incorporate sliding window in the matching network (Vinyals et al., 2016). Later on, Zhang et al. (2020) consider weak video-level annotation of untrimmed training videos. Yang et al. (2021) performed few-shot spatio-temporal action detection with focus on a single new class at a time. Recently, Nag et al. (2021b) used the Transformer for adapting the support learned features to the query features in untrimmed videos.

Zero-shot temporal action detection Alternatively, zero-shot learning allows for recognizing new classes with no labeled training data. This line of research has advanced significantly due to the promising power of large vision-language (ViL) models, for instance, CLIP trained by 400 million image-text pairs (Radford et al., 2021). Many follow-ups further boost the zero-shot transferable ability, *e.g.*, CoOp (Zhou et al., 2021), CLIP-Adapter (Gao et al., 2021), and Tip-adapter (Zhang et al., 2021b). In video domains, similar idea has also been explored for transferable representation learning (Miech et al., 2020), and text based action localization (Paul et al., 2021). CLIP is used recently in action recognition (Wang et al., 2021) and TAD (Ju et al., 2022; Nag et al., 2022b).

Instead of improving FS-TAD or ZS-TAD, in this work we combine their respective advantages by introducing a new problem setting - multimodal few-shot temporal action detection (MMFS-TAD). We further contribute a novel MUPPET method designed particularly for this new problem.

3 METHODOLOGY

MMFS-TAD setting Given a new task with a few labeled support videos per unseen class (*i.e.*, visual modality) and class names (*i.e.*, textual modality), we aim to learn a TAD model for that task. We have a base class set C_{base} for training, and a novel class set C_{novel} for test. For testing cross-class generalization, we ensure they are disjoint: $C_{base} \bigcap C_{novel} = \phi$. The base and novel sets are denoted as $D_{base} = \{(V_i, Y_i), Y_i \in C_{base}\}$ and $D_{novel} = \{(V_i, Y_i), Y_i \in C_{novel}\}$ respectively. Under the proposed setting, each training video V_i is associated with segment-level annotation $Y_i = \{(s_t, e_t, c), t \in \{1, .., M\}, c \in C\}$ including M segment labels each with the start and end time and action class. In evaluation, for each task, we randomly sample a set of classes $L \sim C_{novel}$ each with the support set S (K videos) and the query set Q (one video) respectively. The labels of S are accessible for few-shot learning whilst that of Q only used for performance evaluation.

3.1 MODEL ARCHITECTURE

We construct our MUPPET in the recent global segmentation mask based TAD architecture that predicts mask based action instances per snippet (Nag et al., 2022a;b). An overview is depicted in Fig. 2. Grounded on the pretrained (frozen) ViL model (*e.g.*, CLIP (Radford et al., 2021)), our MUPPET brings two key components: **(a)** Multimodal prompt learning, and **(b)** Query re-weighting.

3.2 MULTIMODAL PROMPT META-LEARNING

Per-task video feature adaptation Existing pretrained ViL models ((Radford et al., 2021; Wang et al., 2021)) are not designed for TAD, with a need for domain adaptation. Given the big model size and scarce labeled training data, we adopt the adapter (Chen et al., 2022) strategy so that only a fraction of parameters need to be learned. This eases the training. Let θ the pretrained vision

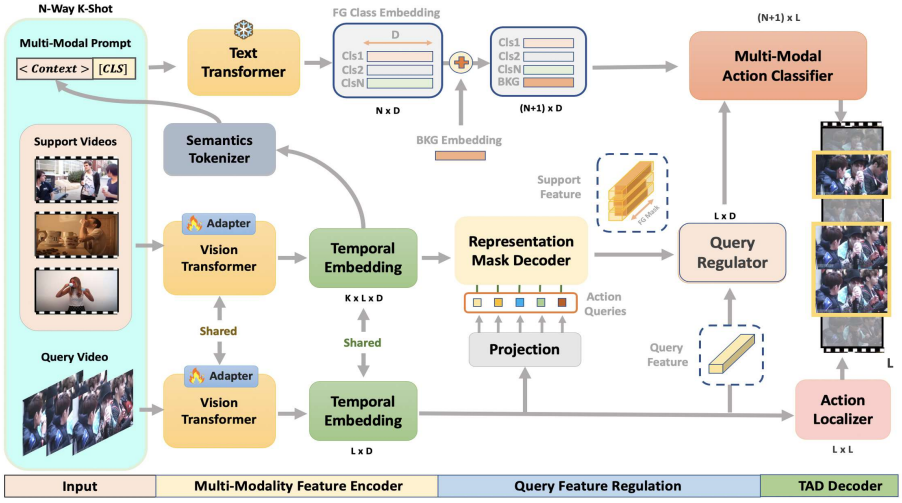

Figure 2: **Overview of our *Multi-Modality Prompt Meta-Learning* (MUPPET) method**. We adopt the global mask based TAD architecture (Nag et al., 2022a;b). Key components of MUPPET include (1) multimodal prompt meta-learning (Sec. 3.2), (2) query feature regulation (Sec. 3.3).

