DOCTORAL THESIS

On the Quest of Generalizable Video Understanding

Doctorate Program in Mathematics and Informatics

Author: David PUJOL PERICH

Director and Tutor: Prof. Sergio ESCALERA GUERRERO

Co-Director: Dr. Albert CLAPÉS SINTES

Department of Mathematics and Informatics

UNIVERSITAT DE BARCELONA

Life isn't about finding yourself. Life is about creating yourself.
George Bernard Shaw

Acknowledgements

Despite how impossible it feels to properly acknowledge all the people that have supported me throughout this journey, this is my most sincere attempt to show my gratitude and recognition. Thank you Sergio, for taking me on as a student without ever showing a glimpse of doubt, for trusting my judgments and intuitions, and for always radiating a contagious optimism. Thank you Albert, not only for being an incredible, dedicated supervisor, but also for always making the time. The time for engaging in technical discussions, but perhaps even more importantly, for ours Friday-morning chats about life. Your humbleness, generosity, kindness and guideness provided me with much-needed certainties and confidence during times of doubts and uncertainty. I am also deeply grateful to the other exceptional advisors from my previous research experiences. Thank you, Pere, Albert, Igor, and Volkan, for sharing your passion for this job... it was your enthusiasm that ultimately convinced me that pursuing a Ph.D. was the right path.

On a more personal note, thanks to all Hupba members. My times at Altell have been a true pleasure that I shall surely miss. Thank you Germán, Rubén, Artur, Hunor, Smrity, Johnny, Guillem, Edison, John, Iñaki, Manu, Yudong, Julio, Sorina, Meysam and Javi! I would also like to thank everyone from the MaVi Lab at the University of Bristol. Thank you Dima and Mike for offering me such a warm welcome, your invaluable guidance, and crucial support during my stay. Thank you also to the rest of the MaVi team: Saptarshi, Rodrhi, Siddhant, Prajwall, Sam, Ahmad, Jiahe, Zhifan and Omar.

Gràcies als meus amics, amb una especial menció a l'Oriol i el Pol, per aguantar interminables histories, queixes, preocupacions, i per sort, moltes alegries. Finalment, gràcies a tota la meva familia per sempre donar-me suport en totes les aventures i reptes. Gràcies als meus germans, l'Arnau, la Jordina i la Judit, per recolzar-me, per entendre les meves frustracions i per recordar-me cada dia de l'importància de les petites coses. Gràcies a tu, Paula... sobren les paraules. Simplement, gràcies per ser allà. Sempre. Per treure'm un somriure, i per recordar-me que el got sempre està mig ple. Sense tu això hagués sigut impossible. Finalment, gràcies mama i papa, per confiar cegament en mi, per donar-me una educació i uns valors, que són els que m'han fet la persona que sóc avui. Res m'hagués fet més feliç que ho poguessis veure, papa, però confio que ens veus, allà on siguis.

Abstract

Recent advances in deep learning have enabled remarkable progress in automatic visual scene understanding, particularly for complex and dynamic data such as videos. Despite strong performance on controlled benchmarks, modern video understanding systems continue to struggle with a fundamental property of intelligence: generalization. Their performance often degrades under distribution shifts, ambiguous user inputs, or when transferred to new tasks and domains. This thesis addresses these limitations by investigating generalization in video understanding along three axes: visual out-of-distribution (OOD) generalization, linguistic OOD generalization, and efficient knowledge transferability.

First, Chapter 2 studies visual OOD generalization in Temporal Action Localization (TAL), addressing the lack of generalization associated with temporal features with visual domain shifts. We have demonstrated how domain-specific models were prone to failing on new domains because they relied on spurious correlations. To mitigate these effects and improve performance, we introduce an adaptive learning method for action localization within SADA, which aims at learning domain-invariant spatio-temporal dynamics. Second, Chapter 3 addresses linguistic OOD generalization in Video Moment Retrieval (VMR). We identify a significant gap between the highly descriptive, caption-like queries commonly used during training and the ambiguous, underspecified queries encountered in real user scenarios. We formalize this linguistic generalization gap and propose a method to quantify and mitigate it, enabling models to better generalize across diverse query formulations. Finally, Chapter 4 focuses on knowledge transferability and efficiency in video-language models. Given the high computational cost of adapting large pretrained models, we introduce SDST, a parameter-efficient fine-tuning framework that facilitates the effective reuse of pretrained video representations for grounding tasks.

Beyond these core contributions, this thesis introduces new cross-domain evaluation settings for temporal action understanding and proposes benchmarking protocols to assess linguistic generalization in retrieval tasks. Together, these contributions advance the state of the art in OOD generalization and knowledge transferability, paving the way toward scalable, robust, and linguistically flexible video understanding systems suitable for real-world deployment.

Resum

Els avenços recents en aprenentatge profund han permès un progrés notable en la comprensió automàtica d'escenes visuals, especialment en dades complexes i dinàmiques com els vídeos. Malgrat el seu bon rendiment en benchmarks controlats, els sistemes moderns continuen tenint dificultats amb una propietat fonamental de la intel · ligència: la generalització. El seu rendiment sovint es degrada davant canvis en la distribució de les dades, *queries* ambigües o quan es transfereixen a noves tasques i dominis. Aquesta tesi aborda aquestes limitacions investigant la generalització en la comprensió de vídeos al llarg de tres eixos: la generalització fora de distribució (OOD) visual, la lingüística i la transferència eficient de coneixement.

En primer lloc, el Capítol 2 estudia la generalització OOD visual en Temporal Action Localization (TAL), abordant la manca de generalització associada a les característiques temporals sota canvis de domini visual. Demostrem que els models específics de domini tendeixen a fallar en nous entorns a causa de la seva dependència de correlacions espúries. Per mitigar aquests efectes i millorar el rendiment, introduïm un mètode d'aprenentatge adaptatiu per a la localització d'accions dins del *framework* SADA, amb l'objectiu d'aprendre dinàmiques espai-temporals invariants al domini. El Capítol 3 aborda la generalització OOD lingüística en Video Moment Retrieval (VMR). Identifiquem una bretxa significativa entre les *queries* altament descriptives, utilitzades habitualment durant l'entrenament, i les *queries* ambigües i subespecificades, més pròpies d'escenaris reals d'ús. Formalitzem aquesta bretxa de generalització lingüística i proposem un mètode per quantificar-la i mitigar-la, permetent que els models generalitzin millor davant formulacions de *queries* diverses. Finalment, el Capítol 4 es centra en la transferibilitat del coneixement i l'eficiència en models visió-llenguatge (VLMs) per a vídeo. Donat l'elevat cost computacional associat a l'adaptació de models pre-entrenats de gran escala, introduïm SDST, un framework que facilita la reutilització efectiva de representacions pre-entrenades per a tasques de *grounding*.

Més enllà d'aquestes contribucions principals, aquesta tesi introdueix nous escenaris d'avaluació inter-domini per a la comprensió temporal d'accions i proposa protocols de benchmarking per avaluar la generalització lingüística en tasques de localització. En conjunt, aquestes aportacions avancen l'estat de l'art en generalització fora de distribució i transferència de coneixement, i obren el camí cap a sistemes de comprensió de vídeo escalables, robustos i lingüísticament flexibles, adequats per al seu desplegament en entorns reals.

Resumen

Los avances recientes en aprendizaje profundo han permitido un progreso notable en la comprensión automática de escenas visuales, especialmente en datos complejos como los vídeos. A pesar de su buen rendimiento en benchmarks controlados, los sistemas modernos siguen teniendo dificultades con una propiedad fundamental de la inteligencia: la generalización. Su rendimiento suele degradarse ante cambios en la distribución de los datos, o cuando se transfieren a nuevas tareas y dominios. Esta tesis aborda estas limitaciones investigando la generalización en la comprensión de vídeos a lo largo de tres ejes: la generalización fuera de distribución (OOD) visual, la lingüística y la transferencia eficiente de conocimiento.

En primer lugar, el Capítulo 2 estudia la generalización OOD visual en Temporal Action Localization (TAL), abordando la falta de generalización asociada a las características temporales bajo cambios de dominio visual. Demostramos que los modelos específicos de dominio tienden a fallar en nuevos entornos debido a su dependencia de correlaciones espurias. Para mitigar estos efectos y mejorar el rendimiento, introducimos un método de aprendizaje adaptativo para la localización de acciones dentro del *framework* SADA para aprender dinámicas espacio-temporales invariantes al dominio. El Capítulo 3 aborda la generalización OOD lingüística en Video Moment Retrieval (VMR). Identificamos una brecha significativa entre las *queries* altamente descriptivas, utilizadas habitualmente durante el entrenamiento y las *queries* ambiguas, mas propias de escenarios reales de uso. En este capítulo formalizamos esta brecha de generalización lingüística y proponemos un método para cuantificarla y mitigarla, permitiendo que los modelos generalicen mejor ante formulaciones de *queries* diversas. Por último, el Capítulo 4 se centra en la transferibilidad del conocimiento y la eficiencia en modelos visión-lenguaje (VLMs) para vídeos. Dado el elevado coste computacional asociado a la adaptación de modelos pre-entrenados de gran escala, introducimos SDST, un framework que facilita la reutilización efectiva de representaciones pre-entrenadas para tareas de *grounding*.

Más allá de estas contribuciones principales, esta tesis introduce nuevos escenarios de evaluación inter-dominio para la comprensión temporal de acciones y propone protocolos de benchmarking para evaluar la generalización lingüística en tareas como VMR.. En conjunto, estas aportaciones avanzan el estado del arte en generalización fuera de distribución así como de transferencia de conocimiento, allanando el camino hacia sistemas escalables, robustos y lingüísticamente flexibles, adecuados para su despliegue en entornos reales.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Generalization—the ability to reuse knowledge across tasks and environments—is a defining property of human intelligence [8]. Humans rarely learn from scratch; instead, they rapidly adapt prior experience to new but related situations (see Fig. 1.1). A person who learns how to throw a stone can immediately adapt to throwing a spear or throwing a ball, transferring knowledge of force, trajectory, and coordination. Similarly, in the visual domain, when humans learn to recognize a dog, they do so independently of whether it is presented in a photograph, a cartoon, or even a silhouette [7, 23]. This flexibility arises from the capacity to extract invariant structures rather than memorizing appearance-specific traits [25, 15]. Such invariance emerges from neural plasticity and hierarchical sensory processing shaped after millions of years of evolution [20, 46]. Neuroscientific evidences [23, 30] show that the brain systematically builds representations that are stable across transformations. These evidences also demonstrate its ability to reuse neural circuits across tasks, particularly in hierarchical sensory pathways such as the ventral visual stream [6, 23], which supports object recognition across changes in illumination, viewpoint, and occlusion [10]. Biological cognition thus provides a compelling model of efficient, robust, and transferable generalization.

Inspired by these generalization capabilities of animals, humans have long sought to replicate these mechanisms in artificial intelligence systems. Early approaches to artificial intelligence—particularly symbolic and rule-based systems developed in the mid-20th century [3]— relied on hand-crafted abstractions that proved limited when faced with even minor deviations from the assumptions encoded during design or training [113]. These limitations became especially apparent after the emergence of computer vision, where early systems struggled to cope with the variability inherent in real-world visual data [12]. The rise of deep learning marked an important paradigm shift. Convolutional Neural Networks (CNNs) [11], inspired by the hierarchical organization of the visual cortex [4], enabled data-driven learning of multi-level representations directly from raw sensory input [24]. Rather than relying on manually engineered features, these models automatically learn task-relevant representations from data, leading

to considerable improvements across a wide range of visual recognition tasks [118]. More recently, Transformer-based architectures [68], originally designed for sequence modeling in language, have extended this paradigm to both vision and multi-modal domains. By leveraging self-attention mechanisms, Transformers capture long-range dependencies and global context more effectively than convolutional operations, enabling models to learn richer, more flexible representations that can transfer across tasks, modalities, and domains.

These advances naturally extended to video understanding. By incorporating temporal modeling through recurrent neural networks [33], temporal convolutions [67], and later Transformer-based architectures [68, 108], modern systems achieved strong performance in tasks such as action recognition [50], temporal segmentation [59], event detection [43], and video captioning [37]. Among these, the ability to search and retrieve specific moments from large video collections has become increasingly important. This task, known as Video Moment Retrieval (VMR) [47, 53], requires localizing the temporal segment in a video that corresponds to a natural language query. VMR, thus, lies at the intersection of visual perception, temporal reasoning, and language understanding, laying a considerably potential in applications such as large-scale content indexing [35], video moderation [64], surveillance analysis [82], assistive technologies, and media retrieval [198] (see Fig. 1). A closely related task, Temporal Action Localization (TAL) [43, 89], removes the linguistic component of VMR, simplifying the open-ended textual queries to a predefined set of action categories—e.g., abnormal behavior for surveillance applications, or inappropriate content for content moderation. While less expressive, TAL provides a more controlled setting that avoids the open challenges associated with vision–language alignment, allowing the study of visual-temporal generalization in isolation.

Despite their empirical success, modern deep learning models exhibit persistent generalization failures. Performance often degrades significantly when deployment conditions differ from those observed during training, even if such changes may appear superficial to humans [22, 113]. This issue is already well documented in image-based models [76], where systems are known to rely on dataset-specific cues such as background textures, color statistics, or object co-occurrences rather than semantic content [42]. In video understanding, these limitations become significantly more pronounced due to additional sources of variability. Videos introduce long-range temporal dependencies [9], persistent background, camera motion and viewpoint changes [71], or an intrinsic ambiguity in action boundaries and annotations [32, 171]. As a consequence, optimizing the corresponding complex objectives often leads

Figure 1.1: The role of generalization, presented in various ways, is key to explain human intelligence. For instance, allowing the men in the paleolithic to rapidly generalize the principles of *"throwing a spear"* to that of *"throwing stones"* (top). From a more visual perspective, humans are able to effectively interact with the environment by relying on domain invariant visual representations. This enables a robust generalization, for instance across dog shapes, silhouettes, backgrounds, etc (bottom). Images source: ChatGPT5.2.

models to the exploit of spurious correlations—such as background motion, camera dynamics, or filming style— rather than the underlying causal structure of actions [38]. For example, a system trained on indoor surveillance footage may fail outdoors due to changes in lighting and scene layout, or a model trained on a third person perspective may fail with egocentric videos [75]. Addressing these failures is essential for deploying reliable video systems in

Figure 1.2: Illustration of different applications that could potentially benefit from highly generalizable video search and retrieval systems, including video indexing (top left), content moderation (top right), surveillance and security systems (bottom left) or assistive technologies (bottom right). Image source: ChatGPT5.2.

real-world and safety-critical settings, including human–robot interaction [41], healthcare [87], and intelligent surveillance, where incorrect predictions can lead to significant consequences.

These challenges motive this thesis, which focuses on addressing generalization limitations in video understanding, with particular emphasis on video search and retrieval. Generalization, however, is a multifaceted concept that encompasses several distinct but interrelated challenges. Following the literature [85, 97], this thesis frames generalization along various dimensions, with a particular focus on **Out-of-Distribution (OOD) Generalization** [113] and **Knowledge Transferability** [18, 103]. OOD generalization concerns a model's ability to maintain performance when the test distribution differs structurally from the training distribution [16]. Or in other words, focusing on *generalization to other environments within the same task*. In the context of video search and retrieval, such shifts may arise from changes in visual appearance—e.g., domain, environment, camera viewpoint, or recording conditions— or from shifts in the linguistic modality—where user queries may be more ambiguous or under-specified compared to the curated training captions— [171]. Robust OOD generalization therefore requires learning representations that capture task-relevant invariances across both

visual and linguistic dimensions, rather than overfitting to dataset-specific traits [15].

Knowledge transferability addresses another aspect of generalization, which can be thought of as *generalization to other tasks*. More specifically, instead of focusing on robustness to distributional shifts (OOD generalization), this concerns to how effectively the knowledge extracted by large-scale pretraining—e.g., CLIP [118], BLIP [130]— can be reused and adapted to new downstream tasks with limited additional supervision [31]. This property is particularly important in video understanding, where collecting densely annotated data is expensive and often infeasible. Models that exhibit strong transferability can leverage pre-trained visual or multi-modal representations to support new tasks, or datasets efficiently, enabling scalable and sustainable deployment.

Accordingly, this thesis investigates robust video search and retrieval along three technical axes:

- **Visual OOD Generalization**: *How can models learn temporal representations that remain stable across environments, viewpoints, and recording conditions?* (Chapter 2)
- **Linguistic OOD Generalization**: *How can models mitigate biases in textual queries, preventing linguistic ambiguity from degrading the overall performance on realistic search and retrieval scenarios?*(Chapter 3)
- **Knowledge Transferability**: *How can existing video representations be reused efficiently to support new tasks with minimal additional supervision?* (Chapter 4)

Goal of this thesis: Overall, the goal of this thesis is to enable models to not merely match the patterns and traits of the data seen during training, but rather to extract patterns and knowledge that naturally generalize to diverse environments and contexts, as well as constituting an effective knowledge base that allow the rapid learning of new tasks, similarly to what humans do.

With this goal in sight, in this thesis we propose three models that address different aspects of generalization and transferability in the context of video understanding. Concretely, these focus on improving the robustness to different visual domain shifts, mitigating existing linguistic biases, and efficiently transferring the knowledge extracted by large-scale VLMs for different video understanding tasks such as VMR. Concretely, in Chapter 2 we introduce a model that leverages a semantic adversarial domain adaptation framework to learn domain invariant video representations, allowing a robust

generalization to new unseen environments. In Chapter 3 we identify a critical linguistic bias in existing video-moment grounding datasets, stemming for their reliance on full captions as textual queries. To address this, we propose a semi-automatic pipeline to generate more realistic, potentially under-specified textual queries, and introduce a method that mitigates such linguistic biases, demonstrating improved grounding capabilities under such more realistic scenarios. Finally, in Chapter 4, we focus on the transferability of vision-language features to efficiently learn related tasks. We propose a lightweight fine-tuning mechanism for large-scale VLMs, enhancing their ability to generalize when only a limited number of training samples are available.

1.1 Contributions and thesis outline

This thesis is organized into four chapters. The first three chapters introduce the core contributions while the final chapter summarizes the main contributions of this thesis, discussing their main limitations, introducing various potential future research lines, and discussing different ethical and economical implications. Below we briefly summarize each of the chapters:

- **Chapter 2 - SADA: Semantic Adversarial Domain Adaptation:** TAL is a complex task that poses relevant challenges, particularly when attempting to generalize on new—unseen— domains in real-world applications. These scenarios, despite realistic, are often neglected in the literature, exposing these solutions to important performance degradation. In this work, we tackle this issue by introducing, for the first time, an approach for Unsupervised Domain Adaptation (UDA) in sparse TAL, which we refer to as Semantic Adversarial unsupervised Domain Adaptation (SADA). Our contributions in this chapter threefold: (1) we pioneer the development of a domain adaptation model that operates on realistic sparse action detection benchmarks; (2) we tackle the limitations of global-distribution alignment techniques by introducing a novel adversarial loss that is sensitive to local class distributions, ensuring finer-grained adaptation; and (3) we present a novel set of benchmarks based on EpicKitchens100 and CharadesEgo, that evaluate multiple domain shifts in a comprehensive manner.
- **Chapter 3 - Beyond Caption-based Queries:** Current VMR models are trained on videos paired with captions, which are written by annotators after watching the videos. These captions are used as textual queries—which we term **caption-based queries**. This

annotation process induces a visual bias, leading to overly descriptive and fine-grained queries, which significantly differ from the more general **search queries** that users are likely to employ in practice. In this work, we investigate the degradation of existing VMR methods, particularly of DETR architectures, when trained on caption-based queries but evaluated on search queries. For this, we introduce three benchmarks by modifying the textual queries in three public VMR datasets—i.e., HD-EPIC, YouCook2 and ActivityNet-Captions. Our analysis reveals two key generalization challenges: (1) A language gap, arising from the linguistic under-specification of search queries, and (2) a multi-moment gap, caused by the shift from single-moment to multi-moment queries. We also identify a critical issue in these architectures—an active decoder-query collapse—as a primary cause of the poor generalization to multi-moment instances. We mitigate this issue with architectural modifications that effectively increase the number of active decoder queries.

