# Open-Vocabulary Temporal Action Localization using Multimodal Guidance

Akshita Gupta[1,3]
agupta22@uoguelph.ca
Aditya Arora[2,3]
adityac8@yorku.ca
Sanath Narayan[4]
sanath.narayan@tii.ae
Salman Khan[5]
salman.khan@mbzuai.ac.ae
Fahad Shahbaz Khan[5]
fahad.khan@mbzuai.ac.ae
Graham W. Taylor[1,3]
gwtaylor@uoguelph.ca

[1] University of Guelph, Guelph, Ontario
[2] York University, Toronto, Ontario
[3] Vector Institute, Toronto, Ontario
[4] Technology Innovation Institute, Abu Dhabi, UAE
[5] Mohamed Bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE

**Abstract**

Open-Vocabulary Temporal Action Localization (OVTAL) enables a model to recognize any desired action category in videos without the need to explicitly curate training data for all categories. However, this flexibility poses significant challenges, as the model must recognize not only the action categories seen during training but also novel categories specified at inference. Unlike standard temporal action localization, where training and test categories are predetermined, OVTAL requires understanding contextual cues that reveal the semantics of novel categories. To address these challenges, we introduce OVFormer, a novel open-vocabulary framework extending ActionFormer with three key contributions. First, we employ task-specific prompts as input to a large language model to obtain rich class-specific descriptions for action categories. Second, we introduce a cross-attention mechanism to learn the alignment between class representations and frame-level video features, facilitating the multimodal guided features. Third, we propose a two-stage training strategy which includes training with a larger vocabulary dataset and finetuning to downstream data to generalize to novel categories. OVFormer extends existing TAL methods to open-vocabulary settings. Comprehensive evaluations on the THUMOS14 and ActivityNet-1.3 benchmarks demonstrate the effectiveness of our method. Our code is available at https://github.com/adityac8/OVFormer.

## 1 Introduction

Temporal action localization (TAL) aims to localize and classify every action instance in a long untrimmed video. This task is crucial for tasks such as video understanding, surveillance and summarizing videos. In recent years, numerous methods have emerged to address

TAL [6, 19, 24, 42], achieving significant performance at localizing and recognizing a fixed set of action categories. However, most works are restricted to a closed-set setting. To localize novel action categories unseen during training, these approaches require training the model on the combined set of base and novel categories using additional annotated instances from the novel classes under consideration. With the increasing volume of videos, annotating every action instance in videos is impractical. In this work, we relax the restriction of localizing closed-set action classes in the TAL setting and propose an Open-Vocabulary TAL (OVTAL) approach, called OVFormer. Our OVFormer strives to localize both base actions defined during training as well as novel action classes during inference.

Predicting novel classes during inference poses a significantly greater challenge compared to standard TAL or its closely related problems such as the open-set [2, 7], zero-shot [16, 27, 29, 31], and few-shot [18, 26, 36] settings. While open-set approaches typically assign an "unknown" label to novel action categories, zero-shot methods rely on a text encoder's ability to provide meaningful representations based on the class name. However, the latter approaches have a tendency to overfit and are likely to be biased towards base categories. Recent work [31] finetunes CLIP [32], which comprises a vision and text encoder for encoding images and corresponding text labels. Although finetuning CLIP's text encoder helps bridge the domain gap between the videos and text in the downstream task, it comes at the cost of losing the generalization learned between the CLIP visual and text encoders. This is because only the text encoder is finetuned with fixed prompts involving only the class names for the downstream task. In contrast, we propose to encode rich class-specific language descriptions (extracted from an LLM) using the CLIP text encoder and utilize them as guidance features for learning the visual cues and semantic context related to novel action categories. Overall, our approach harnesses the power of LLMs and the internal representation of the CLIP text encoder to provide rich and informative descriptions for novel action categories.

Language descriptions enable the ability to clearly distinguish between closely related actions having similar visual cues. For example, `javelin throw` and `pole vault` actions have visual similarities such as `sports fields`, `equipment`, and body motion such as `running`, `jumping` and `throwing`. To leverage these descriptions for localizing actions, we propose to learn multimodal guided features by first cross-attending the language descriptions with frame-level (spatial) features. These guided features are then fused with snippet-level (spatio-temporal) features to achieve multimodal snippet-level features. Such a progressive integration of language descriptions to spatio-temporal features through the spatial features achieves a better alignment between textual embeddings and visual action features. This alignment aids in correctly localizing the novel actions based on their descriptions during inference. Furthermore, we employ a two-stage training pipeline, in which we first train our proposed OVFormer on a larger vocabulary dataset, followed by finetuning it on the downstream data to adapt to its characteristics.

To the best of our knowledge, this is the first work on OVTAL. We formulate a simple but strong solution by leveraging LLMs and crafting task-specific prompts as input to generate class-specific language descriptions. We introduce the modality mixer module for fusing class-specific language descriptions with frame-level features to yield multimodal guided features. These features help learn the mapping between text embeddings and the visual cues related to the action. When fused with snippet-level features, this mapping is transferred to recognize novel action categories. We conduct extensive experiments on two popular benchmarks and significantly outperform existing SOTA approaches on THUMOS14 [14] and ActivityNet-1.3 [12] for both OVTAL and ZSTAL tasks.

## 2 Related Work

**Temporal Action Localization (TAL):** Existing TAL methods fall into two categories: two-stage approaches, which involve proposal generation followed by classification (based on anchor windows [3, 9, 13], action boundaries [10, 20, 21, 25, 43], graphs [1, 37], or transformers [5, 33, 35]), and single-stage approaches [19, 40], which are anchor-free and trained end-to-end. However, a key limitation of all current TAL methods is their closed-world assumption — they require the same action categories, ranging from around 20 to 200, to be present both during training and inference, preventing generalization to novel action categories unseen during training.