transformer $\mathbb{V}$ (part of ViL model) and ϕ_{Ti} the adapter. Given a set of few-shot tasks $T_i = \{S_i, Q_i\}$ and the corresponding video frames f_{T_i}, we optimize the adapters in a episodic fashion. We obtain the episodic video features from the adapter infused video encoder as:

$$F_{T_i} = \mathbb{V}(\theta, \phi_{T_i}, f_{T_i}) \in \mathbb{R}^{t \times D} \tag{1}$$

where D is the snippet feature dimension and t is the number of temporal snippets. Following (Lin et al., 2019; Xu et al., 2020b), for each video we uniformly sample L equidistant points over the entire snippets t to obtain the episodic adapter features $F_E \in \mathbb{R}^{L \times D}$. To capture global context, we further leverage self-attention (Vaswani et al., 2017). Formally, we set the input (`query`, `key`, `value`) of a Transformer encoder $\mathcal{T}()$ as the features (F_E,F_E,F_E). Positional encoding is not applied as it is found to be detrimental. The final video snippet embedding is then obtained as

$$E = \mathcal{T}(F_E) \in \mathbb{R}^{L \times D}, \tag{2}$$

with D being the embedding dimension. We denote support features as $E_s \in \mathbb{R}^{N \times K \times D \times L}$ and query features as $E_q \in \mathbb{R}^{N \times D \times L}$ where N is the number of classes per episode and K the number of shots per class respectively.

Multimodal prompt meta-learning We aim to fuse visual and textual modalities for stronger representation. Inspired by recent prompt learning for vision tasks (Zhou et al., 2021), we design a novel *multimodal prompt meta-learning strategy* particularly for our MMFS-TAD problem, as shown in Fig. 3. Specifically, we introduce a visual semantics tokenizer f_θ to align the support videos with the ViL's text tokenizer as:

$$\hat{w}_c = f_\theta(\hat{E}_s^k | k = 1, 2, ...K) \in W, \tag{3}$$

where $\hat{E}_s$ denotes the action feature of support videos obtained by masking E_s with the annotation, and W the textual token space. A set Transformer (Lee et al., 2019) is used to implement f_θ.

Instead of learning a common prompt embedding for all the target classes (Zhou et al., 2021), we now learn a *class-specific token embedding*. This allows to generate more discriminative tokens and class representation (Table 2). We design the multimodal prompt as $\hat{p} = [\hat{w}_c][T_c]$ where T_c is the token of action class name obtained by the text tokenzier. As such, we can leverage the ViL model's text encoder $\mathbb{T}()$ as

$$\hat{z}_c = \mathbb{T}(\hat{p}) \in \mathbb{R}^{C \times D}, \tag{4}$$

to obtain the multi-modal representation $\hat{z}_c$ with both visual (support set) and textual (class name) encoded for action class c.

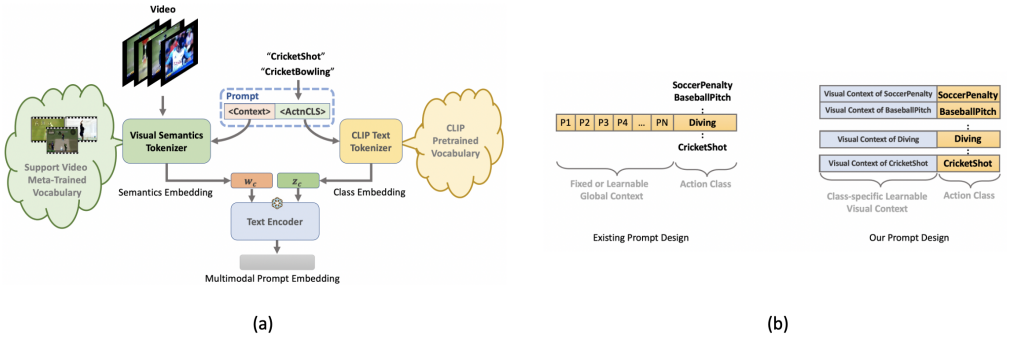

Figure 3: **Multimodal prompt meta-learning**. (a) We meta-learn a visual semantics tokenizer for translating the support videos (*i.e.*, visual modality) to the textual token space of a pretrained ViL model. Together with the tokens of class names, this mapping facilitates the creation of multimodal prompts using the pretrained text encoder. (b) Unlike previous class-generic visual prompts, we consider more discriminative class-specific counterparts.

For TAD, we need a background class which however is lacking from the vocabulary of ViL model. To solve this, we learn a specific background embedding, denoted as $\hat{z}_{bg} \in \mathbb{R}^D$, from random initialization. We append this to $\hat{z}_c$, yielding a complete multimodal representation $E_{mm} \in \mathbb{R}^{(C+1)\times D}$.

3.3 Query Feature Regulation

To facilitate the association of action instances across support and query videos in the same action class with typically large differences (*i.e.*, large intra-class variation), we design a *query feature regulation* scheme based on support-conditioned representation masking. This is inspired by representation masking for suppressing the background (Nag et al., 2022b).

Support-conditioned representation masking Concretely, given per-class temporal features of a query video $E_q \in \mathbb{R}^{D\times L}$, we obtain a transformed feature $Q_{act} \in \mathbb{R}^{1\times D}$ using a MLP layer. We then repeat Q_{act} for $\mathbb{N}_q$ times to obtain the action query $Q_{act} \in \mathbb{R}^{\mathbb{N}_q\times D}$. Together with the support video features $E_s \in \mathbb{R}^{K\times D\times L}$, we use a mask-attention based Transformer decoder (Cheng et al., 2022) to generate $\mathbb{N}_q$ latent embeddings, followed by a masking projection layer to obtain a mask embedding for each segment as $B_q \in \mathbb{R}^{K\times q\times D}$ where q indexes a query. A binary mask prediction w.r.t each query can be then calculated as:

$$L_q = \sigma(B_q * E_s) \in \mathbb{R}^{K\times q\times L}, \tag{5}$$

where σ is sigmoid activation. As such, each snippet location of support videos is associated with q queries. To choose the optimal query per location, we deploy a tiny MLP to weigh these queries in a location specific manner. This is realized by learning a weight vector $W_q \in \mathbb{R}^{K\times q}$ as:

$$\hat{L} = \sigma(W_q * L_q + b_q) \in \mathbb{R}^{K\times L}. \tag{6}$$

where b_q is a bias term. We then binarize this support video mask at a threshold θ_{bin} and select the foreground mask $\hat{L}_{bin}$. The support masked representation E_s^{fg} is obtained by using $\hat{L}_{bin}$ to retrieve the snippet embedding E_s.