- **Chapter 4 - SDST: Sparse-Dense Side-Tuning:** VMR often involves both Moment Retrieval (MR) and Highlight Detection (HD) based on textual queries. For this, most methods rely solely on final-layer features of frozen large pre-trained backbones, limiting their adaptability to new domains. While full fine-tuning is often impractical, parameter-efficient fine-tuning—and particularly side-tuning (ST)—has emerged as an effective alternative. However, prior ST approaches this problem from a frame-level refinement perspective, overlooking the inherent sparse nature of MR. To address this, we propose the Sparse-Dense Side-Tuner (SDST), the first anchor-free ST architecture for VMR. We also introduce the Reference-based Deformable Self-Attention, a novel mechanism that enhances the context modeling of the deformable attention—a key limitation of existing anchor-free methods. Additionally, we present the first effective integration of InternVideo2 backbone into an ST framework, showing its profound implications in performance.

More specifically, Chapters 2, 4 present and discuss the contributions and findings of two papers published in the main track of top-tier computer vision venues (WACV 2025, ICCV 2025).

- David Pujol-Perich, Albert Clapés, and Sergio Escalera. "SADA: Semantic adversarial unsupervised domain adaptation for Temporal Action Localization." *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2025.

- David Pujol-Perich, Sergio Escalera, and Albert Clapés. "Sparse-dense side-tuner for efficient video temporal grounding." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2025.

Moreover, Chapter 3 corresponds to the contributions and findings of the paper *Beyond Caption-Based Queries for Video Moment Retrieval*, resulting from a collaboration with Prof. Dima Damen and Dr. Michael Wray from the University of Bristol, along with Prof. Sergio Escalera and Dr. Albert Clapés. This work is currently under review at CVPR 2025.

Chapter 2

SADA: Semantic adversarial unsupervised domain adaptation for Temporal Action Localization

As discussed in the previous chapter, building Video Understanding models that remain robust under visual domain shifts is essential for deployable deep learning systems. In real-world applications, videos are recorded under changing conditions—e.g., different backgrounds, lighting, or viewpoints— which can significantly degrade performance if models overfit to domain-specific traits. Addressing this challenge requires extracting representations that are invariant to superficial details while still capturing the underlying temporal dynamics.

In this chapter, we explore this idea by proposing a domain adaptation framework for Temporal Action Localization that performs semantics-informed adversarial alignment. To the best of our knowledge, at the time of this writing, this constituted the first method specifically designed for domain adaptation on this task. Our approach aligns features across domains while respecting the semantics of each of the actions, resulting in a more robust performance when transferring from one domain to another.

2.1 Introduction

Recent advances in the field of video understanding have played a critical role in the surge of novel video-based applications— e.g., video indexing, summarization or recommendation. A critical task of this field is *Temporal Action Localization* (TAL), which involves identifying actions in a video consisting of both their time intervals and action categories. This is particularly difficult given the inherent variabilities of videos. These can be presented, among others, in the form of *appearance variability*— e.g., different kitchens and/or lighting conditions—, *acquisition variability*— e.g., different recording devices— or *viewpoint variability*— e.g., first- or third-person. All

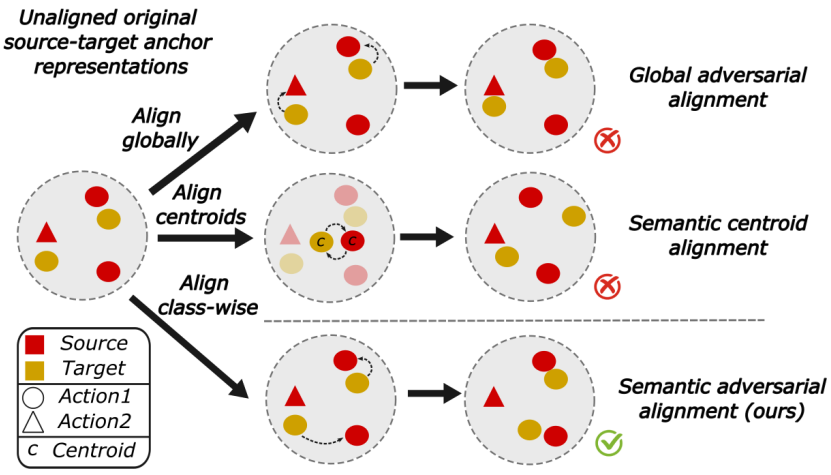


Figure 2.1: Illustration of the differences between the two most similar domain-adaptation methods [39, 83], and our proposal, *SADA*. For this, we present a simple scenario with various anchor embeddings of different actions (identified by shapes) and domains (identified by colors). In this scenario, [39] (upper row) aligns embeddings in a class-agnostic manner, making it liable to aligning domain embeddings of unmatching action labels. [83] (middle row) computes class-wise mean centroids, and aligns them across domains, but as shown, minimizing their distance does not yield a proper adaptation. *SADA* (last row) improves [39] by aligning class-wise distributions, yielding the correct alignment by not aligning unmatching anchors.

these prompt a certain degree of confusion between similar actions to be discriminated.

Traditionally, fully supervised methods attempt to address this issue by leveraging enough training data to cover all the possible sources of variability. Unfortunately, this becomes virtually impossible when dealing with realistic scenarios. This compels these methods to operate under the influence of unseen data variations— i.e., *domain gaps*—, which exposes them to a considerable decline in performance. Overcoming this typically involves relabeling data from the new domain, so as to retrain and adapt the model. Unfortunately, this approach is impractical due to the considerable time and resource consumption involved, a challenge exacerbated when dealing with high-dimensional inputs like videos.

Unsupervised Domain Adaptation (UDA) has recently become a hot topic given its potential to leverage unlabelled data to mitigate this domain-induced degradation [34, 63, 151]. Despite its considerable success

in image-based tasks [52] the application of UDA to video understanding remains underexplored. In fact, to the best of our knowledge, no prior work addresses UDA for TAL setups. The closest proposal is SSTDA [95], which approaches the problem of action segmentation. This focuses on making per-frame action predictions (i.e., *dense TAL*), tackling datasets where no concurrent actions take place [21, 27]. This approach inevitably requires additional smoothing techniques to preserve temporal coherence. This motivates the focus of this chapter on the more general *sparse TAL* problem—i.e., segment-level predictions— avoiding the need for additional losses, while intrinsically adapting to multi-label scenarios.

Consequently, in this chapter, we propose the first UDA method for sparse multi-label detection on TAL, which we name *Semantic Adversarial unsupervised Domain Adaptation*, or *SADA* for short. Concretely, our proposal specifically builds upon an anchor-based architecture given their recent success on sparse TAL [144, 167]. Thus, our goal is to minimize the discrepancy between anchor representations of a labeled source domain and an unlabelled target domain. These anchors are extracted with a multi-resolution architecture [167] that we couple with a novel adversarial loss that improves the limitations of existing UDA works. Concretely, existing works normally align domain distributions globally [34], applying adversarial methods on the feature embeddings regardless of the action class they represent. As we will show, the coarse alignment of anchors of different action classes with even *background* (no-action) anchors can hurt performance. We propose instead to first use pseudo-labeling [158] to assign an action or background class to each anchor representation. With this, we can factorize the global alignment loss [34] into independent per-class and background distribution alignments. This results in a more sensitive alignment strategy, less prone to *semantic feature misalignment*— i.e., semantically meaningless alignments across domains— and as we will show, better performing.

Assessing the effectiveness of UDA methods for video understanding is a challenging, still unresolved task. Existing proposals on action segmentation [95] follow a subject-based strategy where they aim to adapt a model to new unseen subjects. Here we refer to *subject* as a person appearing in a video. Nevertheless, little data is normally available from a single subject, which inevitably requires grouping several of them for training. This allows the model to generalize over the subject variability under study, making it unsuitable for domain adaptation. To address these limitations, we draw inspiration from the work of [146] on action recognition and investigate the impact of *viewpoint domain shifts* on sparse TAL in CharadesEgo [81]. However, we contend that a more comprehensive evaluation necessitates

setups with more controllable shifts. For this, we also propose a suite of 6 new setups based on EpicKitchens100 [126] which study the effect *appearance* and *acquisition* domain shifts. These benchmarks demonstrate that *SADA* mitigates the performance degradation, improving by a large margin the existing fully supervised (namely *source-only*) and UDA-based proposals. In short, our main contributions are:

1. We propose for the first time an UDA method suitable for sparse detection scenarios on TAL.
2. We introduce a novel adversarial loss that factorizes standard global alignment into independent class- and background-wise alignments (see Fig. 2.1).
3. We present new benchmarks to test sparse detection scenarios when facing 7 different domain shifts, improving the state-of-the-art in all of them.

2.2 Related work

Temporal Action Localization. At the time of this writing, most of the literature on the task of *Temporal Action Localization* follows a traditional *source-only* approach. In other words, they restrict the models' visibility solely to a training domain, while other domains seen during testing are not available. These works can be categorized as follows: **(1) Anchor-based methods** [49, 72, 111, 112, 117, 120, 144, 167] propose a two-stage pipeline, consisting of a proposal generation and classification. The first applies heuristic methods—e.g., uniform sampling [49, 144] or action boundaries' grouping [70, 104]— to generate a dense set of proposals—i.e., temporal segments. In the second stage, they leverage a learnable classifier to predict the corresponding action class and localization offsets of every anchor. Our work falls into this category motivated by the recent success of these methods achieving state-of-the-art results in many TAL benchmarks [55, 92, 126]. **(2) Anchor-free methods** [65, 60, 112, 69] avoid this two-stage approach making per-frame predictions of their corresponding action labels. These methods, however, often suffer from a tendency towards over-segmentation given the potential discrepancy between neighboring frames. Consequently, they require often complex smoothing techniques to improve the boundary predictions [95]. **(3) Query-based methods** [121, 114, 133] recently emerged as an alternative paradigm that follows the principles presented by [94]. This approach exploits the use of a Transformer encoder-decoder architecture [68] to learn a fixed

small set of queries given refined video features, each identifying one potential action segment. Intuitively, this results in a non-heuristic-based proposal generation. This comes with the limitation of an increased rigidity, as the number of proposals needs to be fixed beforehand.

Unsupervised Domain Adaptation. Domain Adaptation techniques emerge as an effective solution to bridge the gap between data collected from a source and a target distribution, respectively. A large suite of approaches has been proposed to perform this alignment between labeled and unlabeled domains— e.g., discrepancy minimization [57, 142] or entropy minimization [13, 124]. Arguably, nowadays the most popular approach is based on adversarial training [34, 66, 109, 153, 151]. These learn domain-invariant embeddings [96] by training in a min-max fashion a domain classifier to discern if samples come from the source or the target domain. Despite convenient, the simplicity of these methods often degrade the quality of the alignment [158], as they potentially align embeddings of source and target domain that represent different semantic information— e.g., different class labels. Few works have been proposed to do this alignment in a more sensitive way [158]. Works like [83, 153], for instance, reduce the distance of per-class centroids, normally computed as the mean feature embeddings of a given class. Its effectiveness, however, relies on the assumption that the data is distributed somewhat homogeneously around the center, as otherwise, the centroids are not necessarily meaningful. In our work, we couple the advantages of both adversarial domain adaptation and semantic alignment and propose for the first time a pure adversarial semantic loss that yields domain invariant representations in a semantically meaningful way, without making explicit assumptions of the distributions (see Fig. 2.1).

Unsupervised Domain Adaptation for TAL. Despite the considerable success of UDA methods, their applicability has been mostly restricted to image-based scenarios such as image classification [54, 34, 36, 63] or object detection [74, 164]. Much less attention has been dedicated to video-based applications such as action recognition [86, 77, 102] or spatio-temporal action segmentation [93, 161]. To the best of our knowledge, at the time of this writing, there is no direct comparison with our work focusing on UDA for *sparse* TAL. The closest work is SSTDA [95] that applies UDA for Action Segmentation. SSTDA proposes the use of two global-distribution-based auxiliary tasks to jointly align cross-domain feature spaces. Unlike our proposal, their work falls into the category of anchor-free, making per-frame action predictions. This restricts its applicability to action segmentation scenarios, where current datasets [21, 27] are designed to deal with frame-based single-action classification. In our work, we overcome this limitation by

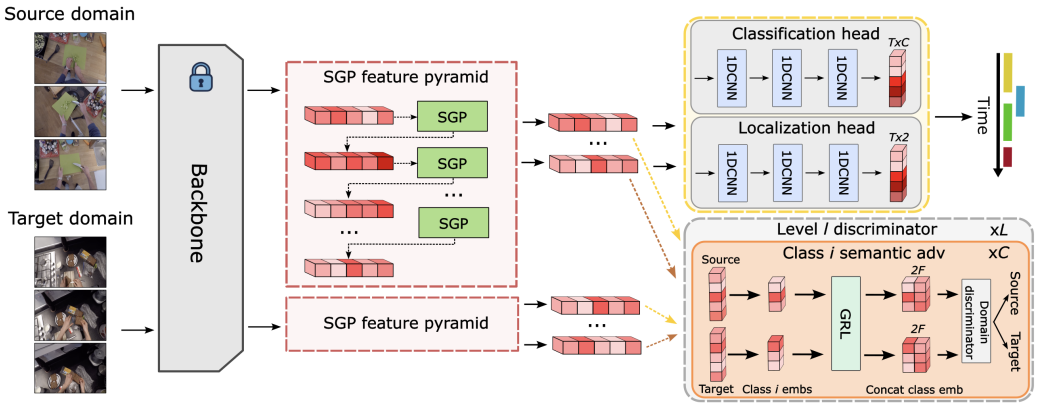


Figure 2.2: Overview of the main model architecture of SADA. This takes as input videos from a Source and a Target domain, which are both fed to a shared multi-resolution feature extractor pyramid. The output embeddings of both of these domains are then aligned using the semantic alignment loss, *SADA*. This is done with a level and class-wise domain discriminator of the filtered embeddings, based on GT information and pseudo labels, for the source and target domains, respectively. Finally, the resulting domain invariant representations of the source domain are used to train a classification and localization head to learn the underlying task.

leveraging an anchor-based architecture that enables a natural adaptation to more realistic multi-label scenarios.

2.3 Method

2.3.1 Problem definition and notation

In this chapter, we address the problem of unsupervised domain adaptation for TAL. For this, we define a source domain $\mathcal{S}$ and a target domain $\mathcal{T}$. Domain $\mathcal{S}$ consists of $N_{\mathcal{S}}$ labeled input videos $\{(V_k^{\mathcal{S}}, Y_k^{\mathcal{S}})\}_{k=1}^{N_{\mathcal{S}}}$, where each video $V_k^{\mathcal{S}}$ is a sequence of T frames $(X_{k,1}, \ldots, X_{k,T})$ with $X_{k,t} \in \mathbb{R}^{H \times W \times C}$. Here $Y_k = \{(b_{k,i}, e_{k,i}, c_{k,i})\}_{i=1}^{G_k}$ contains the begin, end, and class actions of all the ground-truth (GT) segments G_k of the k-th video, respectively. The target domain $\mathcal{T}$ is similar to $\mathcal{S}$ but lacks the GT information. Concretely, it consists of $N_{\mathcal{T}}$ unlabeled input videos $\{V_k\}_{k=1}^{N_{\mathcal{T}}}$. Our goal is to train a model that can identify the action segments, including both segment coordinates and action labels, in videos from domain $\mathcal{S}$, while minimizing the performance degradation on the unlabeled domain $\mathcal{T}$.

2.3.2 Framework overview

We propose a model based on a feature pyramid and an anchor-based classification and localization head (see Fig. 2.2). This architecture is coupled with a novel *semantic adversarial loss* that aligns the anchor embeddings across domains $\mathcal{S}$ and $\mathcal{T}$ in a semantically meaningful way. More in detail, the model takes as input two videos from domain $\mathcal{S}$ and $\mathcal{T}$, respectively. The model first processes the raw input videos using a frozen pre-trained video backbone. The resulting embeddings of both domains are then passed through a shared SGP pyramid [167] that outputs a set of multi-resolution anchor embeddings for each of the predefined resolution levels. The main goal of our model is to make these anchor embeddings domain invariant. For this, we introduce a level-wise *semantic adversarial loss* that learns in an adversarial manner to align the embeddings of both domains belonging to a given action class i at a given resolution level l. Recall that GT information is only available for domain $\mathcal{S}$, therefore we rely on the use of pseudo labeling techniques [158] to infer the *probable* class labels of the data from domain $\mathcal{T}$. Finally, we use the domain invariant anchors of domain $\mathcal{S}$ to train a classification and localization head that learns the underlying tasks in a standard supervised fashion. In short, this permits to learn a classification and localization head that minimizes the decline of performance when applied to the unseen domain $\mathcal{T}$.

2.3.3 Backbone and SGP pyramid

Our model first takes two input videos $V^{\mathcal{S}}$ and $V^{\mathcal{T}}$ of both domains $\mathcal{S}$ and $\mathcal{T}$. For simplicity, both videos $V^{\mathcal{S}}$ and $V^{\mathcal{T}}$ have length T, which we enforce using padding. The method then processes the two videos applying a frozen pre-trained backbone— e.g., I3D [50] or Slowfast [88]. This permits to extract, in an effective way, temporal cues of the video into a set of refined video features. These embeddings are then fed to an SGP feature pyramid [167] which combines the use of SGP blocks and the progressive downsampling of the temporal length by a ratio of 2. This outputs a set of multi-resolution anchor embeddings $Z^{\mathcal{S}} = \{Z_l^{\mathcal{S}}\}_{l \in L}$ and $Z^{\mathcal{T}} = \{Z_l^{\mathcal{T}}\}_{l \in L}$, for the two domains, respectively. Here L denotes the set of predefined resolution levels and $Z_l \in \mathbb{R}^{T_l \times F}$ the anchor embeddings of level l of a given domain. Concretely, this permits to obtain embeddings for a set of uniformly sampled anchors at each of the $l \in L$ resolution levels. The use of a multi-resolution model is favorable for naturally adapting to different action lengths and abstraction levels.

