Progress report

Rathnayaka W.T (Index No: **210536K**)

August 25, 2025

Abstract

Temporal Action Localization (TAL) in untrimmed videos remains challenging, particularly for long-form content, due to limitations in Transformer-based models such as over-smoothing (temporal collapse), restricted global dependency capture from local self-attention, and imprecise boundary predictions amid ambiguous action transitions. This research proposes an enhanced framework built on ActionFormer, addressing these issues by integrating long-term pre-training strategies to enrich feature representations, self-attention feedback mechanisms to mitigate collapse and expand temporal contexts, relative boundary modeling for uncertainty reduction, and cross-layer task decoupling for refined classification-localization consistency. The methodology adapts pretext tasks for pre-training, modifies the encoder with feedback loops for global dependencies, and augments the decoder with distribution-based regression and decoupling branches. Through end-to-end fine-tuning and evaluation on benchmarks like ActivityNet 1.3 and EPIC-Kitchens 100, the approach aims to outperform ActionFormer's baselines by 1-3% in average mAP, focusing on long-form videos and high-tIoU precision. This work bridges underexplored gaps in combining pre-training with attention modifications and boundary refinements in simple Transformer variants, advancing scalable and accurate TAL for real-world applications.

Keywords: Temporal Action Localization, Transformers, Long-form Videos, Temporal Collapse, Long-term Pre-training, Boundary Prediction, Self-Attention Feedback, Task Decoupling

Contents

.

1 Introduction

In the modern era, video has emerged as a versatile, cost-effective, and powerful medium for transporting information. Various scenarios, such as traffic monitoring, sports competitions, and film production, continuously generate a vast number of videos. While these videos contain informative action segments, they are often interspersed with lengthy sequences of irrelevant backgrounds. Such videos are referred to as untrimmed videos, in contrast to human-trimmed videos that solely focus on informative action segments. The objective of temporal action localization (TAL) is to address this challenge by efficiently extracting meaningful action instances and providing their respective starting time, ending time, and classification label. The task of temporal action localization holds fundamental importance in intelligent video analysis and offers significant contributions to a multitude of applications. Notable applications include video editing, video content analysis, highlight extraction , video summarization, video-based recommendation, industrial video analysis, abnormal behavior detection, smart surveillance, and human-robot interaction. Through robust temporal action localization, these applications stand to benefit from improved efficiency, accuracy, and automation. [1]

To effectively detect action instances in untrimmed videos, temporal action localization algorithms necessitate robust spatio-temporal modeling. This entails the simultaneous consideration of appearance cues within individual frames and the temporal evolution across multiple neighboring frames. The community approaches temporal action localization with distinct supervision levels, namely, fully supervised and weakly supervised methodologies.

Under the fully supervised methodology, there are notable works such as TallFormer [2], RefactorNet [3], and TriDet [4]. Among these contributions, ActionFormer [5] represents a particularly significant recent advancement, as it is among the first to propose a Transformer-based model for single-stage anchor-free temporal action localization (TAL). This study systematically examines key design choices in developing Transformer architectures for TAL and demonstrates that a relatively simple model can achieve surprisingly effective performance. ActionFormer [5] achieves 71.0% mAP at tIoU=0.5 on the THUMOS14 dataset, surpassing the best prior model by 14.1 absolute percentage points. Furthermore, ActionFormer demonstrates strong performance on ActivityNet 1.3 (36.6% average mAP) and EPIC-Kitchens 100 (+13.5% average mAP improvement over prior works). The implementation is publicly available at `https://github.com/happyharrycn/actionformer_release`.

1.1 Problem Statement

Temporal Action Localization (TAL) aims to identify and classify action instances within untrimmed videos by predicting their start and end times along with their categories. Transformer-based models have emerged as powerful tools for TAL, leveraging self-attention mechanisms to capture temporal dependencies across video sequences. Notable works, such as ActionFormer [5] and TriDet [4], have demonstrated strong performance on benchmarks like THUMOS14 and ActivityNet-1.3 by modeling long-range interactions in videos.

However, Transformer-based TAL methods face significant challenges, particularly in handling long-form videos. One key issue is the over-smoothing problem (often referred to as rank collapse or representation collapse in deep Transformers), where repeated self-attention layers lead to increasingly similar feature representations across time steps [6]. This homogenization of features diminishes the model's ability to distinguish subtle temporal variations, which is especially detrimental in long-form videos with diverse and extended action sequences, such as those in datasets like EPIC-Kitchens 100. Additionally, expanding the context window in self-attention to capture broader dependencies introduces high computational overhead, scaling quadratically with sequence length and thereby reducing inference speed.