Query feature regulation Next, we use the support masked feature for regularizing the query feature by cross-attention. Specifically, for a transformer encoder $\mathcal{C}$, we set the query video feature as its query $\mathbb{Q}$, and the support masked feature as its key $\mathbb{K}$ and value $\mathbb{V}$. As the number of support videos per class varies, we aggregate $\mathbb{K}$ and $\mathbb{V}$ by averaging over the number of shots to match a query video. We then concatenate $\mathbb{K}$/$\mathbb{V}$ with the query feature to form an enhanced version as:

$$\mathbb{K}_{agg} = (\mathbb{Q}, \frac{1}{K}\sum_{i=1}^{K} E_s^{fg}) \in \mathbb{R}^{2L\times D}, \quad \mathbb{V}_{agg} = (\mathbb{V}, \frac{1}{K}\sum_{i=1}^{K} E_s^{fg}) \in \mathbb{R}^{2L\times D}. \tag{7}$$

The query feature is finally regulated via $\overline{E}_q = \mathcal{C}(E_q, \mathbb{K}_{agg}, \mathbb{V}_{agg})$.

3.4 TAD DECODER (HEAD)

We adopt the TAD head of (Nag et al., 2022b;a) with parallel classification and mask prediction.

Multimodal classifier We exploit $\hat{E}_{mm} \in \mathbb{R}^{(C+1)\times D}$ as a multimodal classifier and apply to the regulated query video features $\overline{E}_q \in \mathbb{R}^{L\times D}$ as:

$$\mathcal{P} = (\hat{E}_{mm} * (\overline{E}_q)^T)/\tau \in \mathbb{R}^{(K+1)\times L}, \tag{8}$$

where each column of $\mathcal{P}$ is the classification result $p_l \in \mathbb{R}^{(K+1)\times 1}$ of each snippet $t \in L$, and $\tau = 0.7$ is a temperature coefficient.

Action mask localizer In parallel to classification, this stream predicts 1-D binary masks of action instances over the whole video. We use stacks of 1D dynamic-convolution layers to form the mask classifier $\mathbb{H}$. Specifically, given t-th snippet $\overline{E}_q(t)$, it outputs a 1-D mask $m_t = [q_1, ..., q_L] \in \mathbb{R}^{L\times 1}$ with each $q_i \in [0, 1](i \in [1, L])$ giving action probability at i-th snippet. We write formally:

$$\mathcal{M} = \rho(\mathbb{H}(E_q)) \tag{9}$$

where ρ is a sigmoid activation and t-th column of $\mathcal{M}$ is the mask prediction by t-th snippet.

3.5 MODEL TRAINING AND INFERENCE

Learning objective Following (Nag et al., 2022b), we adopt cross-entropy loss L_c for classification, binary dice loss L_m and binary mask loss L_{comp} for masking. We further impose a contrastive criterion (Chen et al., 2020) to optimize the visual semantics tokenizer. Given the multimodal representation $\hat{z}_c$, original prompt embedding z_c, and video embedding $\overline{z}_c$ for each class c, the contrastive loss is defined as:

$$L_{tok} = -log\frac{\exp(cos(\overline{z}_c, \hat{z}_c))}{\exp(cos(\overline{z}_c, \hat{z}_c)) + 2\exp(cos(z_c, \hat{z}_c))}, \tag{10}$$

where $cos(.)$ is cosine similarity and the factor 2 is for contrasting z_c with both visual and textual embedding. To contrast the background (z_{bg}) from foreground ($\hat{z}_c$), we minimize:

$$L_{bg} = argmin \sum_{j=1}^{C} (cos(z_{bg}, z_c^j) - \delta_{bg})^2, \tag{11}$$

where δ_{bg} is the margin hyper-parameter.

Training Our MUPPET is trained in two stages. In stage-1 for supervised training, we deploy the objective $L_{base} = L_c + L_m + L_{comp} + L_{tok} + L_{bg} + L_{const}$; In stage-2 for meta-training, L_m and L_c are removed from the objective, due to no access to the ground-truth of query videos. Our model is trained end-to-end in each stage, with the pretrained text encoder frozen.

Inference At test time, we generate action instance predictions for each query video by the classification $\boldsymbol{P}$ and mask $\boldsymbol{M}$ predictions following (Nag et al., 2022b). We aggregate the class scores in $\boldsymbol{P}$ by taking the average over all the K-shots. For each such top scoring action snippet in $\boldsymbol{P}$, we then obtain the temporal masks by thresholding t_i-th column of $\boldsymbol{M}$ using a set of thresholds $\Theta = \{\theta_i\}$. We apply SoftNMS (Bodla et al., 2017) to obtain top scoring outputs.