2.3.4 Classification and localization head

To learn the underlying TAL task, we train in a fully supervised manner a classification and localization module with the labeled source domain $\mathcal{S}$. Due to the anchor-based nature of our model, we first require a matching strategy between the set of candidate anchors to the actual GT segments. For this, we follow a center sampling strategy [90, 167]. In other words, for a given level l, we define an anchor as *action anchor* if the time instant t that it represents is near the center of an action. All the rest are marked as *background anchors*. We define $\mathcal{B}_l$,$\mathcal{E}_l$ and $\mathcal{C}_l$ as the begins, ends and action classes of their matching GT segments. We identify *background anchors* with action label 0. With this, we design a classification head $H_{cls} : \mathbb{R}^{T_l \times F} \rightarrow \mathbb{R}^{T_l \times C}$ that maps each of the anchor embeddings to their class distribution. More specifically, we model this as a sequence of 1D convolutions, and train it using a sigmoid focal loss [61]:

$$\mathcal{L}^l_{SFL} = SFL(H_{cls}(Z^{\mathcal{S}}_l), \mathcal{C}_l). \tag{2.1}$$

Similarly, we model a localization head $H_{loc} : \mathbb{R}^{T_l \times F} \rightarrow \mathbb{R}^{T_l \times 2}$ identically as H_{cls}, which predicts the begin-end offsets. We thus define the localization loss as a standard mean squared error (MSE) loss over the *action anchors* only:

$$\mathcal{L}^l_{loc} = MSE(H_{loc}(Z^{\mathcal{S}}_{l_+}), (\mathcal{B}_{l_+} || \mathcal{E}_{l_+})), \tag{2.2}$$

where l_+ refers to the filtered *action-anchor* representations from the GT of the l-th level only, and $||$ to the concatenation operation. This yields the final task loss defined as:

$$\mathcal{L}_{task} = \lambda_{cls} \sum_{l \in L} \mathcal{L}^l_{SFL} + \lambda_{loc} \sum_{l \in L} \mathcal{L}^l_{loc}. \tag{2.3}$$

Here λ_{cls} and λ_{loc} are two tunable hyperparameters.

2.3.5 Our proposal: Semantic adversarial multi-resolution alignment

One of the main contributions of this chapter is the design of a novel adversarial-based loss that we name *SADA* loss, which attempts to overcome the limitations of the extensively used *global adversarial loss* [34]. Traditional adversarial domain adaptation relies on the idea of designing a domain classifier that learns to identify the domain that each of the embeddings belongs to.

The rest of the model learns concurrently the opposite objective which results in the learning of domain invariant representations[34]. While this approach has been shown to be effective in other fields— e.g., image classification or object detection— we find that its performance in more challenging video understanding setups like TAL presents important challenges. One of the main issues, as argued by [158], is that this loss often suffers from *feature misalignment* which greatly declines its effectiveness. This refers to the cases where these methods align embeddings of non-matching class labels— i.e., aligning embeddings of domain $\mathcal{S}$ of an action i with embeddings of domain $\mathcal{T}$ of a class j. This issue is further exacerbated in TAL given the noise induced by the alignment of the many *background anchors* with the *action anchors*.

Local adversarial alignment: To fix this *feature misalignment* in realistic scenarios like TAL, we propose an alternative adversarial loss formulation that provides a finer-grained alignment. This loss first attempts to perform a local class-aware alignment. This is, for every given resolution level l, we group the *action anchors*— those matched with a GT action— of $\mathcal{S}$ and $\mathcal{T}$ according to their action label i. This is straightforward for domain $\mathcal{S}$ as we have the GT information. In contrast, for domain $\mathcal{T}$, we use a hard-pseudo labeling strategy [158] that classifies a given embedding as class i if this is the highest-confidence score of the predicted class distribution, and this is above a threshold α. Formally we define the pseudo-label of an anchor z as:

$$\hat{c}_z = \begin{cases} \underset{i}{\text{argmax}}\, P_l[z,i] & \text{if } P_l[z,i] > \alpha \\ 0 & \text{otherwise,} \end{cases} \tag{2.4}$$

where $P_l = H_{cls}(Z_l^{\mathcal{T}}) \in \mathbb{R}^{T_l \times C}$ are the predicted class probabilities of the anchors. Notice that we mark with class 0 the *background anchors*, which are not assigned to any action class. From this, we obtain the newly grouped embeddings of source and target domain of class i on level l

$$A_i^l = \{Z_l^{\mathcal{S}}[z] \,:\, c_z = i\}_{z \in T_l}, \tag{2.5}$$

$$B_i^l = \{Z_l^{\mathcal{T}}[z] \,:\, \hat{c}_z = i\}_{z \in T_l}, \tag{2.6}$$

for $A_i^l \in \mathbb{R}^{T_{l,i} \times F}$, $B_i^l \in \mathbb{R}^{T_{l,i} \times F}$. Also, c_z is the GT action label of anchor z and $\hat{c}_z$ is its computed pseudo-label from Eq. 2.4. We then adversarially train a single domain classifier $D : \mathbb{R}^{2F} \to \{0,1\}$ to identify the domain of each of

these embeddings using a binary cross entropy (BCE) loss:

$$\mathcal{L}^l_{local} = \sum_{i=1}^{C} (\mathcal{L}_{\text{BCE}}(D(A^l_i \,||\, E_i), d_{\mathcal{S}})) + (\mathcal{L}_{\text{BCE}}(D(B^l_i \,||\, E_i), d_{\mathcal{T}})), \tag{2.7}$$

where $d_{\mathcal{S}}$ and $d_{\mathcal{T}}$ are the domain labels. We then introduce a Reverse Gradient Layer (GRL)[34] before the discriminator D to invert the gradients sign, creating a min-max game where the feature extractor learns to *confuse* the discriminator. We condition the discriminator to class i using a learnable class embedding $e_i \in \mathbb{R}^F$ that we replicate for every selected anchor into an embedding E_i.

Local and global alignment (*SADA*): Eq. 2.7 aims solely to align the *action anchors*— which are classified as one of the C classes— but *what happens with the background embeddings that fall below the threshold* α*?* In this case, the loss ignores their influence, yielding only partial alignment.

To overcome this issue, we propose our final *SADA* loss which attempts to combine the best of both *global alignment loss* [34] and Eq. 2.7. For this, we introduce a new loss term for the *background anchors* as follows:

$$\mathcal{L}^l_{bkg} = \mathcal{L}_{\text{BCE}}(D(A^l_0 \,||\, E_0), d_{\mathcal{S}}) + \mathcal{L}_{\text{BCE}}(D(B^l_0 \,||\, E_0), d_{\mathcal{T}}), \tag{2.8}$$

where again A^l_0 and B^l_0 are the selected *background anchors*, and $E_0 \in \mathbb{R}^F$ is the learnable *background* embedding. Coupling Eq. 2.7 and Eq. 2.8 yields the final formulation of our proposed loss, combining *local* (class-wise) alignment with the *background* anchors alignment. Formally:

$$\mathcal{L}_{sada} = \sum_{l \in L} \lambda_l \, (\mathcal{L}^l_{local} + \mathcal{L}^l_{bkg}), \tag{2.9}$$

where λ_l is a hyper-parameter that modulates the importance of level l on the final alignment loss.

2.3.6 Training

During training, we formulate the final loss as a min-max game where the main model architecture is optimized over the classification and localization loss while maximizing the adversarial loss. In parallel, the discriminator model D attempts to minimize the discriminator loss only. Formally,

$$\mathcal{L} = \lambda_{task}\mathcal{L}_{task} + \lambda_{sada}\mathcal{L}_{sada}. \tag{2.10}$$

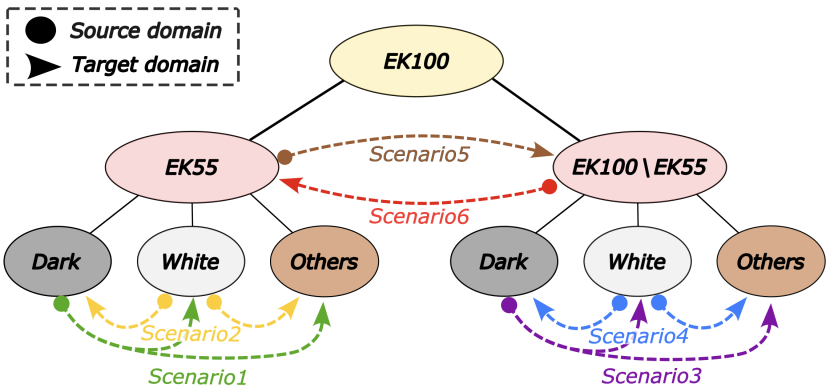


Figure 2.3: Overview of the 6 proposed experimental setups for EpicKitchens100. Concretely, S1 and S2 evaluate the videos from the original EK55. They define the dark-counter and white-counter kitchens as Source, respectively, and the rest as Target. S3 and S4 are similar except that they consider only the *newest* videos from EK100. S5 and S6 use the *old* videos as Source and the *new* videos as Target, and vice versa.

Note again $\mathcal{L}_{task}$ is optimized with domain $\mathcal{S}$ while $\mathcal{L}_{sada}$ promotes the alignment between both domains $\mathcal{S}$ and $\mathcal{T}$. Moreover, λ_{task} and λ_{sada} are tunable parameters.

2.4 Benchmarks

Evaluating domain adaptation-based methods in the context of video understanding is a challenging issue that requires the definition of a reasonable domain gap and identifying a sufficiently large set of intersecting action classes. Dividing existing datasets into different domains that comply with these conditions often restricts the amount of data to learn and adapt. SSTDA [95] approaches this problem on GTEA [21] and Breakfast [27] by defining a subject-based partitioning where they aim to adapt the model to new unseen subjects. However, as little data is available from a single subject in those datasets, they group several users for training. This allows the model to generalize over the subject variability under study, making it unsuitable to test for domain adaptation. Closely related to our work, [172, 100] propose several UDA scenarios for video classification based on EpicKitchens100 [126]. These define 3 domains based on the data of 3 different kitchens, thus performing cross-kitchen evaluation. This limits even further the amount of data in each domain— i.e., between 15 to 29 videos per domain.

2.4.1 EpicKitchens100

To overcome these limitations, we first propose a new set of 6 different scenarios *(S1, . . . , S6)* for sparse TAL based on EK100 [126], (see Fig. 2.3). EK100 presents an ideal base for our tasks as it has become a gold standard to evaluate complex sparse detection scenarios on long egocentric videos (up to 45 minutes). We identify two domain gaps in this dataset: an *acquisition domain shift* that results in the differences of lighting and camera conditions when extending EK55 [126] into its new version EK100 [126]; and an *appearance domain shift* based on the different colors of the kitchen counters. This permits to define a rich set of benchmarks that provide a more fine-grained and comprehensive evaluation. Importantly, this strategy is also suitable for single-source settings which do not allow an easy generalization over the shift under study. For example, if a model is trained with dark-counter kitchens only, it cannot easily generalize to white-counter kitchens. Below we describe in more detail the aforementioned shifts:

Studying acquisition shifts: The first domain shift that we identify in the EK100 [126] is what we call the *acquisition shift*. This shift refers to the changes induced by differences in the acquisition conditions of the videos. In the case of EK100 [126] this results from the extension of the original dataset EK55 [75]. The original dataset, concretely, was formed by 55 hours of non-scripted videos, and nearly 40K action segments. Hence, we find that this presents a suitable setup to define one domain as the *old* videos— recorded in the original EK55 [75]— and the other formed by the *new* videos— recorded for the extended version. As argued by [126] this results in several important domain gaps that are captured by these data splits:

- **Changes in the acquisition devices:** The *new* videos that were recorded during the extension relied on a newer camera device. Importantly, this camera incorporates camera stabilization techniques. Fig. 2.4 visually depicts the improvement in the stability of *new* over *old* videos.
- **Lighting conditions:** Given the changes in the hour of the recording, these domains also present differences in the lighting conditions of the videos.

With this, given the two splits between *EK55* and *{EK100 \EK55}* videos, we propose 2 setups of *EK55* $\rightarrow$ *{EK100 \EK55}* and *{EK100 \EK55}* $\rightarrow$ *EK55*. These measure the adaptability to different acquisition conditions— i.e., lighting and camera conditions.

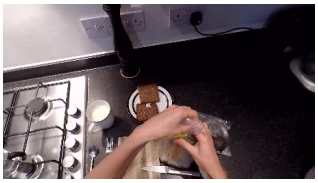
(a) *Old* videos

(b) *New* videos

Figure 2.4: Visualization of the differences in camera stability between *old* and *new* videos. These images were obtained with differences of 3 frames, which correspond to a period of 0.1 seconds. We highlight with a red circle in 2.4a a reference in the images to visualize the high instability of the video.

Studying appearance shifts: Another important domain shift that we are interested in capturing is one induced by changes in the background. In this regard, there are several possibilities for this, but we find that the vast majority allow only a vague intuition of the domain shift that is under study. For instance, [126] argues that in the extension, several kitchens might have some changes in the furniture or the order of the main kitchen utensils. But, *which kitchens really contain these changes?* Not understanding this issue in sufficient depth results in an obscure evaluation, which we try to avoid in this work. For this reason, we identify a clear, understandable domain gap. This is the color of the kitchen counters. This is an essential part of the background information, and thus, we hypothesize that adapting to this factor is also critical to obtain high-performing models. In this regard, we split the data into three different domains: *dark*, *white*, and *other* types of kitchen counters. This provides a clear domain gap that we can visualize and thus, understand (see Fig. 2.5).

Intuitively, this permits to define the following 2 scenarios. Firstly, we can define a scenario for the black-counter kitchens as the source domain, while the rest are the target domain. Similarly, we define another scenario where the white-counter kitchens are the source domain, while the rest are the target domain. Notice, however, that this mixes acquisition conditions as we do not differentiate *old* and *new* videos. To ensure that this does not impede a clear understanding of the scenario, we consider *old* and *new* videos, independently. This results in the 4 different setups of single-source domain and multi-target domain— i.e., *dark* $\rightarrow$ *white* and *other* kitchens; and *white* $\rightarrow$ *dark* and *other* kitchens, for *old* and *new* videos, respectively.



(a) Dark kitchen

(b) White kitchen

(c) Other kitchen

Figure 2.5: Visualization of the 3 different appearance-based domain shifts resulting from the split of dark-counter kitchens, white-counter kitchens, and finally all the other kitchens.

Scenario	Domain	Split	# vids	# segs	Avg len (s)	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10
S1	Dark(55)	Train	126	8336	3.11	1982	1611	1208	868	561	768	315	492	243	288
		Val	35	2944	3.05	699	603	446	273	194	271	131	124	92	111
S2	White(55)	Train	124	9995	2.95	2453	2398	1739	963	658	439	477	321	352	195
		Val	31	1595	3.75	385	313	184	189	111	101	78	120	56	58
S3	Dark(100)	Train	45	5487	2.34	1576	1209	672	429	314	453	262	186	226	160
		Val	13	1504	1.95	430	320	177	127	80	113	98	11	69	79
S4	White(100)	Train	49	6895	2.26	2236	1889	818	579	417	87	313	109	245	202
		Val	19	2328	2.79	705	553	377	184	135	67	107	73	83	44
S5	All(55)	Train	342	25634	3.12	5529	6449	1762	2717	1708	1013	3828	760	1109	759
		Val	90	5510	3.29	1352	1110	451	587	400	170	715	209	284	232
S6	All(100)	Train	150	20512	2.10	930	6347	1319	788	1173	5021	1715	545	459	2215
		Val	44	6526	2.27	1573	540	395	306	1964	241	228	291	812	176

Table 2.1: Analysis of the Source domains of each of the proposed scenarios.

All in all, we stress that all our proposed scenarios provide sufficiently even splits to design single-domain setups where we can ensure that there is no easy generalization of the target domain by means of the source training data.

Statistics: To gain additional insights, in Tab. 2.1 and Tab. 2.2 we present the relevant statistics of the 6 proposed scenarios—i.e.,., number of videos, the number of segments, the average length of these segments, and finally the class-wise number of segments to depict the overall distribution of actions aforementioned statistics for the source and target domain of each scenario, respectively. Observe that thanks to our careful experimental setup, we are able to define scenarios with considerably large source and target domains. This contrasts with other experimental setups like that proposed by [172] which define the source and target domain of a scenario as the videos of a single kitchen, respectively. We find that this, in consequence, yields domains with only 15-28 videos, which we deem insufficient to train state-of-the-art large architectures.

One of the key challenges of this dataset is the presence of 97 different actions, forming a very long tail action distribution. This inherent characteristic is

Scenario	Domain	Split	# vids	# segs	Avg len (s)	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10
S1	All-Dark(55)	Train	216	17298	3.12	3918	4467	1201	1849	940	698	2620	517	617	471
		Val	55	2566	3.56	653	507	180	314	206	78	269	98	160	101
S2	All-White(55)	Train	218	15639	3.23	3131	3996	1104	1754	1269	536	2089	408	788	564
		Val	59	3915	3.10	967	797	350	398	289	114	531	151	164	154
S3	All-Dark(100)	Train	105	15025	2.02	668	4771	1005	562	720	3812	1286	385	273	1543
		Val	31	5022	2.38	1253	413	315	208	1534	172	149	178	635	165
S4	All-White(100)	Train	101	13617	2.01	617	4111	902	543	1086	3132	1136	343	350	1397
		Val	25	4198	1.99	1020	356	260	199	1259	158	184	224	435	103
S5	All(100)	Train	150	20512	2.10	930	6347	1319	788	1173	5021	1715	545	459	2215
		Val	44	6526	2.27	1573	540	395	306	1964	241	228	291	812	176
S6	All(55)	Train	342	25634	3.12	5529	6449	1762	2717	1708	1013	3828	760	1109	759
		Val	90	5510	3.29	1352	1110	451	587	400	170	715	209	284	232

Table 2.2: Analysis of the Target domains of each of the proposed scenarios.

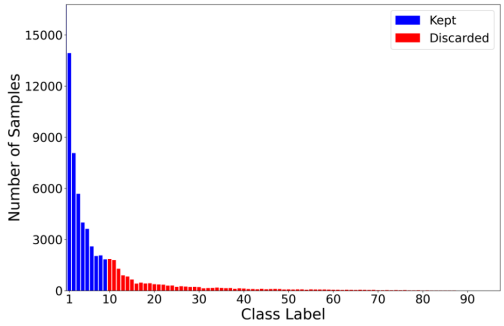


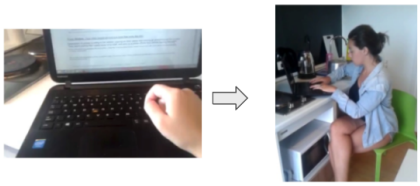
Figure 2.6: Histogram of the number of GT action segments of every class. We also depict in blue the 10 majority classes—which we keep in our setups– while leaving as red the remaining class that we discard to avoid the long-tail action distribution problem.

however an issue that falls beyond the scope of this thesis, given the important additional challenges that this poses in existing domain adaptation-based methods. For this reason, in all our experiments we consider only the 10 majority classes— i.e., *take*, *put*, *wash*, *open*, *close*, *insert*, *turn-on*, *cut*, *turn-off* and *pour*. As shown in Fig. 2.6 this pruning keeps most of the labeled action segments of the original dataset, concretely keeping up to 80% of it (marked in blue in the histogram). Moreover, we find that this design decision does not diminish in a relevant way the challenging nature of this task. Proof of this is that the best-performing methods at the time of this writing, on this dataset— i.e., Actionformer [144] and Tridet [167]– still attained mAP metrics below 30% in all our proposed scenarios.