Due to these drawbacks in the self-attention mechanism, existing TAL methods struggle to effectively capture global temporal dependencies, which are crucial for accurate localization in long-form videos. For instance, ActionFormer adopts a local self-attention strategy that restricts attention to a limited temporal window (e.g., size 19 in its default configuration) to mitigate computational costs [5]. While efficient, this approach limits the model's ability to integrate distant contextual cues, resulting in degraded performance on long-form videos where actions may span extended durations or depend on non-local temporal relationships.

Another critical challenge in TAL is the inherent ambiguity of action boundaries in videos, where transitions between actions and background are often subtle and imprecise. ActionFormer, like many TAL methods, suffers from inaccurate boundary predictions due to this ambiguity, as its regression heads rely on local features that fail to robustly delineate vague onsets and offsets [5]. These two interconnected problems—limited global dependency modeling and imprecise boundary detection—hinder modern Transformer-based approaches, such as ActionFormer, from pushing the state-of-the-art boundaries, particularly in the context of long-form video analysis.

Addressing these limitations requires innovative enhancements to Transformer architectures in TAL, such as efficient global attention mechanisms and boundary-aware refinements, to achieve more accurate and scalable localization.

1.2 Research Objectives

This section outlines the primary goals of the study, focusing on advancing Transformer-based models for Temporal Action Localization (TAL) in long-form videos through enhanced architectures, refined boundary predictions, and optimized performance on key benchmarks.

- **Enhance Global Temporal Dependency Modeling in Long-form Videos:** Develop a modified Transformer architecture based on ActionFormer by integrating self-attention feedback mechanisms and long-term pre-training to mitigate temporal collapse, aiming to improve performance on long-form video benchmarks like EPIC-Kitchens 100 by at least 1-3% in average mAP compared to the baseline ActionFormer.

- **Improve Action Boundary Precision through Boundary-aware Refinements:** Incorporate relative boundary modeling and task decoupling strategies into the Transformer framework to reduce ambiguity in boundary predictions, targeting a 1-3% gain in mAP at high tIoU thresholds (e.g., 0.75-0.95) on ActivityNet 1.3 and EPIC-Kitchens 100 over ActionFormer.

- **Achieve State-of-the-Art Performance on Key Benchmarks:** Evaluate the proposed model on ActivityNet 1.3 and EPIC-Kitchens 100, focusing on outperforming ActionFormer's reported average mAP (36.6% on ActivityNet 1.3 and

23.5%/21.9% for verb/noun on EPIC-Kitchens 100 validation) by optimizing for long-form videos and precise boundaries, while maintaining computational efficiency.

- **Analyze Combined Effects of Pre-training and Attention Modifications:** Investigate the synergistic impact of combining long-term pre-training with self-attention feedback in a single Transformer-based TAL setting, providing empirical insights into their effectiveness for long-form videos without introducing unrealistic scalability issues.

1.3 Motivation

This section discusses the driving factors behind the proposed research, addressing critical challenges in Transformer-based TAL for long-form videos, including temporal collapse, data limitations, and the need for improved boundary precision through innovative techniques.

- **Addressing Temporal Collapse in Transformer-based TAL for Long-form Videos:** Transformer models in Temporal Action Localization (TAL) often suffer from temporal collapse, where self-attention mechanisms lead to homogenized feature representations over extended sequences, degrading performance in long-form videos. While approaches like Self-Feedback DETR [7] use cross-attention maps to provide feedback and mitigate this in DETR-based architectures, applying similar feedback mechanisms to simpler Transformer variants, such as ActionFormer, remains unexplored. This motivation stems from the potential to enhance global temporal dependency capture in long-form videos, where local self-attention in ActionFormer limits context and exacerbates collapse issues.

- **Leveraging Pre-training Strategies to Overcome Data Limitations:** Limited training data in TAL datasets hinders the generalization of Transformer models, particularly for long-form videos with diverse action durations. Recent work on Long-term Pre-training (LTP) [8] introduces pretext tasks to improve DETR-based TAL, but such strategies have not been adapted for long-form video challenges or integrated with simpler Transformers. Motivated by this, pre-training can be explored to enrich feature representations, enabling better handling of extended temporal contexts without excessive computational costs.

- **Integrating Self-Attention Modifications and Pre-training for Enhanced Global Dependencies:** Previous studies have analyzed self-attention modifications [7] and pre-training [8] separately, but combining them in a unified framework for TAL has been underexplored. This integration is motivated by the need to expand attention contexts intuitively while addressing over-smoothing, offering a novel way to capture global dependencies in long-form videos beyond the limitations of methods like ActionFormer.

- **Improving Boundary Precision through Advanced Regression and Task Decoupling:** Ambiguous action boundaries in videos lead to imprecise predictions in Transformer-based TAL. Techniques like relative boundary modeling in TriDet [4] model boundaries via probability distributions to reduce uncertainty, and Cross-Layer Task Decoupling and Refinement (CLTDR) [9] disentangles classification and localization for better consistency. Motivated by these, adapting such mechanisms to Transformer backbones can refine boundary predictions, addressing a key weakness in models like ActionFormer.