4 EXPERIMENTS

Datasets We evaluate two popular TAD benchmarks. (1) ActivityNet-v1.3 (Caba Heilbron et al., 2015) has 19,994 videos from 200 action classes. We follow the standard split setting of 2:1:1 for train/val/test. (2) THUMOS14 (Idrees et al., 2017) has 200 validation videos and 213 testing videos from 20 categories with labeled temporal boundary and action class.

Setting We consider two major settings. `Few-shot setting`: To facilitate fair comparison, we adopt the same dataset and class split as (Nag et al., 2021b). For both the datasets, we divide all the classes into three non-overlapping subsets for training (80%), validation (10%) and testing (10%), respectively. The validation set is used for model parameter tuning and best model selection. We

Method		N-way	Modality		ActivityNetv1.3				THUMOS14			
			Visual	Text	0.5	0.75	0.95	Avg	0.3	0.5	0.7	Avg
FS	Feat-RW	5	✗	✗	30.7	16.6	2.9	17.1	35.3	19.6	6.8	20.1
	Meta-DETR				32.9	20.3	4.6	19.4	37.5	20.7	7.5	21.9
	FSVOD				34.5	18.9	5.1	21.6	37.9	23.8	7.3	22.8
	FS-Trans	1			42.2	24.8	5.2	25.6	42.6	25.7	8.2	25.5
	QAT				44.6	26.4	4.9	26.9	38.7	24.4	7.5	24.3
MMFS	PromptDet	1	✗	✓	45.0	28.2	6.1	29.0	40.9	24.8	9.5	30.8
	Owl-Vit				43.7	27.0	6.0	27.2	38.3	21.9	9.0	30.2
	EffDet				45.9	27.9	5.2	29.4	47.2	30.4	9.8	31.1
	STALE				47.7	29.3	7.6	30.3	48.9	32.1	10.3	32.0
	Baseline-I		✓		46.9	28.6	6.9	29.7	47.3	30.5	9.2	31.8
	MUPPET				**49.7**	**32.9**	**9.2**	**32.7**	**50.6**	**33.5**	**11.2**	**33.8**
	PromptDet	5	✗		39.8	22.3	5.4	23.1	40.4	23.9	7.5	24.0
	Owl-Vit				37.9	20.3	5.6	21.9	38.3	21.9	7.7	22.6
	EffDet				41.1	21.6	5.4	23.8	39.5	23.5	7.6	24.8
	STALE				42.3	22.9	6.8	24.5	40.7	24.9	7.1	25.4
	Baseline-I		✓		42.1	22.7	6.0	24.0	40.2	24.7	7.0	25.0
	MUPPET				**45.3**	**25.6**	**6.3**	**26.2**	**42.3**	**27.2**	**7.8**	**27.5**
ZS	EffPrompt	All	✗	✓	32.0	19.3	2.9	19.6	37.2	21.6	7.2	21.9
	STALE		✗		32.1	20.7	5.9	20.5	38.3	21.2	7.0	22.2
	Baseline-I		✓		30.6	18.0	4.1	18.7	35.8	20.5	7.1	20.8
	MUPPET		✗		**33.5**	**21.9**	**6.7**	**22.0**	**40.1**	**22.8**	**8.1**	**24.8**

Table 1: Comparing our MUPPET with prior art few-shot (FS), zero-shot (ZS) and alternative methods. *Setting*: 5-shot; the CLIP model for multimodal few-shot (MMFS) methods; 50%/50% train/test class split for all ZS methods.

consider 1-way/class and 5-way settings. We consider naturally untrimmed support videos. For each N-way K-shot experiment, we divide the base and novel class video into few-shot episodes where each episode consists of $N \times (K+1)$ tasks. We train with 1000 episodes and test with 250 episodes with random tasks and report their average result. `Zero-shot setting`: In this setting, similar to few-shot, we ensure that $D_{val} \bigcap D_{test} = \phi$. We follow the setting and dataset splits used by Nag et al. (2022b) for fair comparison. For both ActivityNet and THUMOS, we train with 50% categories and test on 50% categories. To ensure statistical significance, we conduct 10 random samplings to split categories for each setting, following Ju et al. (2022). More details on splits are provided in `Appendix`.

Implementation details For fair comparison, we use CLIP (Radford et al., 2021) initialized weights for both the datasets. For comparing with CLIP based TAD baselines, we use the image and text encoders from pre-trained CLIP (ViT-B/16+Transformer) (Radford et al., 2021). We also used Kinetics (Kay et al., 2017) pre-trained initialization for showing the robustness of our approach. Video frames are pre-processed to 112×112 spatial resolution, and the maximum number of textual tokens is 77, following CLIP. Given a variable-length video, we firstly sample every 6 consecutive frames as a snippet. Then we feed the snippet into our vision encoder module, and extract the features before the fully connected layer. Thus, we obtain a set of snippet-level feature for the untrimmed video. After this, each video's feature sequence F is rescaled to $T = 100/256$ snippets for AcitivtyNet/THUMOS using linear interpolation. Our model is trained on 6 NVIDIA 3090RTX GPUs with 1000/250 episodes using Adam optimizer with learning rate of $10^{-4}/10^{-5}$ for ActivityNet/THUMOS respectively during base and meta-training. More implementation details are provided in `Appendix`