2.4.2 CharadesEgo

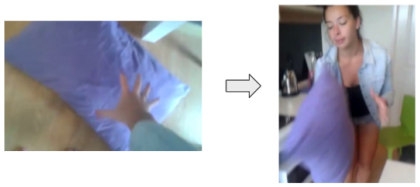
One important aspect of the 6 previous setups is that all the domains share an egocentric viewpoint. For this reason, we evaluate the model degradation under extreme domain shifts caused by major changes in the perspective. Concretely, we leverage CharadesEgo [81], which extends the original third-person videos from Charades [44] into their corresponding egocentric videos. This setup allows us to explore the adaptation of third-person videos to egocentric ones, which often involves an extreme domain shift (see Fig. 2.7). In fact we argue this domain shift can be so extreme that would not be suitable for image-based model benchmarking given that one of the views might lack necessary information that is only contained in the other view. For instance, *a person awakens in bed* could be indiscernible from *a person looking at the ceiling* if only one single frame is considered. Nevertheless, our proposed benchmark deals with video-inputs, which should provide sufficient context from the other frames to infer the true action, making this challenging setup more feasible.

To be more precise on our proposed benchmarking setup, we follow the work of [146] and define the third-person videos as Source domain and the egocentric ones as Target domain. This setup presents an important limitation that we find is critical to make this benchmark viable in this complex task. Concretely, observe that class actions are mostly defined as NLP descriptions, which creates very concrete actions, making it cumbersome to adapt classes across domains. We argue that one of the main reasons is the lack of sufficient data for each of the actions. Consequently, we rely on the originally extended annotations from [81] and propose to use instead the *verbs* of each of the corresponding action labels. This is, for a given action *holding some clothes*, we consider this action with its verb *hold*. One final consideration is that this still results in underrepresented *verb* classes which we fix following a similar approach to that presented by [172] and consider only the 10 majority classes being *take*, *put*, *sit*, *walk*, *hold*, *stand*, *open*, *drink*, *smile* and *laugh*. This still results in a very challenging setup as observed in Tab. 2.5, which indicates that none of the state-of-the-art methods tested attain more than 25% mAP. Moreover, this setup retains 70% of the original annotations.



(a) "A person is typing on a laptop"

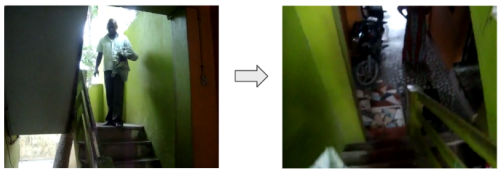
(b) "A person awakens in bed."



(c) "A person picks up a pillow"

(d) "A person pours a drink from the fridge"

(e) "A person is sitting in the pantry holding a mirror. "



(f) "A person is running down the stairs holding a blanket"

Figure 2.7: Comparison in two different scenarios between the egocentric perspective (left) and the third-person one (right).

2.5 Experimental results

2.5.1 Experimental setup

In this section, we present the main experimental results evaluating *appearance* and *acquisition shifts*. All these experiments follow the standard transductive unsupervised DA protocol [18, 51], and report the mean average precision (mAP) at different intersections over union (IOU) thresholds ($10\% - 50\%$).

2.5.2 Implementation details

Here we include all the relevant implementation details that ensure the proper reproducibility of *SADA* in the different setups that we present in this chapter. Note that we report only the hyperparameters of the best model only, which were obtained through a Bayesian optimization process to minimize the downstream loss—i.e., not including the domain adaptation losses– of

Dataset	Scenario	Backbone	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	α	λ_{reg}	λ_{cls}	λ_{sada}
EK100	S1	SF	0.7	0.3	0.9	0.8	0.7	0.0	0.3	1.0	1.0	3.0
	S2	SF	0.7	0.6	1.0	0.7	0.2	0.2	0.3	1.0	1.0	3.0
	S3	SF	0.4	0.8	0.7	0.7	0.9	0.6	0.6	1.0	1.0	1.0
	S4	SF	0.4	0.9	0.1	0.4	0.1	0.6	0.5	1.0	1.0	0.5
	S5	SF	1.0	0.4	0.8	0.7	0.9	0.6	0.5	1.0	1.0	2.0
	S6	SF	0.6	0.1	1.0	0.6	0.2	0.4	0.3	1.0	1.0	2.0
CharadesEgo	-	I3D	0.4	0.8	0.7	0.7	0.9	0.6	0.6	1.0	1.0	1.0

Table 2.3: Summary of the main hyperparameters detailed for each of the presented benchmarks. Here I3D refers to [50] and SF to [88].

the validation split of the Source domain. We follow this approach since by definition we do not have any label information on the Target domain. Nevertheless, we hypothesize that this is a suitable strategy as the goal of our adaptation loss is to *bring both domains closer*. Thus, a set of hyperparameters that performs well on the Source domain should similarly do so on the Target domain. We mitigate the presence of overfitting by using early stopping and a weight decay with a value of 0.05.

All the models that we present are implemented using PyTorch-2.0, CUDA 12.4 and trained for 50 epochs using AdamW [62] optimizer with a learning rate of $1e^{-4}$ with cosine decay and a warm-up phase of 5 epochs on one single NVIDIA GeForce RTX 3090. The training process in all our experiments took between 12 and 24 hours.

Architecturally, all our models define the main feature extractor as a 6-level SGP feature pyramid [167] that follows a max pooling strategy and uses a downsampling rate of 2 and an internal feature dimensionality of 1024. We model the classification and localization heads as a sequence of three 1D-CNNs with a kernel size of 3. These heads are shared across embeddings of different resolution levels. Finally, we introduce level-wise domain discriminators designed as a multi-layer perceptron (MLP) of depth 2 and width 512. Find in Tab. 2.3 the other relevant scenario-dependent hyperparameters related to our main contribution, the *SADA* loss. The rest of the hyperparameters follow the original implementation of Tridet [167].

During training, *SADA* first extracts video features using a frozen video backbone. For all the scenarios based on EpicKitchens100 this relies on a Slowfast [50] video backbone pre-trained on Kinetics [56]. This backbone is fed with raw videos from EpicKitchens100 [126] with a rate of 30 FPS, a feature stride of 16, and a maximum length of 2304 ensured by either

padding— for shorter videos— or random cropping. Alternatively, in the case of CharadesEgo[81] we rely on a frozen I3D video backbone pre-trained on ImageNet[17]. Our model then shuffles the pre-extracted features of the source and target domain videos, respectively. It then feeds a batch of 2 videos of each of the domains, resulting in a batch of 4 videos per training iteration. Note that depending on the scenario under study, one of the domains may have more data than the other. For this reason, in each epoch we repeat the smaller domain until the other finishes, ensuring that all the data of each of the domains is leveraged at least once.

During inference, we follow a similar approach to [144, 167] and define an exponential moving average of the trained model, which we update every iteration with an exponential decay of 0.999. Moreover, given the excess of final predictions, we use the standard SoftNMS [48] with an IOU threshold of 0.1, a minimum score threshold of 0.001, and a sigma value of 0.4.

2.5.3 Main results

EpicKitchens100. In Tab. 2.4 we first present the results obtained in the 4 different scenarios that we designed to evaluate the performance of our method when facing different *appearance shifts* induced by changes in the background information. Concretely, we first evaluate the performance of the Actionformer [144] and TriDet [167], the two best-performing methods on EK100 dataset [126], as well as our proposed architecture without our SADA loss. We then compare these 3 architectures with their extensions which include the *SADA* loss. Observe that our proposed loss improves the source-only (SO) version of all the architectures in all these 4 scenarios, showing robustness across the chosen underlying architecture. For instance on S3, our proposed loss yields an improvement of up to 2.49% mAP over its respective SO version. Additionally, we observe that our final architecture— i.e., Ours(SADA)— improves the best-performing existing SO method— i.e., Tridet[167]— by up to a 3.4% mAP for the *black-counter kitchens* scenarios— i.e., S1 and S3— and 1.82% mAP for the *white-counter kitchens*—i.e., S2 and S4. Similarly, the last two scenarios of Tab. 2.4 present a similar behavior than before. In all but one case on the Tridet [167] architecture, the use of the SADA loss yields a performance gain of up to 2.28% mAP. This is a relative 12.48% improvement. Our final model, moreover, improves the best-performing existing baseline— i.e., Tridet— by 1.73% mAP and 1.25% mAP, respectively.

CharadesEgo. In Tab. 2.5 we report a similar experimental comparison of our proposed loss when evaluated on CharadesEgo. Concretely, we showcase

Scen.	Model	mAP {10,20,30,40,50}%					Avg
S1	Actionformer [144]	30.21	28.73	26.39	22.60	17.09	25.00
	Tridet [167]	29.87	28.39	25.97	22.06	16.94	24.65
	Ours (src-only)	30.74	29.47	27.23	23.53	18.07	25.80
	Actionformer+SADA	31.71	30.30	27.95	24.08	18.69	26.55
	Tridet+SADA	29.55	28.27	26.16	22.89	17.63	24.90
	Ours (SADA)	**31.60**	**30.29**	**28.22**	**24.47**	**18.98**	**26.72**
S2	Actionformer [144]	27.46	26.54	24.61	21.85	17.19	23.53
	Tridet [167]	30.03	28.97	26.96	23.48	18.18	25.52
	Ours (src-only)	29.65	28.69	26.86	23.88	19.14	25.64
	Actionformer+SADA	29.84	28.89	26.83	23.76	18.85	25.63
	Tridet+SADA	30.21	29.33	27.58	24.35	19.44	26.18
	Ours (SADA)	**31.54**	**30.68**	**28.77**	**25.52**	**20.22**	**27.34**
S3	Actionformer [144]	28.11	26.94	24.89	21.43	16.51	23.57
	Tridet [167]	29.47	28.32	25.50	21.99	16.34	24.32
	Ours (src-only)	30.03	28.70	26.62	23.03	17.79	25.23
	Actionformer+SADA	30.54	29.31	27.36	23.75	18.80	25.95
	Tridet+SADA	31.34	30.16	27.94	24.48	19.31	26.64
	Ours (SADA)	**32.69**	**31.49**	**29.17**	**25.51**	**19.72**	**27.72**
S4	Actionformer [144]	33.52	32.31	29.84	26.48	20.11	28.45
	Tridet [167]	34.01	32.52	30.07	26.40	19.78	28.55
	Ours (src-only)	34.41	33.38	30.58	26.99	21.00	29.27
	Actionformer+SADA	34.11	32.87	30.60	27.05	20.93	29.11
	Tridet+SADA	34.50	33.19	30.65	27.25	20.83	29.29
	Ours (SADA)	**34.86**	**33.73**	**31.16**	**27.45**	**21.46**	**29.73**
S5	Actionformer [144]	22.87	21.87	20.10	17.23	13.33	19.08
	Tridet [167]	24.77	22.93	21.49	19.09	15.15	20.48
	Ours (src-only)	25.58	24.79	23.08	19.56	15.15	21.63
	Actionformer+SADA	24.85	23.99	22.21	19.11	15.28	21.09
	Tridet+SADA	24.88	24.00	22.20	19.02	14.88	21.00
	Ours (SADA)	**25.93**	**25.06**	**23.47**	**20.45**	**16.12**	**22.21**
S6	Actionformer [144]	22.16	21.22	19.71	17.44	14.08	18.92
	Tridet [167]	22.47	21.57	20.19	17.87	14.41	19.30
	Ours (src-only)	20.96	20.22	19.08	16.97	14.09	18.27
	Actionformer+SADA	23.05	22.10	20.71	18.31	14.72	19.78
	Tridet+SADA	21.00	20.08	18.64	16.39	13.13	17.85
	Ours (SADA)	**23.94**	**22.95**	**21.47**	**19.16**	**15.24**	**20.55**

Table 2.4: Comparison with SOTA for the 4 appearance-shift scenarios (1-4) and the 2 acquisition-shift scenarios (5-6) on EpicKitchens100.

the performance boost that SADA reports on the three test architectures—i.e., Actionformer [144], Tridet [167] and Ours. Observe that SADA, for instance, improves the performance of Tridet [167] by 1.37% mAP. Similarly, our proposed architecture *Ours (SADA)* improves the SO version by 0.75% mAP and yields the overall highest scores.

Model	mAP {10,20,30,40,50}%					Avg
Actionformer [144]	31.22	28.51	23.82	18.91	13.94	23.28
Tridet [167]	30.18	27.06	22.58	17.81	13.10	22.15
Ours (src-only)	30.68	27.74	23.41	18.46	13.58	22.77
Actionformer+SADA	31.46	28.57	**24.17**	19.04	13.87	23.42
Tridet+SADA	31.53	28.41	24.04	18.88	13.98	23.37
Ours (SADA)	**31.68**	**28.64**	24.09	**19.06**	**14.14**	**23.52**

Table 2.5: Comparison with the state-of-the-art on CharadesEgo.

2.6 Ablation studies

2.6.1 Quantitative study

Below we present various quantitative ablation studies to provide further detail about some of the key contributions of this chapter. Note that unless stated otherwise, all the ablation studies are evaluated on S3.

1) Comparing to other domain adaptation methods. In Sec. 2.5.3 we showed that *SADA* consistently improves the performance of the three tested state-of-the-art SO architectures when evaluated on our newly proposed setups. The question remains however of how well does *SADA* perform compared to existing domain adaptation methods. In this regard, following existing video-based domain adaptation works, we first evaluate various canonical UDA domain adaptation methods—i.e., DANN [39], ADDA [66], WDGRL [80], MSTN [83], FGDA [109] and DRDA [153]. These methods are integrated into our proposed underlying architecture resulting in a fair comparison with *SADA*. Additionally, we find that to the best of our knowledge, there is no existing UDA method for TAL in the literature that is directly comparable to us. Nevertheless, to provide a richer comparison, we adapt the closest action segmentation proposal, SSTDA [95] and a state-of-the-art domain-adaptation method for video classification— i.e., TranSVAE [172]— to our proposed setup. Concretely, in Tab. 2.6 we compare *SADA* with these baselines. These results empirically demonstrate the effectiveness of our method by attaining the best results over all tested UDA methods. More in detail, *SADA* yields an improvement of up to 0.7% mAP over the second best-performing method (ADDA).

2) Analysis of our loss. Next, we ablate over different variants of our proposed *SADA* loss (see Eq. 2.9). Concretely, in Tab. 2.7 we study the effect of class-wise distribution alignment (Eq. 2.7), global distribution alignment (equivalent to DANN [39]) and semantic background alignment (Eq. 2.8). In this regard, we highlight that the global adaptation seems to consistently

Model	mAP {10,30,50}%			Avg
DANN [39]	30.48	27.07	18.12	25.22
ADDA [66]	31.84	28.53	19.11	26.49
WDGRL [80]	25.05	22.08	13.91	20.35
FGDA [109]	26.21	22.99	14.12	21.10
DRDA [153]	30.79	26.97	18.60	25.45
MSTN [83]	31.07	27.16	17.21	25.14
SSTDA [95]	31.17	28.01	18.98	26.05
TranSVAE [196]	29.91	26.12	16.16	24.06
Ours (SADA)	**32.69**	**29.17**	**19.72**	**27.19**

Table 2.6: Comparison with SOTA UDA methods on S3.

Local	Global	Bkg	mAP {10,30,50}%			Avg
			30.03	26.62	17.79	24.81
	✓		30.48	27.07	18.12	25.22
		✓	30.34	26.66	17.04	24.68
	✓	✓	29.76	26.63	17.52	24.64
✓			31.36	28.01	18.67	26.01
✓	✓		30.09	26.88	17.43	24.80
✓		✓	**32.69**	**29.17**	**19.72**	**27.19**

Table 2.7: Ablation of the effect of the components of *SADA* on S3.

improve upon aligning only background embeddings. This is because the latter yields only a partial alignment of the embeddings, not considering *class anchors*. Therefore, a *rougher* yet complete adaptation might seem beneficial. We also observe that aligning local class-wise distributions has a considerable positive effect. Its effect, however, considerably decreases when combined with a global alignment [39], as all the *class anchors* are then subject to the concurrent alignments of domain-level and class-wise distributions. This observation is also consistent with the performance decrease that we observe when combining global and background alignment, which again suggests that the concurrent alignment of background embeddings with two adaptation losses is harmful to performance. Finally, we observe that we consistently obtain the best results when the local alignment loss is coupled with the complementary (non-overlapping) background loss— i.e., SADA loss— indicating that this semantic fine-grained, yet complete, alignment is the most desirable approach.

3) Analysis of the impact of *background anchors*. One critical aspect of anchor-based methods for TAL is the presence of numerous *background anchors*. We argue that the confusion that these embeddings induce is one of the main challenges for the transfer to a different domain. Concretely, we hypothesize this is because of their high intra-class variance, and their low inter-class variance with other action classes—i.e., they share aspects

Method	Mask *bkg anchors*	mAP {10,30,50}%			Avg	Perf. gap
Ours(src-only)		30.03	26.62	17.79	24.81	-
	✓	35.25	33.71	23.25	30.73	5.92
Ours(SADA)		32.69	29.17	19.72	27.19	-
	✓	**35.70**	**34.09**	**24.15**	**31.31**	**4.12**

Table 2.8: Ablation of the effect of the *background anchors* in the performance of the model on S3.

like appearance. Nevertheless, in this ablation we show that SADA partially mitigates this effect. For this, we design an experiment that artificially masks out all the *background anchors* during inference to elucidate the ideal performance if we could ignore entirely the confusion they generate—i.e., by being wrongly predicted as an action. Concretely, in Tab. 2.8 we compare our proposed methods with its SO version, and observe that masking out all the *background anchors* yields an overall improvement of 4.12% and 5.92% mAP, respectively. Hence, SADA reduces the impact of masking the *background anchors* by 1.8% mAP, indicating that our fine-grained alignment permits a better knowledge transfer to the target domain, mitigating the negative effect of these anchors.

4) Ablation of the effect of the λ hyper parameters As described in Eq. 2.9, we define our SADA loss using a set of hyperparameters $\{\lambda_l\}_{l\in L}$, where each of the parameters $\lambda_l \in (0,1)$ controls the influence of a given resolution level in the overall loss. Given the relevance of these parameters, below we ablate over different variations to showcase the effect that they have in the final performance of the model. Concretely, we define the three following scenarios:

1. **All levs 1**: This scenario sets all the parameters $\lambda_l = 1$, keeping the contribution of each of the levels the same.
2. **First 3 levs 1**: This scenario sets the first half of the levels to 1 and the rest to 0. This is $\lambda_0 = \lambda_1 = \lambda_2 = 1$ and $\lambda_3 = \lambda_4 = \lambda_5 = 0$.
3. **Last 3 levs 1**: This scenario sets the first half of the levels to 0 and the rest to 1. This is $\lambda_0 = \lambda_1 = \lambda_2 = 0$ and $\lambda_3 = \lambda_4 = \lambda_5 = 1$.

All in all, this study attempts to clarify how sensitive is our model to this design decision. In this regard, these results (see Tab. 2.9) indicate a mild sensitivity to this choice. Specifically, we observe that the second best-performing method attains an absolute difference over the optimal choice of only -0.08% mAP and -0.57% mAP for the two considered scenarios S1 and S3, respectively. Moreover, we observe that the worst performing

Strategy	mAP {10,30,50}%			Avg
All levs 1	32.42	29.59	19.34	27.11
First 3 levs 1	30.96	27.48	18.29	25.58
Last 3 levs 1	30.54	27.19	18.28	25.34
"Optimal choice"	**32.69**	**29.17**	**19.72**	**27.19**

Table 2.9: Ablation of the effect of the λ parameters, which regulate the influence of each of the resolution levels on the overall SADA loss. The "optimal choice" refers to the hyperparameter choice resulting from a Bayesian optimization process that leverages the labeled source domain to identify the best-performing set of parameters. Moreover, for efficiency purposes, we limit the search space to a uniform sampling in the (0,1] interval—i.e., $\{0.1, 0.2, \ldots, 0.9, 1.0\}$.

naive strategy still outperforms in both scenarios 4 out of the 6 tested UDA baselines (see Tab. 2.6).