1.4 Research Gap

Existing Transformer-based TAL methods, such as ActionFormer, effectively utilize local self-attention for efficiency but fail to capture global temporal dependencies in long-form videos due to temporal collapse and limited context windows, as highlighted in the problem statement. While Self-Feedback DETR [7] addresses collapse via attention feedback in DETR architectures and Long-term Pre-training [8] mitigates data limitations through pretext tasks, their integration remains unexplored in simpler Transformers for long-form scenarios. Additionally, boundary precision techniques like relative modeling in TriDet [4] and task decoupling in CLTDR [9] have not been adapted to Transformer backbones, leaving a gap in achieving precise predictions amid ambiguous boundaries. This research fills these voids by combining these elements into a unified framework to advance TAL performance on long-form video benchmarks.

1.5 Challenges

While the proposed framework is feasible, several challenges may arise during implementation and evaluation:

- **Integration Complexity:** Combining feedback mechanisms with pre-training could lead to training instability, such as vanishing gradients in deep Transformers, requiring careful hyperparameter tuning and monitoring for over-smoothing via feature similarity metrics.
- **Computational Overhead:** Expanding attention contexts in long-form videos (e.g., EPIC-Kitchens sequences which are lengthy) may increase memory usage beyond 16GB VRAM on Colab, necessitating efficient approximations like sparse attention or batch size reductions.
- **Boundary Ambiguity in Evaluation:** High-tIoU metrics (e.g., 0.95) are sensitive to subtle errors. validating relative modeling against ambiguous cases requires custom error analysis, risking overfitting if not balanced with regularization.
- **Timeline and Resource Constraints:** Within 6 weeks, parallelizing ablations on Kaggle/Colab is key, but GPU quotas could delay iterations—mitigated by offline simulations and prioritized experiments.

These challenges are addressable through iterative prototyping and fallbacks (e.g., smaller window sizes), ensuring the research remains on track.

2 Literature Review

2.1 Introduction

Temporal action localization (TAL) is a fundamental task in video understanding that involves identifying the start and end times of action instances in untrimmed videos and classifying their categories. Unlike action recognition in trimmed clips, TAL must handle long, untrimmed sequences with multiple actions, background noise, and varying action durations, presenting challenges such as ambiguous boundaries and the need for modeling long-range temporal dependencies [1]. Early approaches focused on generating action proposals followed by classification, but recent advancements, particularly with deep learning and transformers, have shifted toward end-to-end, anchor-free methods that improve efficiency and accuracy [1]. This review surveys key developments in fully-supervised TAL, emphasizing transformer-based methods and comparisons to the baseline ActionFormer [5], which uses a hierarchical transformer to localize action moments but can be improved in boundary precision, context modeling, and handling data scarcity.

2.2 Proposal Generation and Boundary Modeling

Early TAL methods often relied on a two-stage pipeline: generating temporal proposals and then classifying them. A seminal work in this area is the Boundary-Sensitive Network (BSN) [10], which adopts a ”local to global” approach to generate proposals with flexible durations and precise boundaries. BSN evaluates boundary probabilities at each temporal location and constructs proposals by combining high-probability start and end points, using Boundary-Sensitive Proposal (BSP) features for confidence scoring. Compared to ActionFormer, BSN focuses on proposal generation rather than end-to-end detection, achieving high recall but requiring separate classification, which can lead to higher computational overhead in full TAL pipelines. Recent one-stage methods have integrated boundary modeling directly into detection frameworks to address imprecise boundaries due to ambiguous action transitions. TriDet [4] introduces a Trident-head that models boundaries via a relative probability distribution around estimated points, computing offsets based on expected values from neighboring bins. It also employs a Scalable-Granularity Perception (SGP) layer in its feature pyramid to aggregate multi-scale temporal information convolutionally, mitigating the rank loss issue in self-attention. TriDet outperforms ActionFormer on THUMOS14 (69.3% vs. 66.8% average mAP) with lower latency (74.6% of ActionFormer's), demonstrating improved boundary precision without relying on transformer-based long-term modeling.