4.1 Comparison with state-of-the-art

Competitors We consider extensively three sets of previous possible methods: **(1)** ***Few-shot learning based*** methods: Two action detection methods (FS-Trans (Yang et al., 2021) and QAT (Nag et al., 2021b)). Note, FS-Trans is originally designed for spatiotemporal action detection, and we discarded the spatial detection part. Due to limited FS-TAD models, we adapt 2 object detection

baselines (Feat-RW (Kang et al., 2019), Meta-DETR (Zhang et al., 2021a)). We replaced their backbones with pre-trained frozen CLIP ViT encoders and the object decoders with TAD decoders. We similarly adapted a video based object detection method (FSVOD (Fan et al., 2021)) where temporal action proposals and temporal matching network are applied with TAD decoder. **(2)** ***Multi-modal Few-shot learning based*** methods: As this is a new problem, we need to benchmark from scratch. We adapted object detection methods (PromptDet (Feng et al., 2022), OWL-ViT (Minderer et al., 2022)). For them, we replaced the RPN by a start/end regressor as BMN (Lin et al., 2019), and the encoder and decoder as above. We also considered two state of the art TAD methods (EffPrompt (Ju et al., 2022) and STALE (Nag et al., 2022b)) by finetuning all modules with support set during inference. We further adapted CoCOOP (Zhou et al., 2022) (denoted as `Baseline-I`) based on STALE and adding the meta-network from visual branch to learn the textual tokens. This is the closest competitor of our proposed MUPPET. **(3)** ***Zero-shot learning based*** methods: EffPrompt (Ju et al., 2022) and STALE (Nag et al., 2022b) and `Baseline-I`. We deploy our MUPPET in ZS setting by discarding the few-shot specific components (*e.g.*, visual-semantics tokenizer and query regularizer). All the above methods use the same frozen CLIP for fair comparison.

Results We make several observations from the results in Table 1. **(1)** ***FS setting***: Even with 1-shot support sets, FS-TAD methods (FS-Trans (Yang et al., 2021), QAT (Nag et al., 2021b)) still outperform clearly 5-shot object detection based counterparts ((Feat-RW (Kang et al., 2019), Meta-DETR (Zhang et al., 2021a)), FSVOD (Fan et al., 2021)). This indicates the importance of modeling temporal dynamics and task specific design. **(2)** ***MMFS setting***: However, object detection methods (PromptDet (Feng et al., 2022), OWL-ViT (Minderer et al., 2022)) perform similarly as FS-TAD (EffPrompt (Ju et al., 2022), STALE (Nag et al., 2022b)) ones thanks to using text modality. Our Baseline-I yields competitive performance. Notably, MUPPET surpasses the best FS-TAD model (QAT) by a margin of 5.8%, validating the superiority of our model design and our motivation of introducing MMFS-TAD setting. Similar observation can be drawn in the 5-way case. **(3)** ***ZS setting***: Our MUPPET is superior over recent art models (EffPrompt (Ju et al., 2022), STALE (Nag et al., 2022b)) and `Baseline-I` (an integrated model even using training videos). This verifies the flexibility of our method in deployment, in addition to promising performance.

4.2 Ablation Studies

Prompt learning design We evaluate our multimodal prompt meta-learning that meta-learns the semantic information from few-shot support videos. We compare with three alternatives: (i) *Learnable Prompt from Scratch* (**LPS**): Learning the prompt from random vectors without the text encoder of ViL model (CLIP (Radford et al., 2021) in this case). (ii) *Learnable Textual Prompt* (**LTP**): Learning the prompt from randomly initialized vectors with the text encoder of ViL model. (iii) *Learnable Visual Prompt* (**LVP**): Learning the prompt from vectors initialized by visual features from the visual encoder of ViL model, as `Baseline-I`. We observe from Table 2 that: **(1)** Leveraging the pre-trained text encoder is critical due to its rich knowledge learned from vast training data, otherwise a huge result drop will occur as performed by LPS. **(2)** Learning from only few-shot support set (*i.e.*, LVP), we observe a clear mAP gap below ours, verifying the usefulness of text modality (*i.e.*, class name) and the motivation of MMFS-TAD setting. **(3)** However, using only text modality for prompt learning (*i.e.*, LTP) is even inferior than visual modality only (*i.e.*, LVP). This is not surprising as videos provide more comprehensive and finer information about new classes. This effect is illustrated in Fig 5(a,b) in `Appendix` where the visual information helps in better class-clustering. **(4)** Also, the inferiority of LTP and LVP to ours suggests that learning class-specific tokens as we design is more suitable than learning a set of global prompts shared for all classes in MMFS-TAD.

We further examine the network choices (1D CNN and set-Transformer (Lee et al., 2019)) for visual semantics tokenizer, and the necessary of class-specific prompt. As shown in Table 3, we see that: **(1)** A permutation invariant Set-Transformer is a better choice than 1-D CNN. **(2)** Using a single token per class is enough by our prompting method. This is different from previous prompting methods (Zhou et al., 2021) that instead learn multiple (*e.g.*, 20) global tokens shared by all classes.

Episodic adapters in video encoder We exploit episodic adapters for the video encoder of ViL model. Alternative methods include (i) *Freezing video encoder* without any task adaptation as STALE (Nag et al., 2022b), (ii) *Fine-tuning* the video encoder. We also compare with adapted STALE for MMFS-TAD. We observe from Table 5 that: **(1)** Fine-tuning is indeed useful as expected, as compared to the case of frozen encoder. However, it tends to overfit due to limited

Table 2: Design of prompt learning on ActivityNet. Setting: 5-way.

Design	Shots	Prompt style		mAP	
		Learnable	Context	0.5	Avg
LPS	-	✗	-	9.2	13.5
LVP	5	✓	Visual	42.1	24.0
LTP	5	✓	Text	40.3	21.5
Ours	**1**	✓	**Visual**	**43.7**	**25.1**
Ours	**5**	✓	**Visual**	**45.3**	**26.2**

Table 3: Design of visual semantics tokenizer on ActivityNet. Setting: 5-way 5-shot. #T/C: Tokens per Class.