5) Study of the class embedding: One of the main contributions of our work is to adversarially align distributions in a class-wise fashion. Intuitively, this requires that our level-wise domain discriminator *knows* the class distribution that it is aligning. In this regard, in Sec. 2.3.5 we propose to concatenate to every anchor a learnable embeddings $e_i \in \mathbb{R}^F$ of its corresponding class i (see Eq. 2.7-2.8). In this ablation, we empirically justify our choice. For this, we compare our approach against several non-learnable alternatives to encode a given class i: a naive one-hot encoding, a random class-wise dense initialization, and the sinusoidal encoding originally proposed by [68].

Concretely, in Tab. 2.10 we show the experimental results of each of the variants tested in S3. These indicate that a naive one-hot encoding of a class is the best-performing non-learnable strategy, obtaining considerable improvements over the other two tested non-learnable baselines of up to 2.32% mAP. Moreover, to our surprise, the popular Sinusoidal encoding [68] proves to be the worst-performing method, being consistently outperformed by even the random dense encoding. Finally, we highlight that the use of a learnable class embedding yields the best results, improving by 0.36% mAP the performance of the one-hot encoding strategy, which justifies our choice.

6) Ablation per class: In this section, we complement the analysis from Sec. 2.5.3 with the detailed class-wise metrics. Concretely, in Tab. 2.11 and Tab. 2.12 we show the respective class-wise mAP scores of the 10 considered classes on both S1 and S3. For the analysis, we also include the results obtained

Strategy	mAP {10,20,30,40,50}%					Avg
One-hot	32.29	31.09	28.58	24.88	19.56	27.28
Random emb	30.71	29.50	27.43	23.78	18.52	25.99
Sinusoidal [68]	29.93	28.57	26.44	22.87	17.01	24.96
Learnable	**32.69**	**31.49**	**29.17**	**25.51**	**19.72**	**27.72**

Table 2.10: Ablation of the effect of the use of a learnable class embedding over other static strategies.

	take	*put*	*wash*	*open*	*close*	*insert*	*turn-on*	*cut*	*turn-off*	*pour*
Ours (source-only)	24.20	29.30	37.30	32.69	**23.89**	10.98	37.39	**18.18**	26.24	18.55
DANN [39]	24.85	30.51	37.09	**35.08**	22.93	**11.68**	35.80	17.23	24.19	**20.35**
Ours (SADA)	**26.31**	**30.70**	**39.14**	34.68	23.42	10.58	**39.86**	17.05	**26.58**	18.78

Table 2.11: Per class mAPs (in percentage) obtained by the source only variation of our model, DANN [36] and our proposal *SADA*. These results correspond to S1, which defines *black old* kitchens as the source and the rest as a target.

with the source-only variant of our model as well as our chosen DANN[34] baseline.

Observe that in S3 (see Tab. 2.11) our model attains the best class-wise performance in 5 of the classes, while DANN [34] does so on 3, and the source-only model on 2. In contrast, in S1 (see Tab. 2.12) our method obtains a much clearer improvement over the chosen baselines, yielding the best results in 8 of the 10 classes. Overall, we can observe that our method performs very well in the 3 majority classes of both scenarios, where we highlight the absolute improvement of SADA over DANN [34] of 6.8% mAP for the class *put* in S3. Moreover, our method fails to improve the performance of action *pour* in both scenarios, which we attribute to the lack of sufficient data to operate on our proposed methodology. We also highlight that as observed in Sec. 2.6, S1 presents a more challenging setup, degrading the performance in other actions such as *open*, *close*, *insert* or *cut*. This might indicate the existence of intrinsic qualitative aspects that harden the adaptation when dealing with *old* videos. This is not the case in S3, as aside from the aforementioned *pour* segments, it only fails to achieve the best results for the *turn-on* action.

2.6.2 Qualitative analysis

1) Qualitative VMR results: We complement our quantitative results with a qualitative study. For this, we depict in Fig. 2.8 the segment visualization of S1— i.e., using *old* dark-counter kitchens as the source domain, and the rest as the target. Observe that in this case, both ActionFormer [144] and

	take	*put*	*wash*	*open*	*close*	*insert*	*turn-on*	*cut*	*turn-off*	*pour*
Ours (source-only)	24.11	25.69	29.77	31.96	27.8	6.69	25.52	34.16	17.48	**29.18**
DANN [39]	24.78	24.38	28.75	32.73	29.32	5.38	**27.52**	36.73	17.70	29.05
Ours (SADA)	**28.32**	**31.25**	**31.18**	**34.52**	**31.82**	**7.29**	26.66	**39.12**	**18.68**	28.34

Table 2.12: Per class mAPs (in percentage) obtained by the source only variation of our model, DANN [36] and our proposal *SADA*. These results correspond to S3, which defines *black new* kitchens as the source and the rest as a target.

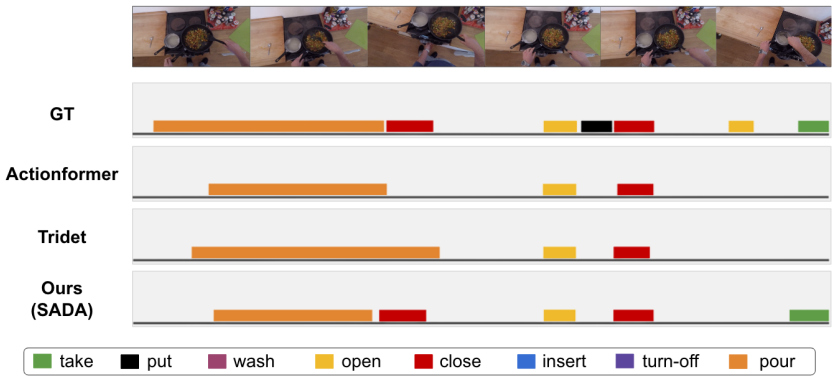


Figure 2.8: Visualization of the predicted segments on S1— i.e., using *dark old kitchens* as source— of our method and the chosen set of source-only baselines. We include on top the ground-truth (GT) segments as a reference.

Tridet [167] perform similarly. Concretely, they mainly miss one *close* action at the beginning, a *put* in the middle section of the video, and finally the last two actions *open* and *take*. In contrast, our model is able to correctly detect the previously missing *close*, and the final *take*. In short, reducing by half the number of undetected actions in the video.

Similarly, Fig. 2.9 depicts a segment visualization of S3 of our proposed method versus the chosen baseline models. In this visualization, we can observe that the Actionformer misses many of the segments in the shown video clip ignoring all the *take* actions and mistaking a *close* for a *take*. Tridet performs better but misses all the *take* actions in the first half of the video while worsening the boundary prediction of the last *close*. *SADA* improves Tridet by predicting the second *take* and considerably improving the boundary of the last *close* action.

Finally, Fig. 2.10 shows a segment visualization that compares the chosen baselines with SADA when applied to the CharadesEgo dataset [81]. Observe that this visualization presents a much more complex setup where there

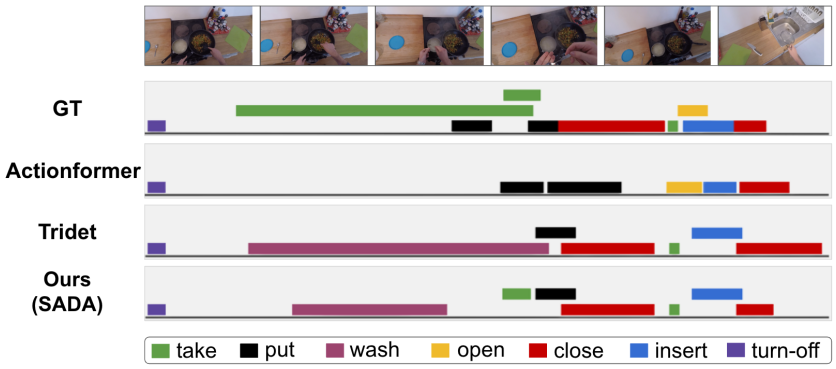


Figure 2.9: Visualization of the predicted segments on S3—i.e., using *XXX* as source– of our method and the chosen set of source-only (SO). We include on top the ground-truth (GT) segments as a reference.

are numerous overlapping actions. In this regard, we highlight that the worst-performing model is the ActionFormer [144]. This model misses the action *walk* and one of the actions *put*. It also presents important limitations in terms of localization, especially for the action labels *drink* and *hold*. The Tridet [167] considerably improves this method. Nevertheless, it still misses the action *walk* and is unable to locate the longest *put* action. Additionally, Tridet predicts an action *take* that does not correspond to any ground truth. Our method, finally, presents much more accurate scores. Importantly, our method does not miss any of the ground-truth actions and in most cases reports considerably accurate segments. We only highlight the persisting confusion in the two actions *hold* which is consistent across all the tested baselines.

2) TSNE study: Figure 2.11 also shows the TSNE plots of the domain-invariant embeddings of *SADA* with respect to learning source-only. Given that *SADA* is an adversarial-based class-wise loss, we depict the plots of the 3 majority action classes (first 3 columns). Observe that SO (top row) yields clearly unaligned distributions with scarce to no overlap in the projected space. Our method, in contrast, presents a considerable distribution mix improving the alignment of class-wise distributions across domains. Given the anchor-based nature of our method, we have numerous *background anchors*—i.e., not assigned to any GT label. As observed in the last column, these are also aligned by our method (see Eq. 2.8) therefore effectively aligning the entire data distributions but in a semantically sensitive way.

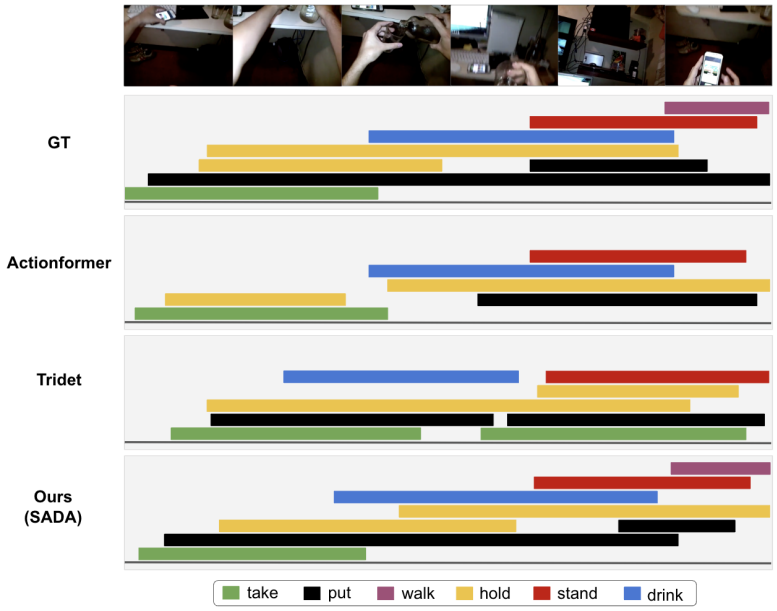


Figure 2.10: Visualization of the predicted segments on CharadesEgo of our method and the chosen set of source-only baselines. We include on top the ground-truth (GT) segments as a reference.

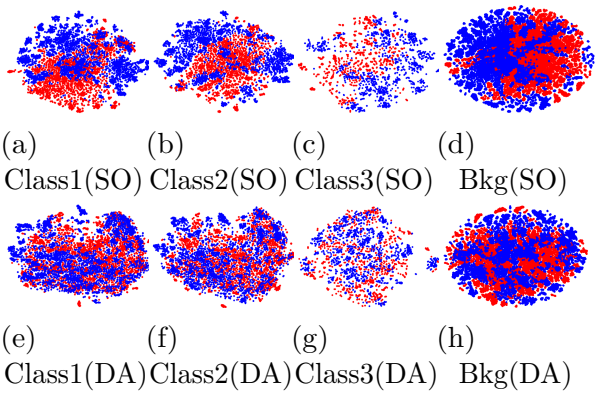


Figure 2.11: TSNE plots of the source-only (SO) variation of our model (top row) and our proposed domain adaptation model (DA) (bottom row). Find in the first 3 columns the TSNE plots of action classes 1 to 3 of the source (red) and target (blue) domain anchors. The last column shows the plot of the background anchors, so those not assigned to any GT label.

From Fig. 2.11 it remains unclear, however, whether this alignment behavior remains consistent across the different resolution levels. To showcase this behavior, in this section we first include in Fig. 2.12 - Fig. 2.14 the TSNE plots

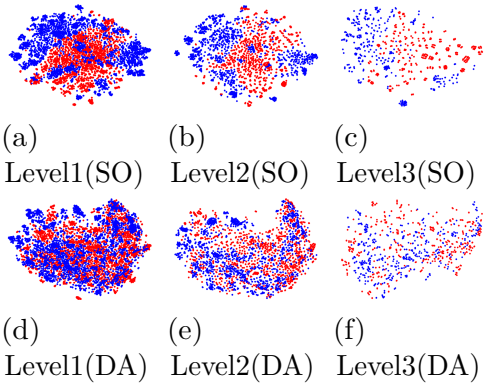


Figure 2.12: TSNE plots of class 2 of the source-only variation of our model (top row) and our proposed model (bottom row). Concretely, find in the 3 columns the TSNE plots of class 2 on the first 3 resolution levels of the source (red) and target (blue) domain anchors.

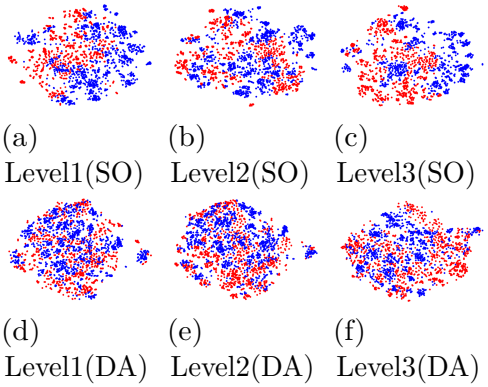
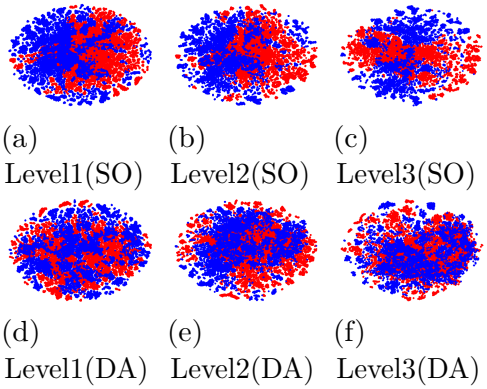


Figure 2.13: TSNE plots of class 3 of the source-only variation of our model (top row) and our proposed model (bottom row). Concretely, find in the 3 columns the TSNE plots of class 3 on the first 3 resolution levels of the source (red) and target (blue) domain anchors.

of the 3 majority classes and of background embeddings, respectively. Each of these figures includes the different plots at each of the first 3 resolution levels, where level 0 is the shallowest, and level 3 is the deepest studied level. We highlight that deeper levels are not meaningful to plot given the scarce number of resulting domain-invariant embeddings, caused by the downsampling that each of the levels of the architecture performs.

Observe that the influence of the alignment loss follows a similar pattern in all the resolutions. Concretely, in the three presented, we observe a considerable



(a) Level1(SO) (b) Level2(SO) (c) Level3(SO)

(d) Level1(DA) (e) Level2(DA) (f) Level3(DA)

Figure 2.14: TSNE plots of *background class* of the source-only variation of our model (top row) and our proposed model (bottom row). Concretely, find in the 3 columns the TSNE plots of *background class* on the first 3 resolution levels of the source (red) and target (blue) domain anchors.

mixing between the source and target distributions (red and blue, respectively). This contrasts with the very clear disentanglement of the representations of both domains in the source-only version, where these present little to no overlap. We highlight what seems to be the only exception which is resolution level 3 of class 1. This one shows little improvement in the mixing of the distributions compared to the source-only variant. We also find that our overall improvement of the mixing of the distributions is consistent when analyzing the *background class* embeddings. Observe in Fig. 2.14 that in all 3 studied resolution levels, *SADA* improves very considerably the alignment, pushing the feature space of both domains to be domain invariant. This emphasizes the positive influence of the background alignment term of our loss (see Eq. 2.8).

2.7 Conclusions and future work

In this chapter, we addressed the challenge of Unsupervised Domain Adaptation in realistic Temporal Action Localization (TAL) scenarios. For this, we introduced SADA, a novel semantics-informed adversarial loss that promotes fine-grained feature alignment, going beyond global distribution matching strategies. To support the study of this setting, we proposed a suite of seven cross-domain benchmarks that provide a comprehensive assessment of the model performance across various domain shifts. Overall, this chapter contributes to the central goal of this thesis by enabling video understanding

models to learn representations that generalize beyond dataset-specific visual cues and remain reliable under changing environments.

Future work: Several open directions naturally arise from this work. Extending SADA to scenarios with partially overlapping actions would further improve its applicability to real-world datasets. In addition, studying its behavior with larger and more expressive video backbones remains an important step toward scalability. Finally, while this chapter focuses on visual domain shifts in isolation, many practical video understanding problems are inherently multi-modal. This observation motivates the next chapter, which shifts the focus from visual domain generalization to linguistic generalization in video understanding, examining how distribution shifts in natural language queries pose complementary challenges to robust video understanding for retrieval and grounding applications.

Chapter 3

Beyond Caption-Based Queries for Video Moment Retrieval

In Chapter 2, we focused on OOD generalization across visual domains for the task of TAL. In this Chapter we explore the similar, yet more challenging grounding task, that is Video Moment Retrieval (VMR). VMR extends the TAL task into an open-vocabulary setting, where predefined action classes are replaced by free-form textual queries that guide the visual grounding process. This more realistic setup introduces an important new dimension in the study of OOD generalization of Video Understanding applications: generalization to language. Concretely, we highlight the importance of real-life video-understanding applications to not only generalize to visual variability, but also to the significant shifts in linguistic formulations. From a VMR perspective, unlike the overly detailed textual queries used in the majority of existing VMR datasets, real-life users' prompts rarely describe the actions or objects they are looking for exhaustively. Instead, these queries are often vague or simply incomplete. This induces the important linguistic domain shift that queries seen during training and deployment often fundamentally stem for significantly different distributions. Consequently, models may perform strongly on caption-based benchmarks while their performance degrades on more realistic scenarios.

Chapter 3 addresses this second axis of generalization, this being achieving robust performance under realistic, potentially under-specified textual queries. Concretely, we demonstrate that current datasets present important linguistic biases and introduce mechanisms to overcome this issue. In doing so, this chapter extends our broader thesis question from learning domain-invariant video representations to learning linguistic-invariant grounding mechanisms.