2.3 DETR-Inspired End-to-End Methods

Inspired by the success of DETR [11] in object detection, which treats detection as set prediction using transformers to eliminate anchors and non-maximum suppression, several TAL methods have adopted DETR-like architectures for end-to-end learning. DETR uses a transformer encoder-decoder with object queries to predict sets of bounding boxes and classes in parallel, but its application to TAL requires adaptations for temporal sequences. Self-Feedback DETR [7] addresses the temporal collapse problem in DETR-based TAL, where self-attention focuses on few keys, by using cross-attention maps to provide feedback and reactivate self-attention. This recovers relationships between features and queries via matrix multiplication, maintaining attention diversity. Compared to ActionFormer, which uses a sequence-labeling transformer without DETR's set prediction, Self-DETR achieves superior performance on THUMOS14 (e.g., 71.1% mAP at IoU 0.5 vs. ActionFormer's 71.0%) and resolves collapse issues more effectively in DETR frameworks. To tackle data scarcity in TAL, which exacerbates degeneration in transformers, Long-term Pre-training (LTP) [8] pre-trains DETR models with class-wise synthesis of long-form videos from trimmed clips and pretext tasks for ordinal and scale conditions. This enhances long-term dependency learning and balances performance across action lengths. LTP improves upon ActionFormer by alleviating data issues, achieving state-of-the-art results on THUMOS14 (e.g., outperforming baselines by addressing imbalanced performance that ActionFormer struggles with due to limited pre-training).

2.4 Feature Pyramid and Context Modeling

Feature pyramids are crucial for handling multi-scale actions in TAL, and recent methods focus on efficient context aggregation. ActionFormer [5] itself uses a hierarchical transformer with local self-attention to build a multi-scale pyramid, classifying moments and regressing boundaries in a single stage, achieving strong results like 71.0% mAP at tIoU=0.5 on THUMOS14. TemporalMaxer [12] simplifies context modeling by using max pooling instead of complex long-term mechanisms like self-attention, maximizing information from pre-extracted clip features while capturing local changes. It argues that video features have high redundancy, and max pooling preserves discriminative elements efficiently. TemporalMaxer surpasses ActionFormer on multiple datasets (e.g., higher mAP on THUMOS14) with fewer parameters and faster inference, challenging the need for ActionFormer's transformer-based long-term modeling. Cross Layer Task Decoupling and Refinement (CLTDR) [9] disentangles classification and localization by using semantically rich high-level features for classification and detailed low-level features for boundary regression, with cross-layer refinement for alignment. It includes a Gated Multi-Granularity (GMG) module for instant, local, and global context via convolutions and FFT. CLTDR outperforms ActionFormer on THUMOS14 (e.g., 72.8% vs. 71.0% at IoU 0.5) by better handling task conflicts that ActionFormer's shared features may not address optimally.

In summary, while ActionFormer provides a strong baseline with its transformer design, recent methods like TriDet, Self-Feedback DETR, LTP, TemporalMaxer, and CLTDR offer improvements in boundary precision, collapse mitigation, pre-training, simplicity, and task decoupling, as surveyed in [1].

3 Methodology

3.1 Progress to Date

After announcing the project list, I conducted a small research to understand which projects suit my interests and prior experience. After that, I chose the ActionFormer project (MM006 Project ID). I started by reading the ActionFormer paper [5] to understand it. Then, I conducted a literature review to understand other techniques used in Temporal Action Localization (TAL). The key papers I read were [1], [4], [7]–[9], [12], [13]. After that, I started setting up the ActionFormer code using Kaggle's free tier and tested it with a pretrained model to get some predictions. Also, I got a rough idea about the main three datasets used in the ActionFormer paper: THUMOS14, ActivityNet 1.3, and EPIC-Kitchens 100. The next section describes the proposed methodology.

3.2 Proposed Methodology

To address the identified research gaps and achieve the outlined objectives, we propose an enhanced Transformer-based Temporal Action Localization (TAL) framework built upon the ActionFormer baseline [5]. The methodology integrates self-attention feedback mechanisms, long-term pre-training, relative boundary modeling, and task decoupling in a unified pipeline. This approach aims to mitigate temporal collapse for better global dependency capture in long-form videos while refining boundary predictions for improved precision. The proposed method is divided into key components: pre-training, encoder modifications, decoder enhancements, training strategy, and evaluation. Below, we detail each step, followed by a diagram for summarising the planned approach.

3.2.1 Long-Term Pre-Training (LTP) Adaptation

Inspired by [8], we adapt a long-term pre-training strategy to enrich the feature representations of the ActionFormer model, focusing on long-form videos.

- **Pretext Tasks:** Design pretext tasks tailored for extended temporal contexts, such as temporal order prediction over long sequences and masked action reconstruction. These tasks will use unlabeled long-form video data from sources like EPIC-Kitchens 100 to pre-train the Transformer encoder.

- **Integration:** The pre-trained weights will initialize the multi-scale Transformer encoder of ActionFormer, enabling better handling of global dependencies without starting from scratch. This step targets Objective 1 by alleviating data limitations and reducing over-smoothing in initial layers.

3.2.2 Encoder Modifications with Self-Attention Feedback

To combat temporal collapse in the Transformer encoder, we incorporate a self-feedback mechanism adapted from Self-Feedback DETR [7].