Network	**Meta-Learn**	**#T/C**	**mAP**	
			0.5	**Avg**
1D-CNN	✗	20	37.4	21.3
	✓	20	40.8	23.0
	✓	1	39.7	22.5
SetTrans	✗	1	43.8	24.7
	✓	1	**45.3**	**26.2**
	✓	20	44.7	25.6

Table 5: Video encoder (VC) on ActivityNet. Setting: 5-way 5-shot.

Method	VC	mAP	
		0.5	Avg
MUPPET	Freeze	41.1	25.3
	Full-tuning	45.0	26.1
	Adapters	**45.3**	**26.2**

Table 6: Query feature regulation on ActivityNet. Setting: 5-way.

K**-shot**	**Query Masking**	**mAP**	
		0.5	**Avg**
-	✗	41.1	24.8
1	✓	43.7	25.1
5	✓	**45.3**	**26.2**

training samples. **(2)** Using our adapters is the best which alleviates the overfit risk by only learning a fraction of parameters.

Query feature regulation MMFS-TAD often presents large intra-class variation due to limited training video data. Our query feature regulation is designed for overcoming this challenge. As shown in Table 6, this scheme is effective with the gain increasing along with the shots of training set. This validates the usefulness of our design. For more in-depth examination, we further test the effect of *representation masking* on support video features. This is inspired by the benefits of recent Mask2Former (Cheng et al., 2022) over its previous variant MaskFormer (Cheng et al., 2021). In Table 4 we observe a gain of 1.3% in mAP@0.5. We also show that randomly initialization leads to inferior foreground prediction (see Fig. 5(c)). Support video features based initialization can improve but still not as strong as query video features (our design).

Table 4: Representation masking on support video features on ActivityNet. Setting: 5-way 5-shot.

Masking decoder	**Initialization**	**mAP**	
		0.5	**Avg**
1-D CNN	-	31.7	21.3
Maskformer	Random	38.2	24.9
Mask2Former	Random	39.5	25.2
	Support	43.7	25.8
	Query	**45.3**	**26.2**

5 Conclusions

We have presented a *multi-modality few-shot temporal action detection* (MMFS-TAD) problem to integrate the advantages of FS-TAD and ZS-TAD. To tackle MMFS-TAD, we propose a novel ***MUlti-modality PromPt mETa-learning*** (MUPPET) method, characterized by prompt meta-learning from multimodal inputs, adapters based ViL model adaptation, and query feature regulation for solving large intra-class challenge. Extensive experiments on two benchmarks show that our MUPPET surpasses both strong baselines and state-of-the-art methods under a variety of settings.

References

Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer*

vision, pp. 5561–5569, 2017.

Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. Sst: Single-stream temporal action proposals. In *CVPR*, 2017.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pp. 961–970, 2015.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *arXiv preprint arXiv:2205.13535*, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. 2020.

Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. 2021.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1290–1299, 2022.

Qi Fan, Chi-Keung Tang, and Yu-Wing Tai. Few-shot video object detection. *arXiv preprint arXiv:2104.14805*, 2021.

Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *CVPR*, pp. 3575–3584, 2019.

Chengjian Feng, Yujie Zhong, Zequn Jie, Xiangxiang Chu, Haibing Ren, Xiaolin Wei, Weidi Xie, and Lin Ma. Promptdet: Expand your detector vocabulary with uncurated images. *arXiv preprint arXiv:2203.16513*, 2022.

Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *ICCV*, 2017.

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.

Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. A simple baseline on prompt learning for efficient video understanding. 2022.

Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *ICCV*, 2019.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer. In *International Conference on Machine Learning*, 2019.

Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. BSN: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018.

Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3889–3898, 2019.

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9879–9889, 2020.

Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection with vision transformers. *arXiv preprint arXiv:2205.06230*, 2022.

Sauradip Nag, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Temporal action localization with global segmentation mask transformers. 2021a.

Sauradip Nag, Xiatian Zhu, and Tao Xiang. Few-shot temporal action localization with query adaptive transformer. *arXiv preprint arXiv:2110.10552*, 2021b.

Sauradip Nag, Xiatian Zhu, Yi-zhe Song, and Tao Xiang. Proposal-free temporal action detection via global segmentation mask learning. In *ECCV*, 2022a.

Sauradip Nag, Xiatian Zhu, Yi-zhe Song, and Tao Xiang. Zero-shot temporal action detection via vision-language prompting. In *ECCV*, 2022b.

Sauradip Nag, Xiatian Zhu, Yi-zhe Song, and Tao Xiang. Semi-supervised temporal action detection with proposal-free masking. In *ECCV*, 2022c.

Sudipta Paul, Niluthpol Chowdhury Mithun, and Amit K Roy-Chowdhury. Text-based localization of moments in a video corpus. *IEEE Transactions on Image Processing*, 30:8886–8899, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *TPAMI*, 39(6):1137–1149, 2016.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014.

Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

Deepak Sridhar, Niamul Quader, Srikanth Muralidharan, Yaoxin Li, Peng Dai, and Juwei Lu. Class semantics-based attention for action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13739–13748, 2021.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080*, 2016.

Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *CVPR*, pp. 4325–4334, 2017.

Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021.

Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *ICCV*, 2017.

Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *CVPR*, 2020a.

Mengmeng Xu, Chen Zhao, David S. Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *CVPR*, June 2020b.