**Zero-Shot Temporal Action Localization (ZSTAL):** To address this limitation, ZSTAL aims to localize and recognize novel action categories in untrimmed videos unseen during training. Traditional zero-shot learning approaches transfer knowledge from "seen" to "unseen" classes through shared semantic embeddings or vision-language alignments. Prior works are classified into semantic embedding-based approaches such as ZSTAD [41], TranZAD [28], and vision-language model-based approaches such as Efficient-Prompt [16], STALE [27], and ZEETAD [31]. However, zero-shot methods still fall short of real-world applications, specifically because of the constraint of identifying "unseen" categories without prior knowledge and relying solely on the base categories. Building upon the limitations of TAL and ZSTAL, we introduce OVTAL, which lifts the restriction of defining "unseen" categories *a priori*.

**Prompt-based techniques:** Prompting refers to designing an instruction which, when passed through the pretrained language model, can guide the downstream task. Prompt-based learning techniques have been widely used in the NLP domain [15, 23]. CLIP [32] introduces prompt-based learning in image recognition tasks, where it shows learning relationships between vision-language models using large-scale image-text pairs. Methods like [34, 46, 47] introduced learnable vectors to the text encoder of CLIP for transfer learning to recognition tasks. We use action description-based prompting in this work to enable the localization of novel action classes in the open-vocabulary setting.

In summary, while previous works like Efficient-Prompt, STALE, and ZEETAD explore low-shot temporal action localization, to the best of our knowledge, our work is the first to investigate the open-vocabulary setting. Our proposed approach leverages pretraining on a larger localization vocabulary dataset, fusing visual features with text descriptions from a language model to obtain rich multimodal representations. This enables the model to capture visual cues and semantic context related to the actions, leading to improved performance on both base and novel actions.

## 3 Open-Vocabulary Temporal Action localization

**Problem Formulation:** Given an input video $X$, frame-level features are denoted by $X_F = \{x_f^1, x_f^2, \cdots, x_f^T\}$ and snippet-level features by $X_V = \{x_v^1, x_v^2, \cdots, x_v^T\}$ over time $t = \{1, 2, \cdots, T\}$. Here, $T$ denotes the total duration of the video. When the feature vectors $\{x^t\}_{t=1}^T$ are fed as input to the OVTAL method, the method is expected to predict action labels $Y = \{y_1, y_2, \cdots, y_N\}$, where $N$ is the number of action instances. Each instance $y_i = \{s_i, e_i, a_i\}$ is defined by a start time $s_i$, end time $e_i$, and action label $a_i$, where $s_i \in [1, T]$, $e_i \in (s_i, T]$, and $a_i \in \{1, \cdots, A\}$, where $A$ is the number of action categories (elaborated on below). Taking inspiration from [17, 48], two datasets are used during training: a large vocabulary-dense annotation dataset $\mathcal{V}_{super}$ with vocabulary $\mathcal{A}_{super}$, and a smaller dataset $\mathcal{V}_{base}$ with vocabulary

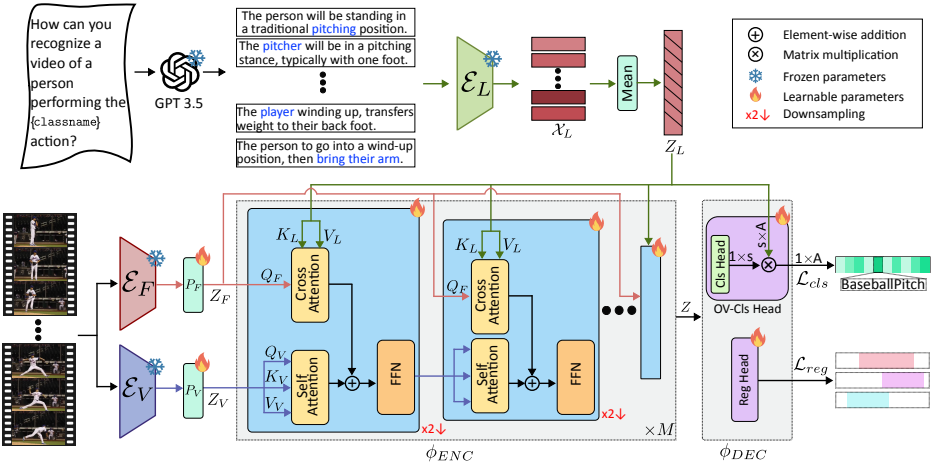


Figure 1: **Overview of OVFormer.** Given a long untrimmed video $X$, frame-level features are extracted using $\mathcal{E}_F$ (DINOv2[31]) and snippet-level features using $\mathcal{E}_V$ (two-stream I3D video encoder[2]). These features are projected into $D$-dimensional feature spaces $Z_F$ and $Z_V$ using the projection functions $P_F$ and $P_V$. $\mathcal{E}_L$, which is the CLIP ViT-B/32 text encoder[32], is used to generate text embeddings $Z_L$. These features are then passed as input to the multi-scale $\phi_{ENC}$ module, which includes our proposed modality mixer. The modality mixer takes $Z_F$ and $Z_V$ as input, where $Z_V$ undergoes self-attention, and $Z_F$ is cross-attended with text embeddings $Z_L$. The resulting multimodal guided features are fused with the self-attended $Z_V$. The output of $\phi_{ENC}$, enriched multimodal snippet-level features $Z$, is used as input for $\phi_{DEC}$, which consists of OV-classification and regression heads. The OV-classification head maps the enriched multimodal snippet-level features to the semantic space, relating them to class semantics and obtaining action candidates. During inference, text embeddings of novel categories are used to enable the OV capability.