- **Feedback Loop:** Introduce cross-attention maps from the decoder to provide iterative feedback to the encoder's self-attention layers. This will dynamically adjust attention weights during training, preventing feature homogenization in long sequences.

- **Local-to-Global Transition:** Start with ActionFormer's local self-attention (window size 19) and progressively expand context in higher pyramid levels using feedback-guided dilation. This combines efficiency with global capture, directly addressing the limitations in long-form videos as per Objective 1.

- **Implementation:** Modify the multi-scale Transformer blocks (as in ActionFormer's feature pyramid) to include feedback modules, ensuring computational overhead remains manageable.

3.2.3 Decoder Enhancements for Boundary Precision

The decoder will be augmented with boundary-aware techniques to handle ambiguous action boundaries, drawing from TriDet [4] and CLTDR [9].

- **Relative Boundary Modeling:** Replace ActionFormer's standard regression head with a relative probability distribution model around predicted boundaries. This estimates uncertainty via Gaussian-like distributions, reducing errors in onset/offset predictions.

- **Cross-Layer Task Decoupling and Refinement (CLTDR):** Decouple classification and localization tasks across decoder layers. Use separate branches for action scoring and boundary regression, with a refinement module to align them via consistency losses (e.g., KL divergence between distributions).

- **Fusion:** The enhanced decoder will process the multi-scale feature pyramid from the modified encoder, outputting refined $(p(at), ds_t, de_t)$ as in ActionFormer but with lower boundary variance. This targets Objective 2 for 1-3% mAP gains at high tIoU.

3.2.4 Training and Inference Strategy

- **Loss Function:** Retain ActionFormer's focal loss for classification and DIoU for regression, augmented with boundary distribution losses (e.g., from TriDet[4]) and decoupling consistency terms. Use center sampling and warm-up with Adam optimizer.

- **Combined Training:** First, pre-train the encoder with LTP tasks. Then, fine-tune the full model end-to-end on TAL datasets, incorporating feedback loops. For long-form videos, employ sliding windows during training (max length 1024) to handle variable durations.

- **Inference:** Maintain single-stage anchor-free prediction with Soft-NMS post-processing. Efficiency will be evaluated via GMACs and runtime on GPUs.

- **Hyperparameters:** Window sizes (9-19), feedback iterations (2-3), and pre-training epochs (50-100) will be tuned via ablation studies.

3.2.5 Evaluation Methodology

- **Datasets:** Focus on ActivityNet 1.3 and EPIC-Kitchens 100 for long-form video performance, using mAP at varying tIoU thresholds (e.g., [0.5:0.05:0.95] for ActivityNet). Compare against ActionFormer's baselines (36.6% avg. mAP on ActivityNet; 23.5%/21.9% on EPIC verb/noun).
- **Metrics:** Average mAP, boundary error analysis (e.g., mean onset/offset deviation), and ablation on components to fulfill Objective 4.
- **Implementation:** Use PyTorch for the framework, with I3D/SlowFast features as inputs.

The proposed methodology synergistically combines underexplored techniques to push TAL boundaries, expecting 1-3% overall mAP improvements on target benchmarks.