Mengmeng Xu, Juan-Manuel Perez-Rua, Xiatian Zhu, Bernard Ghanem, and Brais Martinez. Low-fidelity end-to-end video encoder pre-training for temporal action localization. In *NeurIPS*, 2021.

Hongtao Yang, Xuming He, and Fatih Porikli. One-shot action localization by learning sequence matching network. In *CVPR*, 2018.

Pengwan Yang, Vincent Tao Hu, Pascal Mettes, and Cees GM Snoek. Localizing the common action among a few videos. In *ECCV*. Springer, 2020.

Pengwan Yang, Pascal Mettes, and Cees GM Snoek. Few-shot transformation of common actions into time and space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16031–16040, 2021.

Da Zhang, Xiyang Dai, and Yuan-Fang Wang. Metal: Minimum effort temporal activity localization in untrimmed videos. In *CVPR*, 2020.

Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, and Shijian Lu. Meta-detr: Few-shot object detection via unified image-level meta-learning. *arXiv preprint arXiv:2103.11731*, 2(6), 2021a.

Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021b.

Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017.

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

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16816–16825, 2022.

A APPENDIX

A.1 MORE IMPLEMENTATION DETAILS

Label Assignment To train MUPPET , we follow the same label assignment as (Nag et al., 2022b;a), where the ground-truth needs to be arranged into the designed format. Concretely, given a training video with temporal intervals and class labels (Fig. 4(a)), we label all the snippets (orange or blue squares) of a single action instance with the same action class. All the snippets off from action intervals are labeled as background. For an action snippet of a particular instance, its global action mask is defined as the video-length binary mask of that action instance. Each mask is action instance specific. All snippets of a specific action instance share the same mask. For instance, all orange snippets (Fig. 4(a)) are assigned with a T-length mask (eg. m_{24} to m_{38}) with one in the interval of $[q24, q38]$.

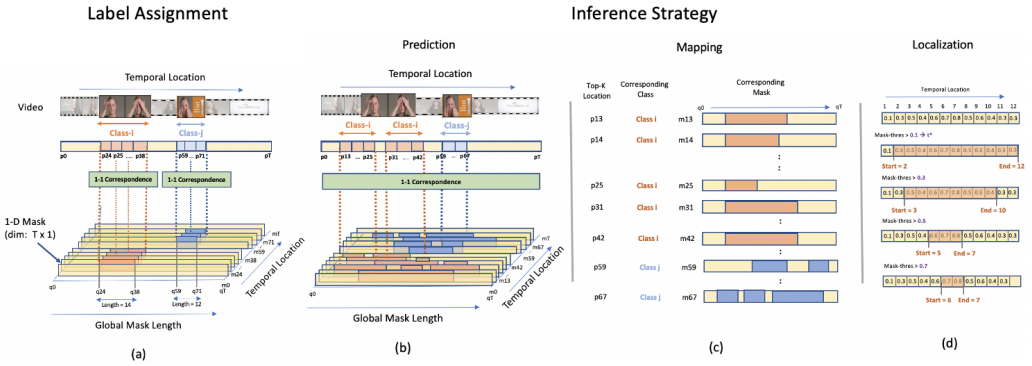

Figure 4: Example of label assignment and model inference (see text for details).

Inference Our model inference is similar as existing temporal action detection methods (Lin et al., 2019; Xu et al., 2020a; Nag et al., 2021a;b; 2022a;c). Given a test video, at each temporal scale s the action instance predictions are first generated separately based on the classification $\boldsymbol{P}^s$ and mask $\boldsymbol{M}^s$ predictions and combined for the following post-processing. Starting with the top scoring snippets from $\boldsymbol{P}$ (Fig 4(b)), we obtain their segmentation mask predictions (Fig 4(c)) by thresholding the corresponding columns of $\boldsymbol{M}$ (Fig 4(d)). To generate sufficient candidates, we apply multiple thresholds $\Theta = \{\theta_i\}$ to yield action candidates with varying lengths and confidences. For each candidate, we compute its confidence score sc_{final} by multiplying the classification score (obtained from the corresponding top-scoring snippet in P) and the segmentation mask score (i.e., the *mean* predicted foreground segment in M). Finally, we apply SoftNMS (Bodla et al., 2017) on top scoring candidates to obtain the final predictions.

A.2 MORE DETAILED ABLATION

Analysis of meta-learning In unimodal few-shot learning, we fine-tune only some task specific modules using meta-learning. However, for multi-modal few-shot setup, the natural question that can come to the mind is : *What to meta-learn ?*. To find this out, we conducted a experiment using ActivityNet dataset in a 5-way 5-shot setting. We found out that while base-training the class-agnostic localization branch generalizes well to unseen classes as seen from last Row of Table 1. However, consistent with previous finding Nag et al. (2022b), these one-stage networks are dominated by classifier performance. Hence, we meta-learn only the classifier specific modules as seen from Table A.2. We observe that, meta-learning Video Encoder , Temporal Embedding, Context Tokenizer and Mask Decoder gives the maximum performance boost. However, not meta-learning the encoder backbone and visual semantics tokenizer leads to inferior performance suggesting the importance of the individual modules towards our model design.