$\mathcal{A}_{base}$. During inference, we use a testing split $\mathcal{V}_{novel}$ with vocabulary $\mathcal{A}_{novel}$ that shares the same data structure as $\mathcal{V}_{base}$. To identify novel categories, text embeddings $Z_L$ are introduced into the training pipeline as input to $\phi_{ENC}$ and $\phi_{DEC}$ to the OV-classification head. In the most general case, there are no restrictions on the overlap or lack thereof between the sets $\mathcal{A}_{super}$, $\mathcal{A}_{base}$, and $\mathcal{A}_{novel}$. In OVTAL, a deep network $f(\cdot)$ is trained to identify novel action categories. The network is the composition of two modules $f = \phi_{DEC} \circ \phi_{ENC}$.

The encoder $\phi_{ENC}(X_V, X_F, Z_L)$ yields multi-scale representations $Z = \{Z^1, Z^2, ..., Z^M\}$, where $Z \in \mathbb{R}^{2^{m-1}T \times D}$ and $m = 1 \cdots M$. $Z$ are then passed through the decoder $\phi_{DEC}(\{Z^j\}_{j=1}^{K})$, which yields predicted labels $\hat{Y} = \{\hat{y}^1, \hat{y}^2, ..., \hat{y}^T\}$. In stage I, $f(\cdot)$ is trained on $\mathcal{V}super$ to learn from a larger vocabulary $\mathcal{A}super$ along with $Z_L$ class-specific language descriptions. This is followed by stage II training, where $f(\cdot)$, previously trained on $\mathcal{V}super$, is finetuned on $\mathcal{V}Base$ to adapt to the dataset characteristics of base action categories, resulting in improved performance. Our goal for $f(\cdot)$ is to predict any action category from the combined set $\mathcal{A} = \mathcal{A}_{base} \cup \mathcal{A}_{novel}$ during inference. The proposed OVFormer aims to generalize effectively to novel action categories while maintaining high performance on base categories.

## 3.1 Overall Architecture

As previously discussed, localizing and recognizing novel action categories while remembering the base action categories is a challenging task. Figure 1 shows the overall archi-

tecture of the proposed OVTAL method. OVFormer adapts the popular ActionFormer [40] as its base architecture and introduces (i) class-specific language descriptions (subsubsection 3.2.1) from an LLM to classify and localize novel action categories; and (ii) a modality mixer (subsubsection 3.2.2) for learning the scene information and semantic context by cross-attending aggregated text embeddings $Z_L$ and the frame-level features $Z_F$. Furthermore, by introducing $Z_L$ into the training pipeline, we are able to separate foreground action regions from the background and emphasize the visual cues and semantic context related to the actions. In the proposed OVFormer, an input video $X$ is fed into modality-specific off-the-shelf encoders (video and visual) to obtain snippet- ($X_V$) and frame-level features ($X_F$). These features are then passed through the projection functions $P_V$ and $P_F$ which embed them into $D$-dimensional space, $Z_V \in \mathbb{R}^{T\times D}$, and $Z_F \in \mathbb{R}^{T\times \hat{D}}$, respectively. Both of these are input to $\phi_{ENC}(\cdot)$ along with class-specific text embeddings $Z_L$. Here, $T$ is the temporal length, $D$ is the dimension of the feature vector for each snippet, and $\hat{D}$ is the dimension of the feature vector for each frame. $\phi_{ENC}$ captures multi-scale feature representations for frame-level and snippet-level features, i.e., $Z_F \in \mathbb{R}^{2^{m-1}T\times \hat{D}}$ and $Z_V \in \mathbb{R}^{2^{m-1}T\times D}$, where $m = 1\cdots M$. These multi-scale representations, along with the class-specific text embeddings $Z_L \in \mathbb{R}^{s\times A}$, where $s$ is the text embedding dimension for each class, are fed into the modality mixer. The output from $\phi_{ENC}(\cdot)$ results in an enriched multimodal snippet-level features representation $Z \in \mathbb{R}^{2^{m-1}T\times D}$. The enriched features are then fed to $\phi_{DEC}(\cdot)$, which consists of OV-classification and regression heads. The OV-classification head feature space is mapped to the class-specific text embeddings $Z_L \in \mathbb{R}^{s\times A}$ to relate to the class semantics. Overall, our proposed OVFormer is trained end-to-end using dedicated classification ($\mathcal{L}_{cls}$) and regression ($\mathcal{L}_{reg}$) loss terms. Next, we present the OVFormer approach in detail.

## 3.2 OVFormer

### 3.2.1 Class-Specific Language Descriptions

Existing approaches, such as Efficient-Prompt [16], make use of simple prompts like "A video of `{classname}`" or "`{classname}`". These methods rely on the strength of the text encoder to understand class attributes and information related to the class solely from the class name. However, such prompts are unable to highlight the important attributes and semantic context responsible for defining the action. This capability is crucial for localization and classification, as it helps to understand the scenes and background context for the action.