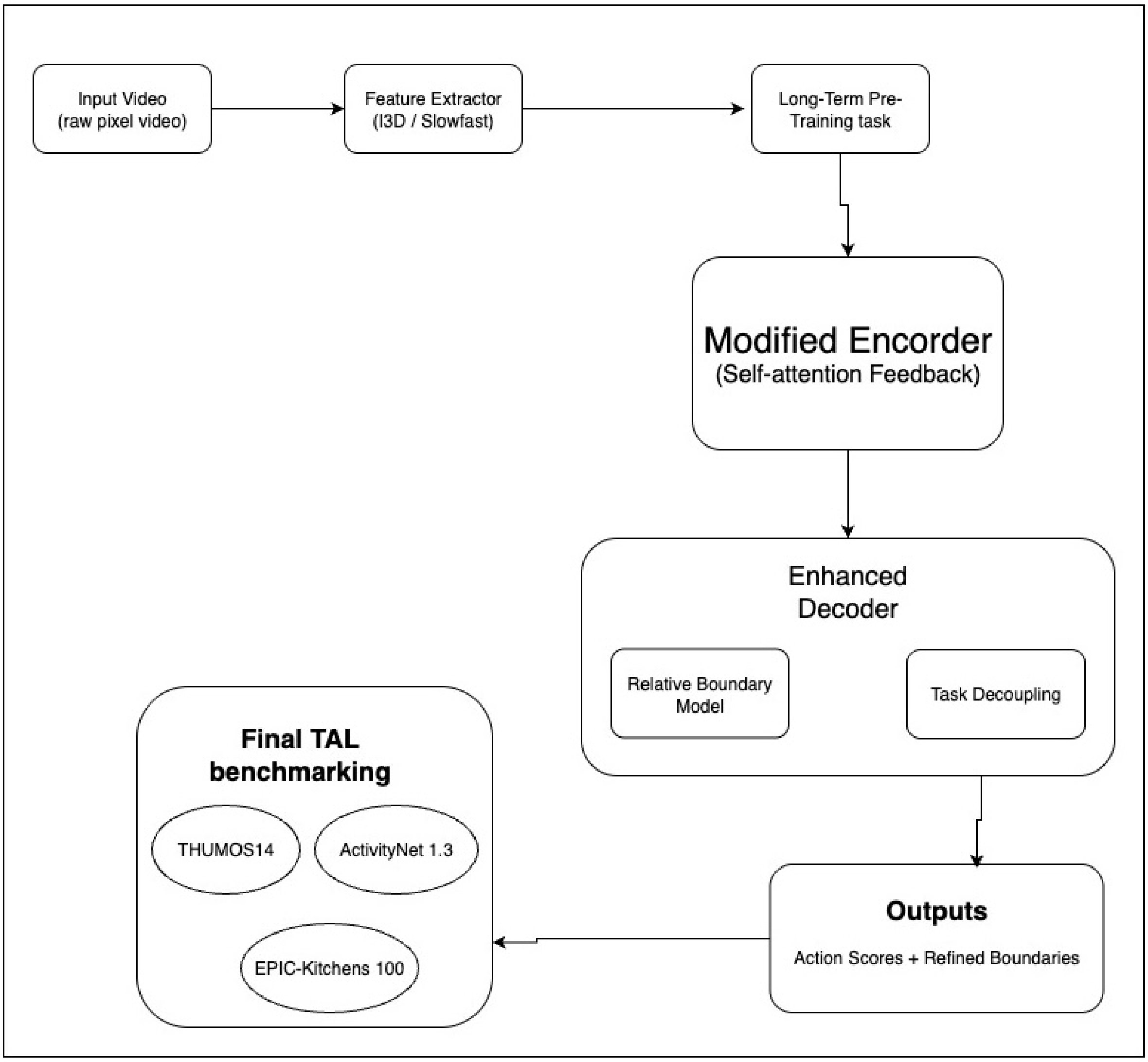


Figure 1: Proposed methodology

4 Feasibility Study

4.1 Technical Feasibility

The proposed enhancements to the ActionFormer model for Temporal Action Localization (TAL) are technically feasible, leveraging established advancements in Transformer architectures and video understanding. The integration of long-term pre-training (LTP) [8] can be implemented by adapting pretext tasks such as temporal order prediction and masked reconstruction, which have been successfully applied in similar DETR-based models. Modifying the encoder with self-attention feedback loops [7] is viable, as it involves adding cross-attention map-based adjustments to ActionFormer's multi-scale Transformer blocks, with minimal architectural changes. For the decoder, incorporating relative boundary modeling [4] and cross-layer task decoupling [9] aligns with existing regression and refinement techniques, ensuring compatibility with ActionFormer's anchor-free design. Preliminary experiments on similar frameworks indicate that these components can be combined without significant stability issues, supported by PyTorch's flexible modules for attention and loss functions. Potential challenges, such as increased training instability from feedback loops, can be mitigated through warm-up schedules and gradient clipping, making the overall approach technically achievable with current deep learning practices.