3.1 Introduction

Video Moment Retrieval (VMR) aims to localize temporal segments in a video given a user-defined textual query. While current models achieve remarkable success on existing benchmarks, in this work we raise awareness

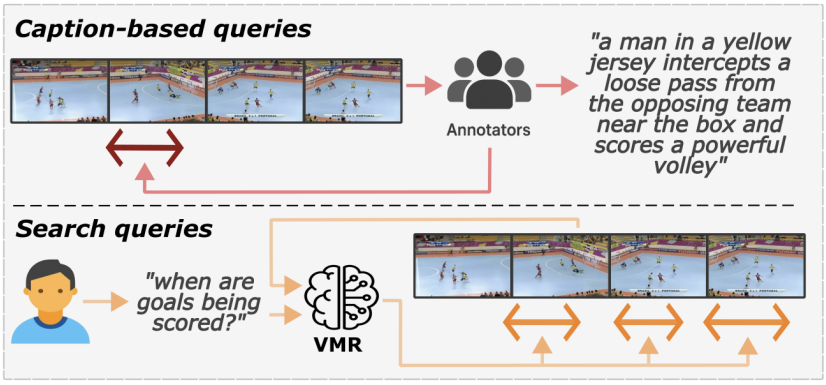


Figure 3.1: After watching a video, annotators write detailed, visually-informed captions that map to a single GT moment. However, at inference time, users formulate less detailed, visually-uninformed search queries that often map to multiple GT moments.

of a key limitation of VMR datasets: text queries are defined using the annotated captions, written after annotators watch the videos. These captions, which we name *caption-based queries*, induce a visual bias—overly descriptive visually-informed textual annotations. In contrast, real users interact through *search queries*, often formulated without watching the video, which can be of a more general and *under-specified* nature [26]. For instance, while an annotator might formulate a caption-based query *"a man in a yellow jersey intercepts a loose pass from the opposing team near the box and scores a powerful volley"*, a typical search query can be *"when are goals being scored?"* (see Fig. 3.1).

Devising VMR methods that robustly perform on search queries requires suitable benchmarks, different from the existing caption-based ones [110, 58, 190, 206]. However, collecting new search-query datasets remains an open challenge, as it is unclear how text annotation and video observation can be decoupled in a feasible manner. We instead re-purpose existing datasets—which include paired videos and captions—by proposing a pipeline that under-specifies captions. We systematically explore levels of under-specificity, by changing the level of details available in the original caption. We consequently propose three {*S*}earch-query benchmarks: *HD-EPIC-S{1,2,3}*, *YC2-S* and *ANC-S* based on HD-EPIC, YouCook2 and ActivityNet-Captions.

DETR [94] has become the cornerstone of most existing VMR methods [162, 211, 193, 156], thanks to its usage of K learnable decoder queries, each of which maps to a potential retrieved moment with a corresponding confidence

score. Our evaluation indicates that these methods, trained on caption-based queries, substantially degrade when evaluated on more under-specified search queries. We identify two key factors driving this degradation: (i) a *language gap*, reflecting the linguistic distribution shift between caption and search queries, and (ii) a *multi-moment gap*, arising from how caption-based queries map to a single ground-truth (GT) moment, while the under-specified search queries often map to multiple moments.

In this chapter, we identify and quantify the impact of both the language and multi-moment gaps, and particularly address the multi-moment gap by proposing architecture modifications including the removal of self-attention mechanisms and the addition of a decoder-query dropout regularizer. These modifications improve generalization to search queries without the expensive task of re-annotating VMR training sets.

In short, our contributions are: 1) we explore the task of VMR beyond using captions as textual queries. We re-formulate these as under-specified versions of existing captions, so they are closer to common user-defined queries, while still making the most of available benchmarks; 2) we create three VMR benchmarks with search queries by mapping caption-based queries to under-specified search queries; 3) we demonstrate the significant degradation in performance and identify its two main causes: a *language* and a *multi-moment gap*; and 4) we mitigate this degradation, particularly that induced by the multi-moment gap, by introducing targeted architectural modifications that boost generalization to search queries.

3.2 Related work

Video Moment Retrieval has become a cornerstone in video understanding, aiming to localize start-end times of moments in a video, based on textual queries. Existing approaches can be broadly categorized into *proposal-based* and *proposal-free* methods. *Proposal-based* approaches generate candidate temporal segments through temporal anchors [53, 49, 91] or sliding-windows [47, 73], later refined via cross-modality modules. However, their performance heavily depends on the quality and redundancy of these proposals. In contrast, *proposal-free* methods avoid the explicit candidate generation, leveraging a single-stage architecture to predict the moment segments. Most of these works adopt DETR-based architectures [94], which refine a fixed set of learnable decoder queries, each of which represents a candidate segment. This paradigm was introduced by [110], with follow-ups of QD-DETR [163], CG-DETR [162] or LD-DETR [211] improving various

aspects such as the cross-modality modules, recursive decoding schemes, etc. Our work also focuses on the DETR architecture, as this is the foundation of the majority of existing state-of-the-art VMR methods [162, 211, 193, 201, 208], thus maximizing the impact of our findings.

Generalization of VMR methods: While DETR-based VMR models have shown remarkable success in existing VMR benchmarks, their generalization capabilities beyond their training data distribution remains largely under-explored. Most existing works address this from a vision-centric perspective, analyzing aspects like action duration and temporal shifts [101, 144, 78] or temporal biases [101, 125, 157]. More recent studies have begun exploring the role of language biases in generalization [14, 99], tackling rare-word usage and grammatical mistakes [210] and the use of unlabeled data to improve language robustness [175]. In parallel, research from the multi-modal and linguistic community [165, 192] emphasizes how under-specification and contextual variation also impact the generalization capabilities of models. In this work, we find that a related phenomenon occurs in VMR due to the use of captions, written by annotators after watching the videos, as textual queries. The visual bias that this induces leads to overly descriptive fine-grained queries, potentially misaligned with the more abstract queries that users often prompt in real-life situations.

Query collapse in DETR: A core issue that we address is the query collapse in DETR-based architectures, whereby only a small subset of decoder queries meaningfully contribute to the final prediction(s), while the rest remain inactive. This issue, reported in object detection [105, 115, 129], temporal action detection [156], and 3D detection [199, 174], is largely attributed to the sparse supervision from the one-to-one matching [94]. We observe a similar phenomenon in VMR, driven instead by the single-moment prior of existing benchmarks, which typically provide one annotated moment per query. This leads to a significant query collapse that hinders generalization to multi-moment queries. While some works introduce alternative mechanisms to provide additional supervision signals [155, 150, 178, 122, 129, 201], these mainly target accelerating convergence, proving unable to overcome this strong prior. Curating new datasets with multiple annotated moments per query could alleviate this issue [110, 202], but would entail costly re-annotations or directly discarding most existing datasets. This motivates our approach, which introduces architectural modifications that counter the single-moment prior, leveraging existing datasets while improving generalization to unseen multi-moment scenarios.

3.3 Problem definition and benchmarking

3.3.1 Problem definition

Video Moment Retrieval (VMR) is defined as follows: Given a video–query pair (v_i, q_i), the task is to predict the start–end times $\{(s_j, e_j)\}_{j=1}^{M_i}$ of all temporal segments in v_i corresponding to the textual query q_i—i.e. moments. In this work we revisit this task, and focus on the often overlooked aspect of how textual queries are defined.

Specifically, we highlight the underlying assumption of all existing VMR benchmarks where queries are created from the captioning annotations—i.e. annotators who watch the videos before writing a sentence that best describes the moment [53, 58, 84, 206]. These *caption-based queries* induce a visual bias that the queries perfectly match the description of the moment, thus descriptive and fine-grained in nature. In contrast, real-world users formulating their textual queries are normally unaware of the detailed content of the video, relying instead on broader, *under-specified* descriptions. Such queries can range from moderately detailed to very general ones, differing substantially from captions. We refer to these as *search queries*.

To model the distribution shift between caption and search queries, we derive $\mathcal{Q}_{search}$ from $\mathcal{Q}_{caption}$ through varying degrees of under-specification. We thus capture the shift from visually-informed to visually-uninformed textual queries. This allows us to study how VMR methods trained on caption-based queries $\mathcal{Q}_{caption}$ perform on search queries $\mathcal{Q}_{search}$, more closely aligned with common situations in which users either do not know or do not exactly remember the contents of the video.

3.3.2 Benchmarks

Common VMR benchmarks rely on caption-based queries, and thus do not include search-query annotations. Re-annotating benchmarks is a challenging task, as it is unclear how to disentangle search-query annotation from video observation in a feasible and scalable manner. Accordingly, we propose to make the most of existing datasets, introducing a pipeline that rewrites a densely-annotated caption-based dataset, to a search-query variant. Dense temporal annotations are essential here since, as queries become more under-specified, these may correspond to multiple additional moments in the video. If dense annotations are provided, the search for these new correspondences can be done automatically. This contrasts with sparsely

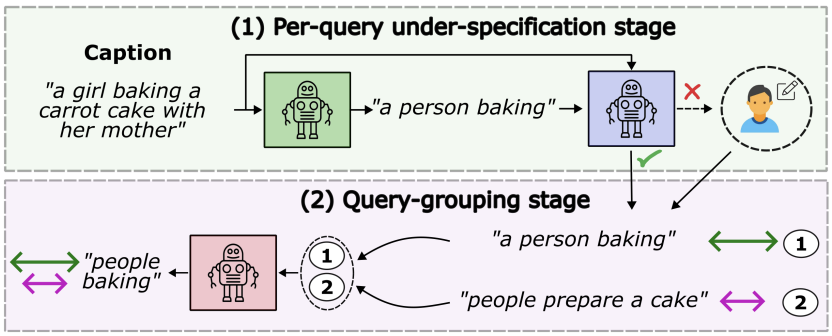


Figure 3.2: Overview of the search-query pipeline. Each of the caption is first processed by an agent that generates per-query under-specifications, which are validated by a second identical agent and manually re-annotated if abnormal. Individual queries mapping to the same under-specified query are then grouped, and a final agent produces a representative search query per group.

annotated datasets [110], which would require extensive manual re-annotation to cover unlabeled moments corresponding to the under-specified search query. In particular, we use three public datasets with dense temporal annotations to introduce our search-query pipelines. We detail our pipeline next.

3.3.2.1 Search-query pipeline

Our pipeline (see Fig. 3.2) comprises two main stages:

1) **Per-query under-specification stage:** To simulate the potential ambiguity of search queries, we generate under-specified queries from fine-grained ones through an LLM-based pipeline. Specifically, we instantiate two cooperative agents based on Gemma-12B [194]: a rewriter and a validator. The rewriter agent receives a fine-grained caption and rewrites it into a less detailed version while preserving the core semantics. For example, the caption *"a man tying his running shoes before starting a marathon"*, will be rewritten as *"a person getting ready to exercise"*. This step allows us to model alternative under-specified search queries where users may omit context information like subject, object or intent. Inspired by [177], we prevent hallucinations by introducing a second agent that acts as a validator. This agent flags any inconsistent rewritings, which are subsequently corrected by human annotators.

2) **Query-grouping stage:** When a query is under-specified, it can correspond to multiple valid moments in the video, since several fine-grained

situations can fit the same broader description. For example, *"a person cooking food"* could match segments showing *"a man sweating onions"*, or *"a woman stirring soup"*. This makes it essential to group all original fine-grained queries that map to the same or highly similar under-specified query. To perform this grouping, we compute pairwise similarities across all under-specified queries using a pre-trained transformer-based sentence encoder [1]. Queries with a high similarity are merged into the same group, forming a multi-moment instance. We then use an LLM-based aggregator that summarizes each group into a single representative under-specified query, removing minor differences across group members while keeping their shared semantics.

3.3.2.2 Search-based VMR benchmarks

The proposed search-query pipeline enables us to introduce three search-query benchmarks, denoted by "-S":

HD-EPIC-S{1,2,3}: HD-EPIC [206] is a large-scale egocentric dataset featuring long cooking videos. Due to the exceptional level of detail of its annotated captions (16.47 words per query on avg.), we derive three progressively under-specified variants—i.e., S1, S2, S3—by gradually removing contextual details. For example, *"Pick up a tissue from inside the plate on the countertop using the right hand"* → *"Pick up a tissue from the plate"* (S1) → *Pick up a tissue* (S2) → *Pick up something* (S3).

YouCook2 (YC2)-S: YC2 [190] contains step-level narrations for instructional cooking videos. YC2-S replaces the original detailed descriptions with under-specified versions, mostly emphasizing the main actions. For example *"Add salt to the pan and mix"* → *"Season food"*.

ActivityNet-Captions(ANC)-S: ActivityNet-Captions [58] consists of third-person open-domain videos. ANC-S derives under-specified versions of its corresponding textual queries, removing fine-grained contextual details. For example, *"The man in white shirt is strumming the bongo drum."* → *"A person plays an instrument"*. Unlike previous benchmarks, ANC-S presents a considerable number of multi-moment search queries with overlapping moments.

Tab. 3.1 depicts the main statistics of these benchmarks. Using our pipeline, we extend existing single-moment datasets into new multi-moment versions, where up to 47.57% of the queries correspond to multiple moments. Due to the linguistic under-specification, the average query length is also reduced by up to 82%.

Dataset	# videos	# queries	Duration per video(s)	Moments per query	Moments per query (multi)	# multi. queries	% multi. queries	Query length
HD_EPIC [206]	156	59,454	954	1.00	0.00	0	0.00	16.47±7.9
HD_EPIC-S1	156	36,819	954	1.61	3.64	8,560	23.25	6.09±2.5
HD_EPIC-S2	156	31,521	954	1.88	3.87	9,717	30.83	3.73 ±1.2
HD_EPIC-S3	156	10,266	954	5.79	11.09	4,873	47.47	3.03 ± 1.0
ANC [58]	14950	71,957	152.8	1.00	0.00	0	0.00	13.16 ±6.1
ANC-S	14950	59,138	152.8	1.21	2.45	8,818	14.91	4.83 ±1.4
YC2 [190]	2268	13,829	326	1.00	0.00	0	0.00	8.86 ±3.97
YC2-S	2268	7,466	326	1.84	2.97	3,212	43.02	2.08 ±0.50

Table 3.1: Statistics of the search-based VMR benchmarks

3.3.3 Metrics for Search-Based VMR

3.3.3.1 Motivation

VMR performance is typically measured using Recall@1 (R1) and mean Average Precision (mAP), capturing top-1 accuracy and overall ranking quality, respectively. However, these metrics are inadequate when evaluating multi-moment queries. When retrieving under-specified queries these often map to additional GT moments. For example, while caption-based queries might target *"a man in a black shirt enters through the kitchen door"* retrieving a single moment, a more general search query *"a person entering a room"* could naturally map to multiple moments, including the previous. This setting thus requires metrics that estimate how individual moments are retrieved, regardless of whether they arise alone or alongside other moments.

Existing metrics like R1 or mAP are unsuitable for two reasons: First, recall metrics like R1 only evaluate accuracy over the top-k predictions—i.e., $k = 1$ for R1. While appropriate when queries map to exactly k moments, these metrics provide an incomplete evaluation when queries map to more than k moments. For instance, for a 2-moment query, R1 assesses if the top-1 prediction matches *any* GT, ignoring if the other GT was retrieved at all. Second, metrics like mAP aggregate all GT moments of a query into a single video-query score, obscuring per-moment retrieval quality. Consider a query q_1 that maps to a GT moment g_1. If a model fails to retrieve it, mAP would clearly indicate failure. However, for a more general query q_2 that maps 4 moments (g_1-g_4), the same model might detect g_2-g_4 but still miss g_1. In this case, mAP would remain high, masking the error of g_1. Thus, the retrieval quality of an individual moment depends on how many moments co-occur with it, making it unsuitable for a fair evaluation.

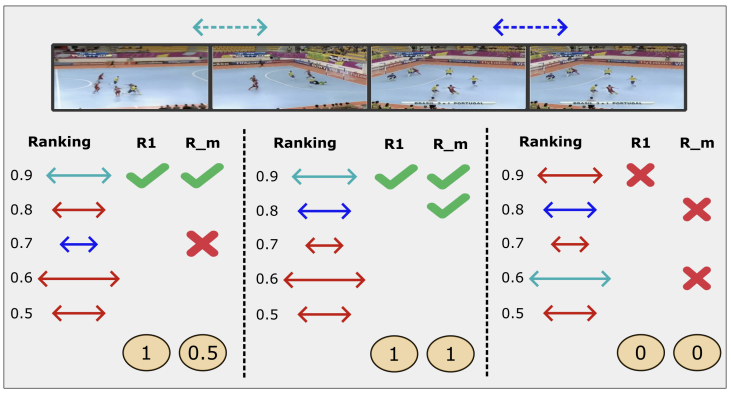


Figure 3.3: Intuitive examples that showcase the behavior of both R_m and $R1$. Here the solid lines correspond to prediction, and the dashed ones correspond to GT moments. Moreover, in R1, the checkmark indicates that the entire query is marked as correct, while for R_m, since it performs a per-GT evaluation, this indicates whether the corresponding prediction "correctly" retrieves a GT moment or not, based on the criterion defines by the metric. The orange circles indicate the global score of the instance, which is consistent with the single score produced by $R1$, or by the average of the multiple per-GT scores for R_m. .

3.3.3.2 Intuitive examples

Observe in Fig. 3.3 an illustration of the behavior of our proposed metric R_m with respect to the more standard $R1$. In the left example, observe that the highest confidence prediction matches one of the GT moments, while the second GT is matched by the third highest-prediction. In this case, standard R1 metric would predict a score of 1, since the top prediction does correspond to one of the GT, but this score does not account for the quality of the detection of the second GT moment. R_m in contrast, would provide a per-GT score, where the first moment would get a score of 1 since it was matched to the highest-confidence prediction, while the second GT moment—matched to the third highest confidence—would get a score of 0. The reason is that the model ranked the second prediction with a higher confidence, which corresponds to a false-positive—not matching any GT. Following the intuition of R1, this prediction has not been "accurately" retrieved.

The contrary happens in the second example, where R_m assigns a score of 1 for both GT, since one corresponds to the highest confidence, and the second, despite being ranked second, it is not penalized since the prediction with a

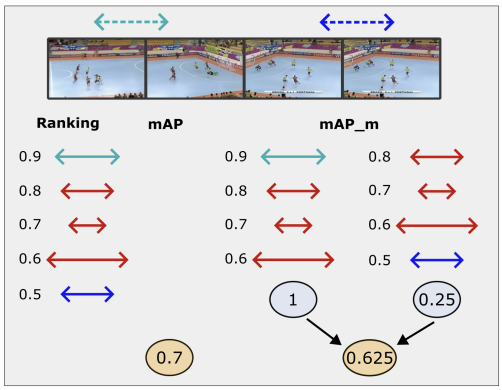


Figure 3.4: Example that showcases the behavior of both mAP and mAP_m. mAP computes a global score for the entire ranking, which is depicted in its corresponding orange circles. mAP_m, in turn, computes a score for each of the two GT moments. Find the two rankings that each of this evaluation leverages, effectively ignoring the influence of predictions that, while match a different GT, should not be considered invalid. The orange circle from mAP_m corresponds to the average of the two respective per-GT scores. .

higher confidence is also a match to a different GT. This hence shows similar behavior to R1.

Finally, in the third example both $R1$ and R_m have a similar behavior. Since a false-positive prediction is ranked on top, all the remaining predictions that correspond to a GT get assigned a score of 0. These examples exemplify how R_m computes per-GT scores, evaluating the quality of the retrieval of each GT moment independently, and without being affected by other potential matches that may co-occur.