To this end, we leverage a pretrained language model, specifically the `GPT-3.5-turbo-instruct` model from OpenAI. We generate 10 detailed descriptions per class (Figure 1 shows four descriptions for clarity, with more examples in Supplementary). For generating rich, detailed descriptions of the class by LLM, we pass a prompt: "How can you recognize a video of a person performing the `{classname}` action?" Given a set of $E$ language descriptions $s_r^a$ for a predefined category $a$, we encode each description using the CLIP text encoder [32], and obtain an aggregated embedding for the action category $a$ as:

$$Z_L = \frac{1}{E}\sum_{r=1}^{E} \mathcal{E}_L(s_r^a). \tag{1}$$

Using the aggregated embedding helps capture the semantics of the class while mitigating biases from individual descriptions. These embeddings $Z_L$ are used as input to the modality mixer and $\phi_{DEC}(\cdot)$ (as shown in Figure 1).

This simple technique of aggregation can summarize the class-wise description very well. During testing, $Z_L$ for novel actions are computed in the same way by passing novel categories as classnames to enable the OVTAL setting.

### 3.2.2 Modality Mixer

A naïve approach for converting a fully-supervised TAL model to an OVTAL model is to simply multiply the classifier output features with textual features. However, such an approach is insufficient to handle novel action categories effectively since a late fusion of the two modalities likely results in the encoder learning less discriminative action features that are not well-aligned with the textual embeddings. Here, we strive to develop a more robust contextualization method for accurately detecting actions in untrimmed videos within an OVTAL setting. To this end, we introduce a modality mixer, a simple yet effective approach that enhances the snippet-level features $Z_V$ using textual embeddings $Z_L$ by capturing long-range temporal dependencies between the visual features and aligning them to the corresponding textual embeddings in a progressive manner, resulting in enriched multimodal snippet-level features $Z$.

Capturing long-range temporal dependencies is crucial in OVTAL, as actions may span across multiple time steps, and the context surrounding an action is likely to provide valuable information for accurate recognition and localization. Thus, our modality mixer first focuses on learning the temporal context across the full sequence. Here, the features $X_V$ are projected into $Z_V$ using a convolutional network consisting of two $1 \times 1$ convolution layers with ReLU, where $Z_V \in \mathbb{R}^{T \times D}$ with $T$ time steps and $D$ dimensional features. These features are projected into a low-dimensional space for creating query, key, and value tensors given by $Q_V^h = Z_V W_Q^h$, $K_V^h = Z_V W_K^h$ and $V_V^h = Z_V W_V^h$, which self-attend to result in enriched features $Z_V^{'}$ given by

$$Z_V^{'} = \left[\alpha^1; \alpha^2; \ldots; \alpha^H\right] W_o, \quad \text{where} \quad \alpha^h = A^h V^h \quad \text{with} \quad A^h = \sigma\left(\frac{Q_V^h (K_V^h)^T}{\sqrt{D_k}}\right). \tag{2}$$

Here, $h \in \{1, 2, \ldots, H\}$, and $W_Q^h, W_K^h, W_V^h, W_o$ are learnable parameters.

Consequently, the enriched snippet-level features $Z_V^{'}$ can encode the long temporal context. Furthermore, we propose to enhance the alignment between the text embeddings $Z_L$ and the snippet-level features $Z_V$ well before the classification stage in a progressive manner. First, we align the frame-level features $Z_F$ with $Z_L$ through cross-attention and then fuse the resulting features with the enriched snippet-level features. Such a text $\rightarrow$ image (frame-level) $\rightarrow$ video (snippet-level) progressive integration aids in better aligning the visual features to the corresponding textual embeddings. The fused features are then passed through a feed-forward network.

The query, key, and value tensors $Q_F^h = Z_F \hat{W}_Q^h$, $K_L^h = Z_L \hat{W}_K^h$ and $V_L^h = Z_L \hat{W}_V^h$ are used to obtain multimodal guided features $Z_F^{'}$, similar to Equation 2. Furthermore, the enriched multimodal snippet-level features are computed as

$$Z = FFN(Z_F^{'} + Z_V^{'}) \tag{3}$$

By embedding class-specific language descriptions within the training pipeline at an earlier stage, we ensure that the snippet-level features are more closely aligned with the textual descriptions by the time they reach the classifier. This early fusion of modalities enables our model to effectively recognize and localize novel action categories in untrimmed videos.

| Method | THUMOS14 | ActivityNet-1.3 |
|---|---|---|
| P-ActionFormer | 0.2 | 0.1 |
| OVFormer (Ours) | **12.6** | **19.0** |

Table 1: Average performance ($mAP_{all}$) of P-ActionFormer [Figure 2(a)] and OVFormer, both trained in Stage I and tested on THUMOS14 and ActivityNet-1.3 over all classes.

Table 2: **OVTAL results on THUMOS14 and ActivityNet-1.3.** Average performance (mAP) over [0.3:0.1:0.7] for THUMOS14 and [0.5:0.05:0.95] for ActivityNet-1.3. Our proposed method, OVFormer, achieves significant gains in mAP over base, novel, and all action categories for both 75-25 and 50-50 splits. For a fair comparison, we evaluate STALE† and obtain results for base, novel, and all action categories. See subsection 4.1 for more details.