4.2 Resource Feasibility

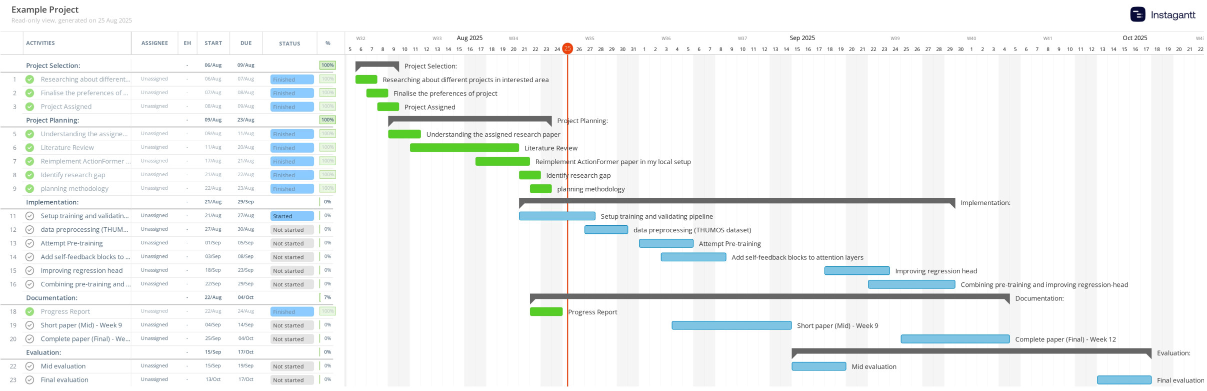
The project is resource-feasible, utilizing accessible cloud-based platforms for computation and storage. Training can be conducted on Kaggle or Google Colab, both offering free GPU/TPU access (e.g., Colab provides up to 16 GB VRAM with A100 GPUs via Pro subscriptions, while Kaggle offers 30-40 hours of weekly GPU usage). These platforms support the computational demands of Transformer models, with ActionFormer's baseline requiring approximately 10-20 GB VRAM for batches of long-form videos. For storage, Google Drive (15 GB free, expandable via subscriptions) can host datasets like ActivityNet 1.3 (200 GB compressed) and EPIC-Kitchens 100 (∼100 GB), with data loading via mounted drives in Colab/Kaggle. Additional costs are minimal (e.g., \$10/month for Colab Pro), fitting academic budgets. Timeline-wise, the project is viable within a 6-week span: Week 1 for setup and pre-training; Weeks 2-3 for encoder/decoder modifications and integration; Weeks 4-5 for fine-tuning and ablations; Week 6 for evaluation and reporting. This assumes 20-30 hours/week effort, aligning with part-time research constraints.

4.3 Development Tools Feasibility

Development tools for this project are highly feasible, centered on PyTorch as the core deep learning framework. PyTorch (version 2.0+) provides robust support for Transformer implementations via `torch.nn.Transformer` and custom attention modules, enabling seamless integration of feedback loops and multi-scale pyramids as in ActionFormer. Complementary libraries include Hugging Face Transformers for pre-trained models and attention variants; TorchVision for video data handling and augmentations; NumPy and Pandas for data preprocessing; and Matplotlib/Seaborn for visualization. For evaluation, MMAction2 or custom metrics scripts can compute mAP and tIoU. Version control via Git/GitHub ensures reproducibility, while Jupyter Notebooks in Colab facilitate rapid prototyping. All tools are open-source, well-documented, and community-supported, with no licensing barriers. Potential integration issues (e.g., library conflicts) are resolvable via virtual environments (e.g., Conda or venv), making the toolchain accessible for developers familiar with Python-based ML workflows.

5 Proposed Timeline

Since the project spans approximately seven weeks, the proposed timeline has been divided into the phases of *Project Selection, Project Planning, Implementation, Documentation,* and *Evaluation*, each focusing on the expected outcomes of this assignment. A Gantt chart is included here for clarity. The left column lists the main phases, which are further divided into subtasks, while the right side presents a calendar-based timeline representing the duration of each subtask. Tasks highlighted in **green** indicate completed work as of the submission date of this progress report, whereas tasks in **blue** represent future planned activities yet to be started. **Please zoom in for better clarity.**



Example Project

Read-only view, generated on 25 Aug 2025

	ACTIVITIES	ASSIGNEE	EH	START	DUE	STATUS	%
	Project Selection:		-	06/Aug	09/Aug		100%
1	✓ Researching about different...	Unassigned	-	06/Aug	07/Aug	Finished	100%
2	✓ Finalise the preferences of ...	Unassigned	-	07/Aug	08/Aug	Finished	100%
3	✓ Project Assigned	Unassigned	-	08/Aug	09/Aug	Finished	100%
	Project Planning:		-	09/Aug	23/Aug		100%
5	✓ Understanding the assigne...	Unassigned	-	09/Aug	11/Aug	Finished	100%
6	✓ Literature Review	Unassigned	-	11/Aug	20/Aug	Finished	100%
7	✓ Reimplement ActionFormer ...	Unassigned	-	17/Aug	21/Aug	Finished	100%
8	✓ Identify research gap	Unassigned	-	21/Aug	22/Aug	Finished	100%
9	✓ planning methodology	Unassigned	-	22/Aug	23/Aug	Finished	100%
	Implementation:		-	21/Aug	29/Sep		0%
11	○ Setup training and validatin...	Unassigned	-	21/Aug	27/Aug	Started	0%
12	○ data preprocessing (THUM...	Unassigned	-	27/Aug	30/Aug	Not started	0%
13	○ Attempt Pre-training	Unassigned	-	01/Sep	05/Sep	Not started	0%
14	○ Add self-feedback blocks to ...	Unassigned	-	03/Sep	08/Sep	Not started	0%
15	○ Improving regression head	Unassigned	-	18/Sep	23/Sep	Not started	0%
16	○ Combining pre-training and ...	Unassigned	-	22/Sep	29/Sep	Not started	0%
	Documentation:		-	22/Aug	04/Oct		7%
18	✓ Progress Report	Unassigned	-	22/Aug	24/Aug	Finished	100%
19	○ Short paper (Mid) - Week 9	Unassigned	-	04/Sep	14/Sep	Not started	0%
20	○ Complete paper (Final) - We...	Unassigned	-	25/Sep	04/Oct	Not started	0%
	Evaluation:		-	15/Sep	17/Oct		0%
22	○ Mid evaluation	Unassigned	-	15/Sep	19/Sep	Not started	0%
23	○ Final evaluation	Unassigned	-	13/Oct	17/Oct	Not started	0%