A similar behavior is shown in Fig. 3.4, which showcases the behavior of mAP_m with respect to mAP. This considers a scenario where the model detects a given GT moment with the highest-confidence prediction, while the second is detected with the lowest one. In this case, mAP computes a global score of 0.7, where the correct prediction of one of the GT moment masks the poor detection of the other. mAP_m, in contrast, computes a score for each GT were the only difference is that each evaluation ignores all the predictions that match any other GT. For instance, to evaluate the dark blue moment (right most example), mAP_m ignores the prediction that corresponds to the light blue one. This avoid penalizing matches that, while different, are still valid and should thus not be considered incorrect. Hence, in this case, the

score for one of the moments is 1, while the other is of 0.25 giving a final mAP_m score of 0.625 when averaged across the two GT moments.

3.3.3.3 Preliminaries

Given a video-query pair, a VMR model outputs a set of K predictions—i.e., candidate moments—, denoted as:

$$\mathcal{P} = \{p_1, \ldots, p_K\}, \tag{3.1}$$

where each prediction p_i is a temporal segment predicted by the model, associated with a confidence score $c(p_i)$. These predictions are sorted in descending order:

$$c(p_1) \geq c(p_2) \geq \cdots \geq c(p_K). \tag{3.2}$$

Moreover, this video-query pair maps to a set of GT moments $\mathcal{G}$:

$$\mathcal{G} = \{g_1, \ldots, g_n\} \tag{3.3}$$

Given a certain IOU threshold τ, we follow the existing literature [19] and define a match of a prediction with a GT moment as:

$$match(p_i, g_j, \tau) = \begin{cases} 1 & IOU(p_i, g_j) \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

For convenience, let us also define the cases where a prediction matches a GT moment that while valid, differs to the moment g_j that is under evaluation:

$$match\_other(p_i, g_j, \tau) = \begin{cases} 1 & \exists g_k \neq g_j \; st. \; IOU(p_i, g_k) \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

3.3.3.4 Multi-moment recall R_m

Standard Recall@1 (R1) assigns a score per video-query pair, this being 1 if the highest-ranked prediction matches any of the GT moments, and 0 otherwise. This provides only a partial performance overview when evaluating multi-moment queries—those matching to multiple GT moments—as this metric does provide information on whether the model was able to successfully detect all the GT moments.

The goal of the R_m metric is to instead evaluate the detection quality for each of the GT moments, independently. Importantly, our metric avoids

the interference of other co-occurring moments in the score assigned to the evaluation of a given GT moment.

More specifically, R_m considers a given GT moment g_j as correctly retrieved if it appears before any false positive predictions, ignoring predictions that do not match g_j as they cannot be considered mistakes, since they match different, equally valid, GT moments.

Formally, let us define the index of the first prediction matching g_j:

$$i^* = \min(i \mid match(p_i, g_j, \tau) = 1). \tag{3.6}$$

Moreover, the index of the first false-positive is defined as:

$$i_j^{FP} = min(i|match(p_i, g_j, \tau) = 0 \ \wedge \ match\_other(p_i, g_j, \tau) = 1). \tag{3.7}$$

With this, the recall score for a given GT moment g_j is defined as follows:

$$R_m(g_j, \tau) = \begin{cases} 1 & i^* \geq i_j^{FP} \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

Finally, in order to obtain global scores for our dataset, we additionally compute an aggregated score:

$$R_m(\tau) = \frac{1}{|\mathcal{G}|} \sum_{g_j \in \mathcal{G}} R_m(g_j, \tau). \tag{3.9}$$

Note that similarly to other metrics, we still obtain a dataset-level metric, however, this score assigns an equal weight to all the GT moments in the dataset, regardless of the number of moments that co-occur with it in the same video-query pair. This is key as otherwise, a multi-moment query comprising of 10 GT moments would have the same weight as that of a single query mapping to a single moment. Fixing this issue is key to ensure a fair comparison across levels of specificity, as well as in general, to provide a more fine-grained evaluation that looks at performance from a per-GT perspective, instead of a query-level one.

3.3.3.5 Multi-moment mAP (mAP_m)

Similarly to R1, mAP has the fundamental limitation that it also produces a query-video level score. Hence, it obscures the performance of the potentially multiple GT moments that may correspond to such query, as the good

detection of a given moment can mask poor detections or even GT moments that were not detected at all. As argued in Sec. 3, this breaks comparability in our setup, even though we argue that this limitation also extends to the evaluation of multi-moment queries in general.

To overcome this issue, similarly to R_m we propose evaluating the detection performance of each of the GT moments, independently, ensuring that a good/bad detection on one GT moment does not interfere with the scores of any other co-occurring moments.

Accordingly, for a given GT moment g_j, we define the set of true positives TP predictions as:

$$TP_j = \{p_i \mid match(p_i, g_j, \tau) = 1\}. \tag{3.10}$$

The false positives, being the predictions that do not match any GT moment is defined as:

$$FP_j = \{p_i \mid match(p_i, g_j, \tau)) = 0 \ \wedge \ match\_other(p_i, g_j, \tau) = 0\}. \tag{3.11}$$

And finally, we define the set of predictions that are ignored since, even though they do not match the moment that is currently evaluated (g_j), they nevertheless match another valid moment (not g_j). Ignoring them prevents these predictions from penalizing the metric for g_j, as this should neither benefit this metric, nor penalize it as a mistake, when it is a perfectly valid prediction. Formally,

$$IGN_j = \{p_i \mid match\_other(p_i, g_j, \tau) = 1\} \tag{3.12}$$

With this, for each of the GT moments g_j we compute the corresponding precision P_j and recall R_j:

$$P_j(k) = \frac{\#\text{TP up to rank } k}{\#\text{TP up to rank } k + \#\text{FP up to rank } k} \tag{3.13}$$

$$R_j(k) = \frac{\#\text{TP up to rank } k}{1}, \tag{3.14}$$

and following the original work [19], compute the corresponding area-under-the-curve (AUC):

$$AP_m(g_j, \tau) = \text{AUC}(P_j, R_j) \tag{3.15}$$

Similarly to R_m, we also compute a dataset level score as:

$$mAP_m(\tau) = \frac{1}{|\mathcal{G}|} \sum_{g_j \in \mathcal{G}} AP_m(g_j). \tag{3.16}$$

This again results in a score where each of the GT moments in $\mathcal{G}$ have an equal weight, making comparisons across levels of specificity fair, as the score of the detection quality for a given GT moment g_j is independent to the moments that it co-occurs with.

3.4 Search-Based VMR

Here, we analyze how current VMR methods trained on caption-based datasets perform when evaluated on search queries, and how to mitigate the multi-moment gap.

3.4.1 Evaluating caption-based models

In our experiments, we evaluate two representative models—i.e., CG-DETR [162] and LD-DETR [211]—on the three proposed benchmarks. For each, we train on the training set of the corresponding caption-based dataset and evaluate on the same test set for both the original caption-based dataset and our proposed search-query benchmark—e.g., train on the training set of HD-EPIC, and evaluate on the test set of HD-EPIC and HD-EPIC-S2. For ANC-S and YC2-S we leverage the original splits from [58, 143], while for HD-EPIC we create an 80-20 train/test split.

Figure 3.5 (left) shows the progressive performance decay across HD-EPIC-S1/S2/S3 benchmarks, with a relative degradation of up to 71.75% and 77.40% of R_m@0.3 for CG-DETR and LD-DETR, respectively. Similar trends are observed on ANC-S (center) and YC2-S (right) with drops of up to 31.8% and 60.7% of R_m@0.3, respectively. These drops evidence a substantial shift between caption-based queries and search queries, showing that existing models, when trained solely on descriptive caption-based queries, significantly degrade on less detailed search queries. This inevitably hinders the deployment of existing VMR systems in real-life scenarios (see Sec. 3.5.3 for the full results).

Below, we isolate two main causes of this degradation:

- Language gap: The linguistic shift between caption and search queries, characterized by missing visual details, looser references or the use of more abstract nouns—e.g., *"food"* instead of *"green peppers"*.
- Multi-moment gap: The gap arising from the mismatch between training on single-moment queries and evaluating on multi-moment ones, as under-specified search queries often correspond to multiple moments.

To quantify the effect of these factors, we partition the test set of the search queries into two subsets: $\mathcal{D}^{search}_{single}$ containing under-specified queries that map to a single GT moment, and $\mathcal{D}^{search}_{multi}$ containing search-queries that map to multiple GT moments.

For a fair one-to-one comparison across specificity levels, we also partition the original caption-based dataset $\mathcal{D}^{caption}$ using the same moment correspondence. Thus, $\mathcal{D}^{caption}_{single}$ and $\mathcal{D}^{caption}_{multi}$ contain the same moments, as $\mathcal{D}^{search}_{single}$ and $\mathcal{D}^{search}_{multi}$, respectively, but paired with their more specific captions. This yields our evaluation setup:

$$(\mathcal{D}^{caption}_{single}, \mathcal{D}^{search}_{single},) \text{ and } (\mathcal{D}^{caption}_{multi}, \mathcal{D}^{search}_{multi},)$$

allowing us to measure degradation due to purely linguistic changes ("single" split) versus the compounded effect of linguistics and multi-moment mapping ("multi" split).

Figure 3.6 reports the performance of CG-DETR on the three benchmarks after the decoupling of "single" and "multi" instances. For HD-EPIC, the language gap (performance on single) increases as we progressively evaluate more under-specific search queries, dropping from 15.9% to 49.6% with respect to the original mAP_m@0.3. Importantly, the compounded effect of the language and multi-moment gaps (performance on "multi") aggravates this effect further, reaching a degradation on HD-EPIC-S3 of 73.8% compared to the original mAP_m@0.3. This highlights the significant additional impact on performance of the multi-moment gap. These observations remain consistent across all benchmarks.

We next focus on addressing this multi-moment gap, an aspect largely under-explored in the VMR literature and key to the model architecture design. We leave addressing the language gap for future work, as it may be resolved with more advanced vision-language models, capable of reasoning across varying levels of specificity.

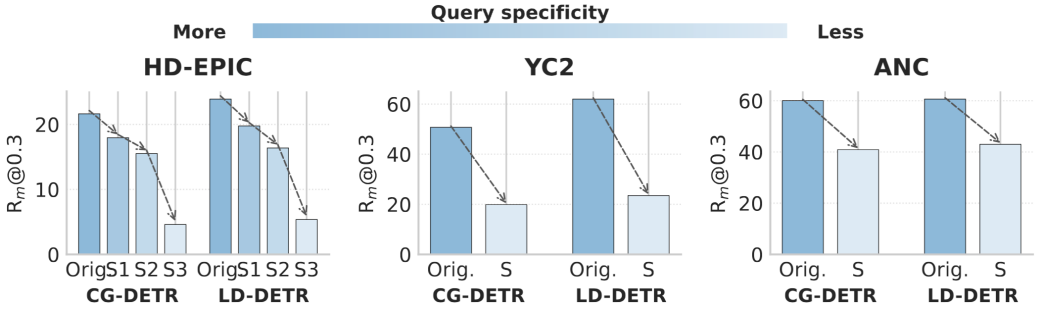


Figure 3.5: Evaluation of the representative models on both the original datasets and their corresponding search query extensions.

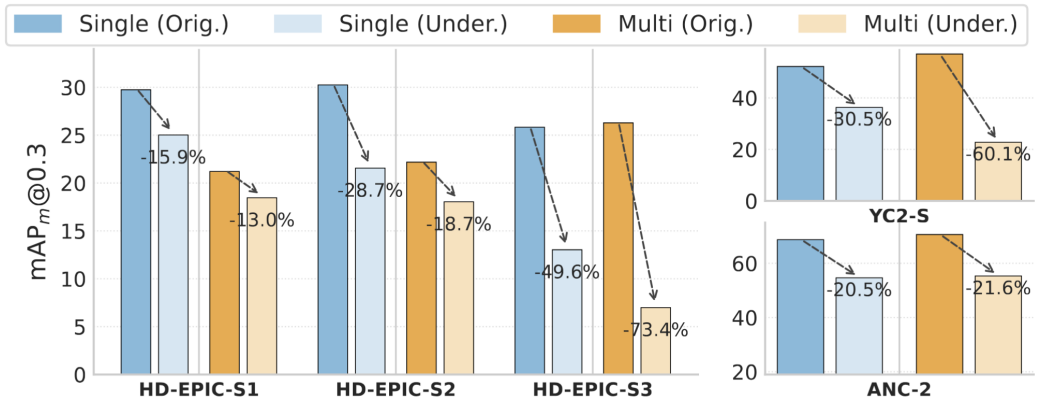


Figure 3.6: Performance degradation for CG-DETR on caption versus search-based evaluation for the "single" and "multi" splits.

3.4.2 Mitigating the multi-moment gap

In this section, we analyze the underlying causes of the performance degradation of VMR models on multi-moment query setups. We argue that this degradation mostly stems from misalignment between caption-based training data—characterized by a single relevant moment as GT—and search-based evaluation data, which frequently contains multiple valid moments. This discrepancy induces a strong single-moment prior—i.e., a bias towards expecting a single GT moment per video-query pair. One could attempt to resolve this by curating more diverse training data, which is impractical due to annotation costs and the uncertain feasibility of devising true search-query datasets. We instead approach this issue purely from a model's perspective which allows us to reuse all existing VMR training regimes.

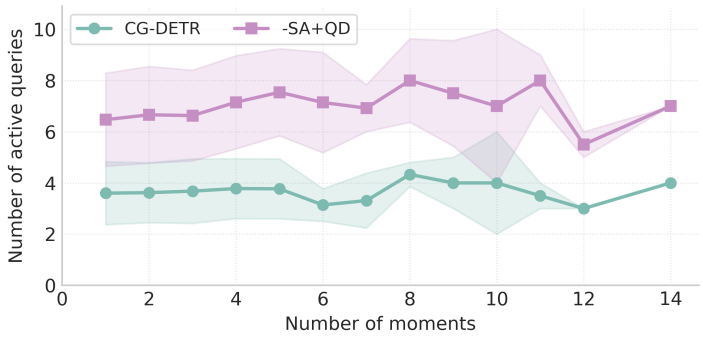


Figure 3.7: Visualization of the active query collapse on HD-EPIC-S2 for the base CG-DETR and our method -SA+QD.

3.4.2.1 Implications of a single-moment prior

We next analyze why DETR-based methods trained on single-moment queries struggle with multi-moment queries at inference time. Concretely, since each decoder query produces a candidate moment, analyzing the number of active decoder queries—i.e., those whose confidence does not vanish—directly reflects the model's capacity to retrieve multiple moments. In VMR, the number of active decoder queries can also be thought as the "compute budget" available to retrieve all the GT moments. When the number of active queries does not scale with the number of moments of the target instance, the model cannot retrieve all moments—e.g., when only 2 queries are activated in a 4-moment instance, the upper bound of retrieved moments is 50%. We term this phenomenon *active decoder-query collapse*.

As shown in Fig. 3.7, this phenomenon indeed affects VMR methods. Concretely, VMR methods trained on standard caption-based datasets (blue line) yield an insufficient number of active decoder queries when evaluated on search queries—around 4, regardless of the number of moments of the search queries (x-axis). This stems from the single-moment training, which impedes generalization beyond queries mapping to 4 moments.

3.4.2.2 Addressing the active decoder-query collapse

Having identified the *active decoder-query collapse* as a key limitation for generalization to multi-moment queries, we explore whether this can be mitigated purely from a model perspective, without altering the standard VMR training regimes and datasets. We find that this is possible by preventing models from overfitting to the single-moment prior encoded in the training

data. Specifically, we identify two structural causes of this prior fitting: (1) Coordination collapse, where the self-attention mechanism causes decoder queries to suppress one another, and (2) Index collapse, where a fixed, small subset of decoder query indices dominate activation.

In the following, we introduce architectural modifications that retain the model's capacity to learn the VMR task, while mitigating these forms of collapse.

Coordination collapse: The first cause arises from how the self-attention (SA) within each decoder layer enforces coordination among decoder queries. This drives them to "agree" on which query should handle the GT moment and which should remain inactive.

Formally, a standard decoder layer can be defined as:

$$\hat{Q}^{l+1} = FFN(CA(SA(\hat{Q}^{l}), M) \tag{3.17}$$

where $M \in \mathbb{R}^{T \times F}$ are the fused multi-modal features, CA denotes cross-attention and FFN a feed-forward network.

As noted by [152], the CA module injects the cross-modality information, while the role of SA is pushing decoder queries apart from each other to avoid redundancy. However, unintentionally, this also drives the majority of decoder queries to deactivate.

Interestingly, we find that an effective way of overcoming this issue is removing this SA module altogether, while leaving the losses unchanged:

$$Q^{l+1} = FFN(CA(Q^{l}, M) \tag{3.18}$$

Eliminating this inter-query communication prevents these coordination-based shortcuts, encouraging each decoder query to act independently. However, this also removes the model's built-in mechanism for avoiding redundant predictions. We address this by applying Non-Maximal-Suppression (NMS) during post-processing, which filters out overlapping/redundant predictions.

Index collapse: Mitigating the coordination collapse alone is insufficient as the model is still able to fit the single-moment prior, and thus still suffers active decoder-query collapse. The reason resides in an *index collapse*, where the same decoder query indices repeatedly dominate the output confidence, while the rest remain inactive—e.g., decoder queries with index 1–4 are the only ones ever activating. During training, the single-moment prior drives the model to associate the detection of the single GT moment with only a

handful of fixed decoder query indices based on their learnable initializations. This dominance is progressively reinforced throughout training, leaving the rest of the decoder queries permanently inactive and thus, unused.

We counter this effect by applying a targeted query dropout strategy, which randomly zeroes out $k\%$ of the learnable queries $Q \in \mathbb{R}^{Q \times F}$ during each training iteration:

$$\hat{Q} = Q \odot M, \quad M \sim \mathbb{B}(1-k) \tag{3.19}$$

where $\mathbb{B}$ is sampling from the Bernoulli distribution with keep probability $(1-k)$. This regularization promotes the model to distribute supervision across more queries, preventing over-reliance on a fixed subset.

Together, these modifications considerably reduce the number of permanently inactive indices, resulting in a consistent increase in the number of active decoder queries (see Fig. 3.7 orange line), boosting search-query generalization.

3.5 Experimentation

3.5.1 Experimental setup

The following experiments evaluate how baselines trained on a caption-based dataset generalize to search-based benchmarks. Specifically, we evaluate HD-EPIC-S, YC2-S and ANC-S; and report the R_m and mAP_m, on IOU $\{0.1, 0.3, 0.5\}$.

3.5.2 Implementation details

In this section we provide the necessary details required for reproducibility. These include the search-query pipeline, the three proposed search-query benchmarks, and the implementation and optimization details of the evaluated baselines.

3.5.2.1 Search-query pipeline

Under-specification stage: As described in Sec. 3.3.2, the Rewriter obtains the under-specifications of the original caption-based queries using an LLM agent based on Gemma3-12b-it. We choose this model due to its open-source availability and its good performance for this task compared to other LLMs we evaluated.

The rewriter follows an in-context learning strategy as a way to guide the agent towards the desired levels of specificity. Concretely, for each of the benchmarks we pass a small set of examples that are manually annotated by a human annotator. It is key that these examples are benchmark-specific so as to avoid the shift between the distribution of the in-context examples and that of the instances that we aim to under-specify.