| Train-Test split | Method | THUMOS14 | | | ActivityNet-1.3 | | |
|---|---|---|---|---|---|---|---|
| | | $mAP_{base}$ | $mAP_{novel}$ | $mAP_{all}$ | $mAP_{base}$ | $mAP_{novel}$ | $mAP_{all}$ |
| 75% Seen 25% Unseen | ActionFormer [40] | 65.1 | - | - | 31.0 | - | - |
| | P-ActionFormer | 51.9 | 13.8 | 41.5 | 30.0 | 15.3 | 26.3 |
| | L-ActionFormer | 52.3 | 14.7 | 42.8 | 30.9 | 16.8 | 27.3 |
| | F-ActionFormer | 50.8 | 24.2 | 44.1 | 30.8 | 22.9 | 28.8 |
| | STALE† [27] | - | - | - | 23.2 | 20.6 | 22.6 |
| | OVFormer (ours) | **56.4** | **27.3** | **49.1** | **31.4** | **25.1** | **29.8** |
| 50% Seen 50% Unseen | ActionFormer [40] | 63.1 | - | - | 28.6 | - | - |
| | P-ActionFormer | 50.9 | 9.9 | 30.5 | 27.6 | 13.0 | 20.3 |
| | L-ActionFormer | 48.3 | 10.1 | 29.2 | 28.3 | 13.5 | 20.9 |
| | F-ActionFormer | 51.2 | 20.5 | 35.8 | 28.8 | 23.5 | 26.2 |
| | STALE† [27] | - | - | - | 23.0 | 20.7 | 22.2 |
| | OVFormer (ours) | **55.7** | **24.9** | **40.7** | **30.2** | **24.8** | **27.5** |

### 3.3 Training and Inference

Our proposed OVFormer is trained end-to-end using the following joint loss formulation:

$$\mathcal{L} = (\mathcal{L}_{cls} + \lambda \mathcal{L}_{reg}) \tag{4}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$ denote the loss terms for the OV-classification and regression heads, respectively. For $\mathcal{L}_{cls}$, we employ the standard focal loss [22] for $A$-way binary classification, while for $\mathcal{L}_{reg}$, we utilize the standard DIoU loss [44] for regression. The weighting factor $\lambda_{reg}$ is set to a default value of 1. At inference time, the novel action categories are passed as classnames to the prompt, which leads to $A_{novel}$ predictions from the OV-classification head, followed by predicted regression ranges from the regression head.

## 4 Experiments

We evaluate OVFormer on two datasets: THUMOS14 [14] and ActivityNet-1.3 [12]. Following other open-vocabulary [11, 38, 45] and TAL methods [8, 21, 39, 40], we report mean average precision over base ($mAP_{base}$), novel ($mAP_{novel}$), and all ($mAP_{all}$) action categories. Snippet- and frame-level features are extracted using a two-stream I3D video encoder [4] and DINOv2 [30] respectively for HACS, THUMOS14 and ActivityNet-1.3. Additional details on the experimental setup are provided in the supplementary material.

### 4.1 Results

As this is the first exploration of Open-Vocabulary in TAL, we study three baselines based on our OVFormer: P-ActionFormer, L-ActionFormer, and F-ActionFormer (Figure 2(a)-(c), respectively) and compare their performances to that of our OVFormer model (Figure 2(d)).
**Pretraining generalization:** In Table 1, OVFormer and P-ActionFormer models with Stage I training alone are directly evaluated on THUMOS14 and ActivityNet-1.3, illustrating the

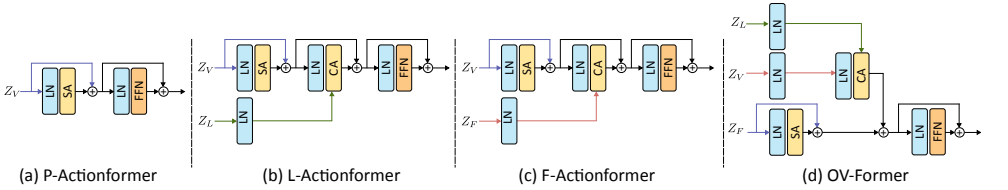


Figure 2: Design choices for the modality mixer which are used as baselines for the OVTAL setting and evaluated in Table 2. From (a-d) the text embeddings $Z_L$ are introduced in the OV-classification head (a) Naïve solution where only snippet-level features. (b) Introduce text embeddings and cross-attend with the snippet-level features. (c) A variation on (b) where frame-level features are cross-attended with snippet-level features. (d) Our proposed method cross-attends text embeddings with frame-level features to learn multimodal guided features, which is fused with snippet-level features.

Table 3: **State-of-the-art comparison for ZSTAL on THUMOS14 and ActivityNet-1.3.** We show the comparison in terms of mAP evaluated over novel action categories and IoU thresholds of [0.3:0.1:0.7] for THUMOS14 and [0.5:0.05:0.95] for ActivityNet-1.3. Our OVFormer achieved significant gains in mAP in comparison to existing approaches. We only include the methods with open-source code available. subsection 4.1 for more details.

| Train-Test split | Method | THUMOS14 | | | | | | ActivityNet-1.3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | mAP | 0.5 | 0.75 | 0.95 | mAP |
| 75% Seen 25% Unseen | B-II [27] | 28.5 | 20.3 | 17.1 | 10.5 | 6.9 | 16.6 | 32.6 | 18.5 | 5.8 | 19.6 |
| | B-I [27] | 33.0 | 25.5 | 18.3 | 11.6 | 5.7 | 18.8 | 35.6 | 20.4 | 2.1 | 20.2 |
| | Eff-Prompt [18] | 39.7 | 31.6 | 23.0 | 14.9 | 7.5 | 23.3 | 37.6 | 22.9 | 3.8 | 23.1 |
| | STALE [27] | 40.5 | 32.3 | 23.5 | 15.3 | 7.6 | 23.8 | 38.2 | 25.2 | 6.0 | 24.9 |
| | OVFormer (ours) | **49.8** | **43.8** | **35.8** | **27.8** | **19.2** | **35.3** ↑11.5 | **46.7** | **29.4** | **6.1** | **29.5** ↑4.6 |
| 50% Seen 50% Unseen | B-II [27] | 21.0 | 16.4 | 11.2 | 6.3 | 3.2 | 11.6 | 25.3 | 13.0 | 3.7 | 12.9 |
| | B-I [27] | 27.2 | 21.3 | 15.3 | 9.7 | 4.8 | 15.7 | 28.0 | 16.4 | 1.2 | 16.0 |
| | Eff-Prompt [18] | 37.2 | 29.6 | 21.6 | 14.0 | 7.2 | 21.9 | 32.0 | 19.3 | 2.9 | 19.6 |
| | STALE [27] | 38.3 | 30.7 | 21.2 | 13.8 | 7.0 | 22.2 | 32.1 | 20.7 | 5.9 | 20.5 |
| | OVFormer (ours) | **42.8** | **37.3** | **30.6** | **23.5** | **15.9** | **30.5** ↑8.3 | **42.8** | **27.3** | **6.0** | **27.2** ↑6.7 |