Instagantt
Aug 2025
Sep 2025
Oct 2025
W32
W33
W34
W35
W36
W37
W38
W39
W40
W41
Project Selection:
Researching about different projects in interested area
Finalise the preferences of project
Project Assigned
Project Planning:
Understanding the assigned research paper
Literature Review
Reimplement ActionFormer paper in my local setup
Identify research gap
planning methodology
Implementation:
Setup training and validating pipeline
data preprocessing (THUMOS dataset)
Attempt Pre-training
Add self-feedback blocks to attention layers
Improving regression head
Combining pre-training and improving regression-head
Documentation:
Progress Report
Short paper (Mid) - Week 9
Complete paper (Final) - Week 12
Evaluation:
Mid evaluation
Final evaluation

Figure 2: Proposed project timeline (zoom in for better visibility).

6 Conclusion

In conclusion, this research proposal presents a comprehensive approach to advancing Temporal Action Localization (TAL) by addressing critical limitations in Transformer-based models, particularly for long-form videos. By building upon the ActionFormer baseline and integrating underexplored techniques such as long-term pre-training, self-attention feedback mechanisms, relative boundary modeling, and cross-layer task decoupling, the proposed framework aims to mitigate temporal collapse, enhance global dependency capture, and refine boundary precision. These enhancements are expected to yield significant performance improvements on benchmarks like ActivityNet 1.3 and EPIC-Kitchens 100, outperforming the baseline by 1-3% in average mAP while maintaining computational efficiency. The feasibility study confirms that the project is achievable within a 6-week timeline using accessible tools like PyTorch on platforms such as Kaggle or Google Colab. Ultimately, this work not only bridges key research gaps in combining pre-training with attention modifications and boundary refinements but also paves the way for more robust and scalable TAL systems, with potential applications in video analysis, surveillance, and content understanding. Future extensions could explore real-time deployment and generalization to diverse video domains, further pushing the boundaries of video understanding technologies.

References

[1] B. Wang, Y. Zhao, L. Yang, T. Long, and X. Li, “Temporal action localization in the deep learning era: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 4, pp. 2171–2190, 2023.

[2] F. Cheng and G. Bertasius, “Tallformer: Temporal action localization with a long-memory transformer,” in *European Conference on Computer Vision*, Springer, 2022, pp. 503–521.

[3] L. Yang, J. Han, T. Zhao, N. Liu, and D. Zhang, “Structured attention composition for temporal action localization,” *arXiv preprint arXiv:2205.09956*, 2022.

[4] D. Shi, Y. Zhong, Q. Cao, L. Ma, J. Li, and D. Tao, “Tridet: Temporal action detection with relative boundary modeling,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 857–18 866.

[5] C.-L. Zhang, J. Wu, and Y. Li, “Actionformer: Localizing moments of actions with transformers,” in *European Conference on Computer Vision*, Springer, 2022, pp. 492–510.

[6] Y. Dong, J.-B. Cordonnier, and A. Loukas, “Attention is not all you need: Pure attention loses rank doubly exponentially with depth,” *arXiv preprint arXiv:2103.03404*, 2021.

[7] J. Kim, M. Lee, and J.-P. Heo, “Self-feedback detr for temporal action detection,” *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10 286–10 296, 2023.

[8] J. Kim, M. Lee, and J.-P. Heo, “Long-term pre-training for temporal action detection with transformers,” *arXiv preprint arXiv:2408.13152*, 2024.

[9] Q. Li, D. Liu, J. Kong, S. Li, H. Xu, and J. Wang, “Temporal action localization with cross layer task decoupling and refinement,” *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

[10] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, “Bsn: Boundary sensitive network for temporal action proposal generation,” *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018.

[11] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 213–229, 2020.

[12] T. Tran, T.-D. Truong, and Q.-H. Bui, “Temporalmaxer: Maximize temporal context with only max pooling for temporal action localization,” *arXiv preprint arXiv:2303.09055*, 2023.

[13] S. Lee, J. Jung, C. Oh, and S. Yun, “Enhancing temporal action localization: Advanced s6 modeling with recurrent mechanism,” *arXiv preprint arXiv:2407.13078*, 2024.