Ablation of temporal modeling Recall that we use a multi-head Transformer (w/o positional encoding) for temporal modeling in MUPPET. We evaluate this design choice by comparing (I) a 1D CNN with 3 dilation rates (1, 3, 5) each with 2 layers, and (II) a multi-scale Temporal Convolutional Network MS-TCN Farha & Gall (2019). Each CNN design substitutes the default Transformer while

captionAnalysis of meta-learning different blocks in 5-Way 5-Shot on ActivityNet; E: Video Encoder, T: Temporal Enc., S: Visual Semantics Tokenizer, M: Mask Decoder

E	T	S	M	mAP	
				0.5	Avg
✓	✓	✓	✓	**45.3**	**26.2**
✓	✗	✗	✓	42.2	23.5
✓	✓	✗	✓	43.8	24.7
✓	✗	✓	✓	44.7	25.3
✗	✓	✓	✓	40.8	22.9

Table 7: Analysis of temporal model design on ActivityNet in 5-shot 5-way setting. ViT-F indicates frozen ViT backbone and ViT-L indicates learnable adapter based ViT backbone

Temporal Model	Backbone	mAP	
		0.5	Avg
1D CNN	ViT-F	31.3	20.2
MST-CNN	ViT-F	35.5	22.6
Transformer	ViT-F	42.3	24.5
	ViT-L	45.3	26.2

remaining all the others. We use this transformer on top of CLIP pretrained ViT backbone encoder. Table 7 shows that the Transformer is clearly superior to both CNN alternatives. It also shows that this gain is consistent if we learn the encoder backbone with the support samples. This suggests that our default design captures stronger contextual learning capability even in low-data setting like MMFS-TAD.

Ablation with different pretraining We experiment our MUPPET with a Kinetics-400 pretraining From Table 8 we observe similar findings as that of CLIP Radford et al. (2021) pretrained features (Table 1). Our MUPPET outperforms the existing baselines by almost similar margin and better than CLIP pretraining by 4% in avg mAP, confirming that the superiority of our method is feature agnostic and succesfull in reducing the domain gap which existed between the downstream task.

Table 8: Analysis of MUPPETwith different pre-training feature on ActivityNet in 5-way 5-shot setting.

Method	Feature	mAP			
		0.5	0.75	0.95	Avg
EffPrompt	CLIP	41.1	21.6	5.4	23.8
STALE	CLIP	42.3	22.9	6.8	24.5
MUPPET	**CLIP**	**45.3**	**25.6**	**6.3**	**26.2**
MUPPET	**K-400**	**48.1**	**29.4**	**10.0**	**30.2**

A.3 Experimentation Details

A.3.1 More Implementation Details

For the vision backbones of ViL in our experiments, we use only use RGB stream unlike two-stream features like in TSN(Simonyan & Zisserman, 2014)/I3D(Carreira & Zisserman, 2017) features. The trainable parts of MUPPET are adapters in vision encoders, visual semantics tokenizer, temporal embedding module, temporal masking modules,cross-modal decoder and TAD decoder heads, whilst the text encoder is frozen. During meta-training, we freeze TAD decoder heads.

A.3.2 Few-Shot Setting

We follow the same dataset-split settings for 1-way and 5-way setting as provided by (Nag et al., 2021b).

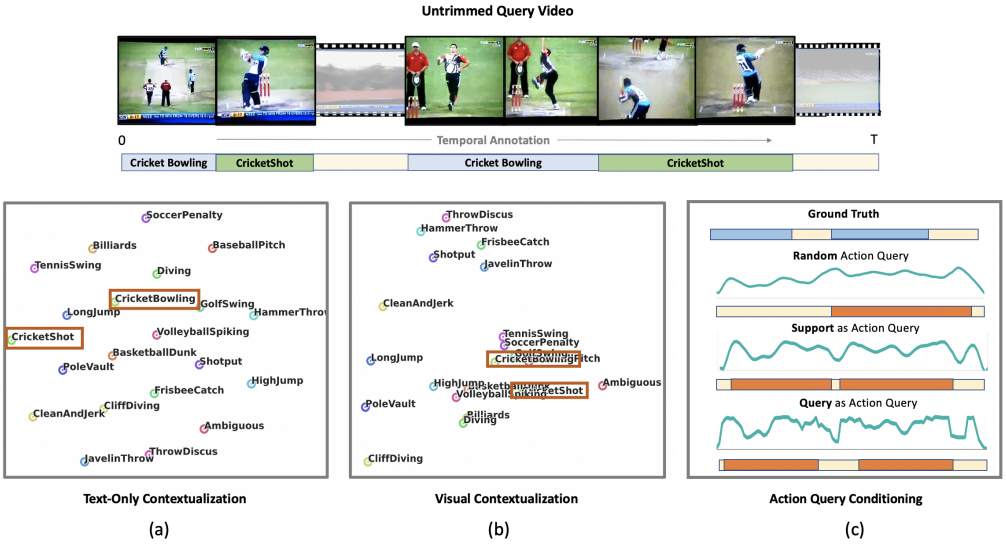

Figure 5: **Illustration of the impact of MUPPETon a random video** (a) PCA plot of our model with textual prompts (b) PCA plot of our model after incorporating visual semantics (c) Impact of various action query initialization method on actionness of representation mask.

A.3.3 Zero-Shot Setting

Here, we initiate one evaluation settings on THUMOS14 and ActivityNet1.3 in this work: train on 50% categories and test on the remaining 50% categories. The number of training and testing categories is 10 for THUMOS14 and on ActivityNet1.3, the number of both training and testing categories is 100. We follow the class splits as provided by (Nag et al., 2022b). Under each setting, we conduct 10 random samplings to split categories for training and testing. Note that, as untrimmed videos in localization are normally minutes long, splitting datasets based on action categories may incur some situations, where the same video contains both training and testing categories. For this multi-label video, we simply divide it into two videos, one containing only training categories and the other containing only testing categories.