outcomes (i) when only Stage I is used without Stage II and (ii) the effect of fusing text embeddings at the classifier. The baseline (P-ActionFormer), which introduces text embeddings *only* in the OV-classification head, performs poorly on novel action categories (0.2% mAP on THUMOS14, 0.1% on ActivityNet-1.3). This indicates that late fusion of text embeddings is insufficient to localize and recognize novel action categories and Stage I alone is insufficient to bridge the gap between datasets with different characteristics. In contrast, our proposed method introduces text embeddings in the training pipeline and fuses them with snippet-level features, focusing on learning scene information and semantic context. This helps to separate foreground and background objects, leading to improved generalization performance on novel categories (12.6% mAP on THUMOS14, 19.0% on ActivityNet-1.3).

**Performance on OVTAL:** Table 2 shows the state-of-the-art performance on the OVTAL task. We report results for our proposed OVFormer as well as the standard ActionFormer [40] for comparison. Since ActionFormer can only localize and recognize base action categories, it is not directly applicable to OVTAL, and its $mAP_{novel}$ cannot be computed. For a fair comparison with an existing ZSTAL approach, we extended STALE$^{\dagger}$ [27] to get $mAP_{base}$, $mAP_{novel}$, and $mAP_{all}$ scores. STALE$^{\dagger}$ achieves 23.2%, 20.6%, and 22.6% for base, novel, and all categories, respectively. Our OVFormer significantly outperforms STALE, achieving 31.4%, 25.1%, and 29.8% for the same categories. The consistent performance gains

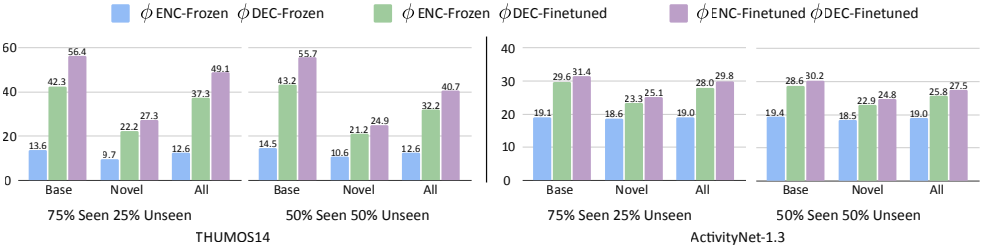


Figure 3: **Finetuning strategies** by freezing or finetuning the $\phi_{ENC}$/$\phi_{DEC}$ on OVTAL setting. Here, for showing the effectiveness of Stage II, Stage I of the training pipeline is always present.

across both the THUMOS14 and ActivityNet-1.3 datasets highlight the effectiveness of our proposed contributions for the OVTAL task.

**Comparison of ZSTAL Methods:** We present a performance comparison for the ZSTAL task in Table 3. We compared our method only with those that have available open-source implementations. Our OVFormer achieves significant improvements on both the THUMOS14 and ActivityNet-1.3. Following [16, 27], the evaluation is performed by considering only novel action categories during inference for the 75-25 and 50-50 splits. OVFormer outperforms existing ZSTAL methods by a substantial margin, illustrating the benefits of learning on a large vocabulary dataset and effectively modelling rich scene information.

## 4.2 Ablation Study

Figure 3 shows different finetuning strategies for Stage II on downstream data, where we observed that finetuning both $\phi_{ENC}$ and $\phi_{DEC}$ in our proposed method helps maintain overall performance while mitigating performance degradation on novel action categories.

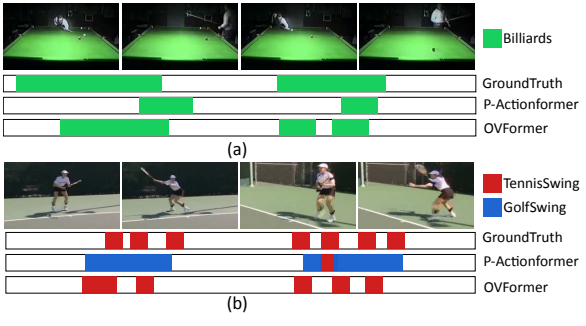


Figure 4: OVFormer performance on THUMOS14 in OVTAL. We compare the performance of P-ActionFormer (Figure 2(a)) and OVFormer (Figure 2(d)) on (a) the `billiards` action, and (b) the `tennis swing` and `golf swing` actions.

Figure 4 demonstrates the superior capabilities of OVFormer over P-ActionFormer. Our model predictions closely align with the ground truth, particularly in `billiards` and `tennis swing`. We examine the performance on Figure 4(a) `billiards`, and Figure 4(b) `tennis swing` and `golf swing` actions. In Figure 4(b), `tennis swing` belongs to the base classes, while `golf swing` belongs to the novel classes. In the case of P-ActionFormer, confusion exists between these actions, as they both have similar visual cues, *i.e.*, a person running with an object in their hands. OVFormer improves scene information and semantic context by obtaining multimodal guided features and fusing them with snippet-level features, enhancing the separation between base and novel actions.

**Effect of different prompt templates:** Table 4 shows the OVFormer performance on manually crafted prompts and our class-specific generated descriptions from an LLM. Here, we

Table 4: **Effect of different prompt templates on OVTAL setting for OVFormer.** Using our rich LLM-generated class-specific language descriptions during training to obtain multimodal guided features for the snippet-level features improves the $mAP_{novel}$ performance compared to manually crafted prompts.

| Split | Prompt | THUMOS14 | | | ActivityNet-1.3 | | |
|---|---|---|---|---|---|---|---|
| | | $mAP_{base}$ | $mAP_{novel}$ | $mAP_{all}$ | $mAP_{base}$ | $mAP_{novel}$ | $mAP_{all}$ |
| 75% Seen 25% Unseen | `{classname}` | 59.3 | 8.0 | 46.3 | 28.6 | 8.1 | 23.6 |
| | A video of `{classname}` | 59.2 | 8.5 | 46.5 | 28.4 | 6.1 | 22.8 |
| | Ours: LLM generated descriptions | 59.0 | **10.2** | 46.8 | 28.7 | **9.5** | 23.9 |
| 50% Seen 50% Unseen | `{classname}` | 59.0 | 6.1 | 32.4 | 26.2 | 5.1 | 15.8 |
| | A video of `{classname}` | 58.9 | 7.0 | 32.8 | 25.9 | 4.3 | 15.1 |
| | Ours: LLM generated descriptions | 58.4 | **7.7** | 33.1 | 26.2 | **6.8** | 16.5 |

| Train-Test Split | Visual Encoder ($\mathcal{E}_{\mathcal{F}}$) | THUMOS14 | | |
|---|---|---|---|---|
| | | $mAP_{base}$ | $mAP_{novel}$ | $mAP_{all}$ |
| 75% Seen 25% Unseen | CLIP | 50.5 | 21.1 | 43.2 |
| | DINOv2 | **56.4** | **27.3** | **49.1** |
| 50% Seen 50% Unseen | CLIP | 50.3 | 17.1 | 33.7 |
| | DINOv2 | **55.7** | **24.9** | **40.7** |

Table 5: **OVTAL results on THUMOS14**. Average performance (mAP) over [0.3:0.1:0.7] for THUMOS14 and [0.5:0.05:0.95] for ActivityNet-1.3. OVFormer using DINOv2 visual encoder $\mathcal{E}_{\mathcal{F}}$ for frame-level features $X_F$ achieves significant improvement over CLIP.

demonstrate the performance using only Stage II of the training pipeline, without using additional data. We observe that using the simplest prompts, "`{classname}`" and "A video of `{classname}`", achieves comparable performance to LLM-generated prompts for $mAP_{base}$ and $mAP_{all}$ but lower performance on $mAP_{novel}$. This demonstrates the importance of capturing the attributes and scene information surrounding the action. Using our proposed generated descriptions, we achieve improvement of 2.2%, 1.4%, 1.6% and 1.7% over 75-25 and 50-50 splits, respectively, for $mAP_{novel}$ compared to manually crafted prompts.

**Effect of Frame-Level Features:** Table 5 presents the performance of our proposed method, OVFormer, using CLIP [32] and DINOv2 [30] visual encoders $\mathcal{E}_{\mathcal{F}}$ for extracting frame-level features $X_F$. We observe that off-the-shelf DINOv2 features significantly outperform CLIP features, with absolute gains of 5.9%, 6.2%, and 5.9% over the 75-25 split for base, novel, and all action categories, respectively. Similarly, on the 50-50 split, DINOv2 achieves improvements of 5.4%, 7.8%, and 7.0% over CLIP for the same categories. These results are consistent with the findings reported in [30], where DINOv2 is shown to capture richer visual descriptions compared to CLIP. This is particularly important for our problem statement, which focuses on body movements for related actions, as DINOv2's ability to capture richer visual descriptions helps in accurately distinguishing subtle differences in these movements. In this setup, both Stage I and Stage II of our method are utilized.

# 5 Conclusions

In this work, we introduced Open-Vocabulary Temporal Action Localization, a novel and challenging task that aims to localize and recognize both base and novel action classes in untrimmed videos. To address this task, we proposed OVFormer, a framework that leverages multimodal guided features to enrich snippet-level features. Our two-stage training strategy, which includes pretraining on a larger vocabulary dataset and finetuning on the downstream data, enables OVFormer to achieve state-of-the-art performance on both THUMOS14 and ActivityNet-1.3. The proposed approach significantly outperforms existing methods in both the OVTAL and ZSTAL settings, demonstrating its effectiveness in recognizing and localizing novel action categories while maintaining high performance on base categories.

# 6 Acknowledgment

Resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and partners of the Vector Institute. GWT acknowledges support from NSERC.

# References

[1] Y. Bai, Y. Wang, Y. Tong, Y. Yang, Q. Liu, and J. Liu. Boundary content graph neural network for temporal action proposal generation. In *ECCV*, 2020.

[2] W. Bao, Q. Yu, and Y. Kong. Opental: Towards open set temporal action localization. In *CVPR*, 2022.

[3] S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. C. Niebles. SST: Single-stream temporal action proposals. In *CVPR*, 2017.

[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017.

[5] S. Chang, P. Wang, F. Wang, H. Li, and Z. Shou. Augmented transformer with adaptive graph for temporal action proposal generation. In *MM Workshops*, 2022.

[6] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar. Rethinking the faster R-CNN architecture for temporal action localization. In *CVPR*, 2018.

[7] M. Chen, J. Gao, and C. Xu. Cascade evidential learning for open-world weakly-supervised temporal action localization. In *CVPR*, 2023.

[8] F. Cheng and G. Bertasius. TALLFormer: Temporal action localization with a long-memory transformer. In *ECCV*, 2022.

[9] V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem. DAPs: Deep action proposals for action understanding. In *ECCV*, 2016.

[10] G. Gong, L. Zheng, and Y. Mu. Scale matters: Temporal scale aggregation network for precise action localization in untrimmed videos. In *ICME*, 2020.

[11] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui. Zero-shot detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.

[12] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[13] F. C. Heilbron, J. C. Niebles, and B. Ghanem. Fast temporal activity proposals for efficient detection of human actions in untrimmed videos. In *CVPR*, 2016.

[14] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah. The THUMOS challenge on action recognition for videos "in the wild". *CVIU*, 2017.

[15] Z. Jiang, F. F. Xu, J. Araki, and G. Neubig. How can we know what language models know? *TACL*, 2020.

[16] C. Ju, T. Han, K. Zheng, Y. Zhang, and W. Xie. Prompting visual-language models for efficient video understanding. In *ECCV*, 2022.

[17] P. Kaul, W. Xie, and A. Zisserman. Multi-modal classifiers for open-vocabulary object detection. In *ICML*, 2023.

[18] J. Lee, M. Jain, and S. Yun. Few-shot common action localization via cross-attentional fusion of context and temporal dynamics. In *ICCV*, 2023.

[19] C. Lin, C. Xu, D. Luo, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu. Learning salient boundary feature for anchor-free temporal action localization. In *CVPR*, 2021.

[20] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang. BSN: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018.

[21] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen. BMN: Boundary-matching network for temporal action proposal generation. In *ICCV*, 2019.

[22] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[23] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CSUR*, 2023.

[24] Q. Liu and Z. Wang. Progressive boundary refinement network for temporal action detection. In *AAAI*, 2020.

[25] Y. Liu, L. Ma, Y. Zhang, W. Liu, and S.-F. Chang. Multi-granularity generator for temporal action proposal. In *CVPR*, 2019.

[26] S. Nag, X. Zhu, and T. Xiang. Few-shot temporal action localization with query adaptive transformer. *arXiv preprint arXiv:2110.10552*, 2021.

[27] S. Nag, X. Zhu, Y.-Z. Song, and T. Xiang. Zero-shot temporal action detection via vision-language prompting. In *ECCV*, 2022.

[28] S. Nag, O. Goldstein, and A. K. Roy-Chowdhury. Semantics guided contrastive learning of transformers for zero-shot temporal activity detection. In *WACV*, 2023.

[29] Sanath Narayan, Hisham Cholakkal, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. D2-net: Weakly-supervised action localization via discriminative embeddings and denoised activations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13608–13617, 2021.

[30] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[31] T. Phan, K. Vo, D. Le, G. Doretto, D. Adjeroh, and N. Le. ZEETAD: Adapting pretrained vision-language model for zero-shot end-to-end temporal action detection. In *WACV*, 2024.

[32] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[33] Kanchana Ranasinghe, Muzammal Naseer, Salman Khan, Fahad Shahbaz Khan, and Michael S Ryoo. Self-supervised video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2874–2884, 2022.

[34] X. Sun, P. Hu, and K. Saenko. DualCoOp: Fast adaptation to multi-label recognition with limited annotations. *Advances in Neural Information Processing Systems*, 35: 30569–30582, 2022.

[35] J. Tan, J. Tang, L. Wang, and G. Wu. Relaxed transformer decoders for direct action proposal generation. In *ICCV*, 2021.

[36] Anirudh Thatipelli, Sanath Narayan, Salman Khan, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Bernard Ghanem. Spatio-temporal relation modeling for few-shot action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19958–19967, 2022.

[37] M. Xu, C. Zhao, D. S. Rojas, A. Thabet, and B. Ghanem. G-TAD: Sub-graph localization for temporal action detection. In *CVPR*, 2020.

[38] A. Zareian, K. D. Rosa, D. H. Hu, and S.-F. Chang. Open-vocabulary object detection using captions. In *CVPR*, 2021.

[39] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan. Graph convolutional networks for temporal action localization. In *ICCV Workshops*, 2019.

[40] C.-L. Zhang, J. Wu, and Y. Li. ActionFormer: Localizing moments of actions with transformers. In *ECCV*, 2022.

[41] L. Zhang, X. Chang, J. Liu, M. Luo, S. Wang, Z. Ge, and A. Hauptmann. ZSTAD: Zero-shot temporal activity detection. In *CVPR*, 2020.

[42] C. Zhao, A. K. Thabet, and B. Ghanem. Video self-stitching graph network for temporal action localization. In *ICCV*, 2021.

[43] P. Zhao, L. Xie, C. Ju, Y. Zhang, Y. Wang, and Q. Tian. Bottom-up temporal action localization with mutual regularization. In *ECCV*, 2020.

[44] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren. Distance-IoU loss: Faster and better learning for bounding box regression. In *AAAI*, 2020.

[45] Y. Zhong, J. Yang, P. Zhang, C. Li, N. Codella, et al. RegionCLIP: Region-based language-image pretraining. In *CVPR*, 2022.

[46] K. Zhou, J. Yang, C. C. Loy, and Z. Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825, 2022.

[47] K. Zhou, J. Yang, C. C. Loy, and Z. Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

[48] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022.