3.5.2.2 Grouping stage

The grouping stage is further divided into two different steps:

Similarity-based grouping: For each video, we compute the STSB-Roberta-Large sentence embeddings [1] of all the queries that occur in this video. We then compute their pairwise cosine similarities and form a graph where the nodes are the queries, and where two nodes are connected if their corresponding queries present a similarity equal or greater to 0.85. Given these connections, we use a DFS algorithm to form the groups/clusters that essentially contain the same semantics.

Representative search query generation: Since members of the same group/cluster can still present some minor differences in their corresponding under-specified queries, we generate a single representative search query that corresponds to all of them. For instance, consider two very similar under-specifications *"hold the pan"* and *"hold the pot"*. Despite their similarity, the first includes unique information, that is not present in the seconds, and vice-versa. Thus, to create a single search query that corresponds to all the members of the group—i.e., containing shared semantics only—we employ a final identical Gemma3-12b-it agent that takes all the corresponding under-specifications of a given group and computes their final search query—e.g., *"hold the kitchenware"* following the previous example.

3.5.2.3 Search-query benchmarks

Below we provide additional details on the evaluation of each of the benchmarks:

HD-EPIC-S1/S2/S3: We extract InternVideo2 [195] features at an FPS rate of 3. Because HD-EPIC contains very long videos, we find that a naive evaluation on the entire videos yields uninformative results given the extremely low baseline scores. Since long-video VMR detection is beyond the scope of this chapter, even though this constitutes a promising line of research, we trim the videos into 500-frame windows, treating each of them as independent instances. We further discard the windows that do not

contain any GT moment, avoiding the important issue of dealing with the assumption that every instance contains at least one GT moment [203]. This is a relevant issue that falls beyond the scope of this chapter, mainly because the majority of baselines, including the ones we evaluate on do not provide built-in mechanisms to deal with these situations.

Moreover, HD-EPIC does not provide pre-defined data splits suitable for VMR. Hence, we construct a training and testing split using an 80-20 split of the per-participant original data.

YC2-S: We extract InternVideo2 features at an FPS rate of 3, and use the original training and validation splits as proposed by [84].

ANC-S: We extract InternVideo2 features at an FPS rate of 3, and use the original training and validation splits as proposed by [58].

3.5.2.4 Models and optimization

As described before, we select CG-DETR and LD-DETR as representatives of the DETR-based VMR models. These models are trained on a single NVIDIA GeForce RTX 3090. We use all the original hyper-parameters from [162] and [211], respectively, across all the benchmarks. Our method (-SA+QD) only introduces one new hyperparameter, being the QD rate, selected via grid search and kept as 0.25 across all the benchmarks and models.

3.5.3 Main experiments

Do our proposed modifications bridge the generalization gap to search queries? From the results, we observe that our proposed architectural modifications, hereby denoted as (-SA+QD), substantially improve performance across all search-query datasets. For instance, on HD-EPIC-S2 (see Tab. 3.2) these modifications increase R_m@0.1 from 24.71 to 26.71 and the mAP_m@0.1 from 32.15 to 35.38. Similarly, on YC2-S (see Tab. 3.3) we observe an absolute improvement of up to 2.96 mAP_m@0.3. Moreover, even with the smaller multi-moment gap for ANC-S, we observe that our modifications lead to comparable or improved performance across all metrics (see Tab. 3.4).

Even though in this chapter we argue that the standard mAP metric does not provide a fair evaluation of our setup, we find it helpful to additionally include standard mAP results completeness and comparability with prior work. Observe that these results show a similar trend to our proposed metrics,

Model	Input	Variant	R_m			mAP_m			mAP		
			@0.1	@0.3	@0.5	@0.1	@0.3	@0.5	@0.1	@0.3	@0.5
CG-DETR	Original	base	34.44	21.63	11.32	42.69	26.96	14.26	42.69	26.96	14.26
		-SA+QD	34.24	22.68	12.59	45.79	30.52	17.16	45.79	30.52	17.16
	S1	base	28.61	17.95	8.99	36.21	22.84	11.59	38.52	24.33	12.27
		-SA+QD	29.87	19.69	10.86	39.74	26.49	14.87	41.65	27.98	15.85
		base (oracle)	28.85	17.44	9.08	40.42	25.12	13.1	42.58	26.74	14.02
	S2	base	24.71	15.52	7.89	32.15	20.1	10.29	34.19	21.29	10.79
		-SA+QD	26.17	17.00	9.40	35.38	23.39	13.04	36.97	24.68	13.80
		base (oracle)	27.34	17.47	8.90	39.82	25.79	13.22	42.13	27.31	14.00
	S3	base	9.50	4.61	2.08	16.20	8.01	3.58	20.99	11.57	5.29
		-SA+QD	10.57	6.52	3.45	17.27	10.65	5.54	22.07	14.19	7.86
		base (oracle)	12.31	6.09	3.06	23.29	10.94	5.11	30.56	15.3	7.37
LD-DETR	Original	base	34.75	23.90	13.46	42.59	29.17	16.42	42.59	29.17	16.42
		-SA+QD	35.33	24.51	13.37	46.83	32.52	18.01	46.83	32.52	18.01
	S1	base	29.42	19.77	10.50	36.55	24.5	13.18	38.94	26.25	14.29
		-SA+QD	30.18	20.26	10.83	40.50	27.54	14.94	42.58	29.15	16.08
		base (oracle)	29.92	18.61	8.74	41.5	26.33	12.78	43.73	28.00	13.61
	S2	base	25.23	16.38	8.46	32.42	21.11	10.93	34.32	22.58	11.89
		-SA+QD	26.36	16.98	8.87	36.37	23.75	12.54	38.24	25.20	13.56
		base (oracle)	29.78	20.82	11.76	41.93	29.43	16.77	44.11	31.16	17.85
	S3	base	10.44	5.37	2.58	16.48	8.65	4.11	18.63	9.41	4.32
		-SA+QD	10.44	5.28	2.39	17.79	9.06	4.19	21.02	10.11	4.47
		base (oracle)	10.35	5.61	2.67	20.59	11.14	5.16	26.97	13.28	5.63

Table 3.2: Results of both CG-DETR and LD-DETR on HD-EPIC-S 1,2,3 benchmarks with respect to our proposed modifications.

highlighting the consistent gains that our method attains in terms of mAP across all the benchmarks.

How do these gains compare to an oracle model? To contextualize these gains, we also compare against an oracle of the base model for HD-EPIC-S1/S2/S3. The oracle corresponds to training the base architectures—CG-DETR and LD-DETR, respectively— directly on the data at the target specificity. For example, for HD-EPIC-S2, the oracle is obtained by training the baseline model on the training split of HD-EPIC-S2 derived from our proposed search-query pipeline. While this is not aligned with the underlying goal of this work, this being training on the standard captions while generalizing to more under-specified search queries, this still provides a meaningful upper-bound on the achievable performance.

More specifically, Tab. 3.2 unveils how our proposed (+SA-QD) significantly closes the gap between base and oracle model. For instance, on HD-EPIC-S2, our proposal closes the gap by up to 82% of mAP_m@0.1. This translates to a recovery of up to nearly 70% of the oracle gap, confirming the effectiveness of

Model	Input	Variant	R_m				mAP_m				mAP			
			@0.1	@0.3	@0.5	@0.75	@0.1	@0.3	@0.5	@0.75	@0.1	@0.3	@0.5	@0.75
CG-DETR	Orig.	base	63.32	50.71	37.79	20.11	69.47	55.55	41.04	17.18	69.47	55.55	41.04	17.18
		-SA+QD	62.46	51.05	37.19	20.27	70.25	57.97	42.79	18.05	70.25	57.97	42.79	18.05
	S	base	28.92	19.87	11.22	4.60	38.83	26.96	15.21	4.07	47.74	33.61	19.40	5.27
		-SA+QD	29.97	20.32	11.38	4.36	41.00	29.4	17.21	4.65	49.52	36.41	21.86	6.33
LD-DETR	Orig.	base	68.06	56.34	39.75	19.69	73.15	60.62	42.79	15.88	73.15	60.62	42.79	15.88
		-SA+QD	70.20	55.66	37.06	17.63	76.35	61.71	42.01	15.05	76.35	61.71	42.01	15.05
	S	base	33.13	23.48	11.70	4.44	41.69	30.04	15.58	4.09	51.86	37.85	20.0	5.45
		-SA+QD	35.86	24.76	13.17	5.15	45.66	33.09	18.74	4.90	56.05	41.26	23.89	6.14

Table 3.3: Results of both CG-DETR and LD-DETR on YC2-S with respect to our proposed modification.

Model	Input	Variant	R_m				mAP_m				mAP			
			@0.1	@0.3	@0.5	@0.75	@0.1	@0.3	@0.5	@0.75	@0.1	@0.3	@0.5	@0.75
CG-DETR	Orig.	base	75.48	60.00	44.02	26.21	82.31	69.36	53.15	25.7	82.31	69.36	53.15	25.7
		-SA+QD	75.19	60.30	44.96	26.47	82.25	69.91	54.5	25.67	82.25	69.91	54.5	25.67
	S	base	60.44	40.89	24.56	12.97	72.18	54.9	36.41	15.07	73.34	55.84	37.49	15.71
		-SA+QD	63.75	43.12	25.50	13.36	74.00	56.42	37.2	15.09	75.54	57.52	38.52	15.78
LD-DETR	Orig.	base	75.72	60.63	45.17	27.15	82.73	70.3	54.46	26.62	82.73	70.3	54.46	26.62
		-SA+QD	76.72	61.44	45.68	27.87	83.15	70.97	55.52	27.29	83.15	70.97	55.52	27.29
	S	base	62.58	43.00	26.08	13.92	73.35	56.17	36.79	15.16	74.65	57.15	37.93	15.93
		-SA+QD	65.21	43.89	25.77	13.36	74.25	56.31	36.69	15.15	76.13	57.58	37.88	15.88

Table 3.4: Results of both CG-DETR and LD-DETR on ANC-S with respect to our proposed modification.

(-SA+QD) in bridging the multi-moment gap, thus improving generalization to search queries.

Where do these gains come from? To disentangle the benefits of our proposed modifications, we separately evaluate single and multi-moment instances. Figure 3.8 shows that while performance of (-SA+QD) on single-moment queries improves modestly, in most cases there is a prominent improvement on multi-moment queries by up to 34.3% mAP_m@0.3. This confirms that our method specifically benefits multi-moment queries while also improving single-moment cases.

3.6 Ablations

Below, we ablate key aspects of our findings, reporting results for CG-DETR on HD-EPIC-S2. We report the average R_m and mAP_m across IoU values $\{0.1, 0.3, 0.5\}$.

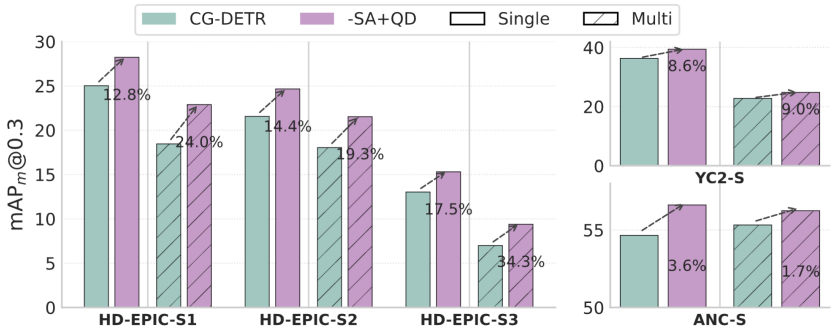


Figure 3.8: Dissection of the performance of CG-DETR on HD-EPIC-S2, for single and multi respectively.

3.6.1 Quantitative analysis:

1) Alternative methods for decoder query activation: We examine whether alternative methods can increase the number of active queries and yield comparable gains to our proposal. Specifically, we evaluate two families of approaches (see Tab. 3.5): (1) alternative matching strategies that provide supervision to multiple decoder queries, and (2) a data-augmentation scheme that simulates multi-moment setups.

While matching strategies increase the number of active queries, these activated queries produce redundant predictions—predicting nearly identical segments around the same GT moment. This can be observed in the number of predictions overlapping a GT (%match P), which nearly doubles, while the number of retrieved GT moments (%match GT) decreases, leading to a drop in generalization.

Similarly, data-augmentation techniques that replicate GT moments in different video locations also fail to improve generalization. We attribute this to the disruption of temporal coherence, which leads to overfitting.

Overall, these results confirm that merely increasing the number of active queries through additional supervision is insufficient; effective generalization to multi-moment setups also requires diversity-promoting mechanisms that encourage complementary behavior in decoder queries.

2) Preserving diversity via 1-to-1 matching: The previous ablation showed that merely activating more queries does not improve generalization if diversity is not preserved. Our proposal addresses this, increasing the number of active queries while also maintaining diversity among them. This is achieved by keeping the 1-to-1 matcher [94]. This strategy enforces competition between

Variant	R_m @0.1	R_m @0.3	R_m @0.5	mAP_m @0.1	mAP_m @0.3	mAP_m @0.5	# active	% match P	% match GT
base	24.71	15.52	7.89	32.15	20.1	10.29	3.64±1.18	0.06±0.07	0.36±0.35
+ 1-to-5 matching [129]	24.00	15.11	7.97	29.35	18.36	9.60	9.56±3.20	0.11±0.16	0.21±0.28
+ 1-to-k matching [129]	18.31	8.80	4.12	18.47	8.87	4.16	20.00±0.0	0.16±0.33	0.07±0.18
+group_matching [150]	24.33	15.85	8.86	31.21	20.12	11.14	8.69±3.08	0.10±0.13	0.27±0.31
+hybrid matching [155]	23.98	15.19	7.75	30.53	19.07	9.71	8.68±2.90	0.10±0.13	0.28±0.32
+ms_matcher [201]	24.13	15.30	8.02	31.72	20.35	10.79	3.58±1.14	0.06±0.07	0.36±0.34
+data_augmentation	20.96	13.14	7.31	31.64	20.52	11.55	4.68±1.78	0.07±0.08	0.38±0.35
-SA+QD (ours)	**26.17**	**17.00**	**9.40**	**35.38**	**23.39**	**13.04**	6.43±2.16	0.11±0.13	0.42±0.37

Table 3.5: Ablation of methods to increase number of active queries.

Variant	R_m @0.1	R_m @0.3	R_m @0.5	mAP_m @0.1	mAP_m @0.3	mAP_m @0.5	# active	% match P	% match GT
-SA+QD (ours)	26.17	17.00	9.40	35.38	23.39	13.04	6.43±2.16	0.11±0.13	0.42±0.37
+ 1-to-k matcher [129]	17.88	9.18	4.10	17.89	9.19	4.11	20.00 ±0.00	0.14±0.35	0.06±0.17
+group_matching [150]	25.72	17.17	9.02	35.36	23.37	12.41	12.70±6.13	0.15±0.17	0.43±0.36
+hybrid matching [155]	**26.59**	**17.60**	**9.55**	**35.79**	**24.13**	**13.24**	10.10±6.23	0.14±0.16	0.50±0.36

Table 3.6: Effect of 1-to-1 matching in promoting diversity.

decoder queries, preventing them from collapsing into a redundant prediction. As shown in Tab. 3.6, replacing it from our (-SA+QD) for a pure 1-to-k matcher [178] leads to redundant predictions, as numerous queries receive the same supervision signal. In contrast, partial relaxations that retain the 1-to-1 matching—e.g., [150, 155]—preserve competition and yield comparable results. This highlights the crucial role of the 1-to-1 matching to ensure that queries that are additionally activated by (-SA+QD) remain diverse and contribute to generalization.

3) Effect of each component: Table 3.7 evaluates variants that only include query-dropout (+QD) or only remove self-attention (-SA). Observe that neither component alone yields a significant performance gain as, by themselves, they do not overcome the query collapse—increasing only marginally the number of active queries. Combining both, in turn, increases mAP_m by up to 3.09 while nearly doubling the number of active queries. This confirms the need of solving both coordination and index collapse jointly.

4) Ablation on the query dropout rate: In Tab. 3.8 we ablate over the effect of various query-dropout rates—i.e., $k = \{0.0, 0.25, 0.5\}$. Observe that performance peaks at 0.25, after which performance decays. This confirms that a light stochastic regularization encourages broader query utilization, without compromising model convergence.

5) Scaling the number of potential queries: Figure 3.9 investigates how the increase of the total number of decoder queries influences the number of

-SA	+QD	R_m @0.1	R_m @0.3	R_m @0.5	mAP_m @0.1	mAP_m @0.3	mAP_m @0.5	# active
		24.71	15.52	7.89	32.15	20.10	10.29	3.64±1.18
✓		23.97	14.73	7.25	32.17	20.58	10.33	3.72±1.16
	✓	24.45	16.24	8.83	31.58	21.07	11.66	3.77±1.28
✓	✓	**26.17**	**17.00**	**9.40**	**35.38**	**23.39**	**13.04**	6.43±2.16

Table 3.7: Impact of the proposed architectural modifications.

k	R_m @0.1	R_m @0.3	R_m @0.5	mAP_m @0.1	mAP_m @0.3	mAP_m @0.5
0.00	24.71	15.52	7.89	32.15	20.10	10.29
0.25	**26.17**	**17.00**	**9.40**	**35.38**	**23.39**	**13.04**
0.50	1.82	0.86	0.31	6.72	3.49	1.32

Table 3.8: Effect of the QD dropout rate.

active queries as well as performance. The base model quickly saturates. The number of active queries remains nearly constant, and performance severely degrades after peaking at 20 queries. In contrast, our method presents a more steady increase in the number of active queries and performance (mAP_m@0.1). This trend holds up until 20 queries, after which performance stabilizes.

5) Impact of calibration in the query collapse: In this ablation we examine if existing confidence calibration methods could resolve the query collapse issue. To this end, Fig. 3.10 reports, for each of the learnable queries, the correlation between its regression quality—i.e., measured as the proportion of times it achieves an IOU of at least 0.1 with a GT segment—and its confidence score.

The plot reveals that the issue does not stem from confidence scores that fail to reflect the true quality of the regression estimate—i.e., marking as inactive, queries that in fact produce accurate predictions. Instead, what we observe is that inactive queries—i.e., with low confidence scores—produce substantially worse regression estimates. Thus, to some extent, confidence scores do capture the true quality of the regression estimate. Therefore, the core problem lies not in calibration, but in the lack of mechanisms to encourage more queries to produce accurate moment predictions.

To further support this claim, in Tab. 3.9 we evaluate several confidence calibration mechanisms [186, 134]. These results demonstrate how these methods actually lead to a performance degradation, reinforcing that calibration alone cannot overcome active-query collapse.

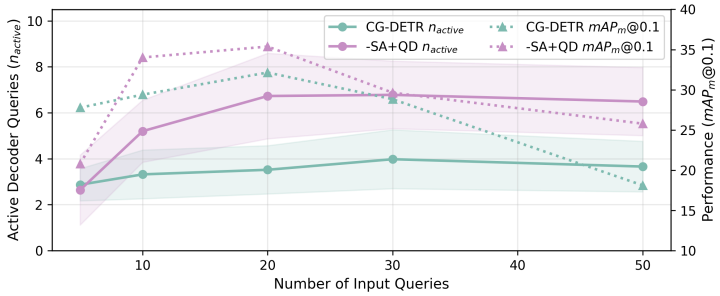


Figure 3.9: Evolution of the number of active queries and performance over the total number of decoder queries.

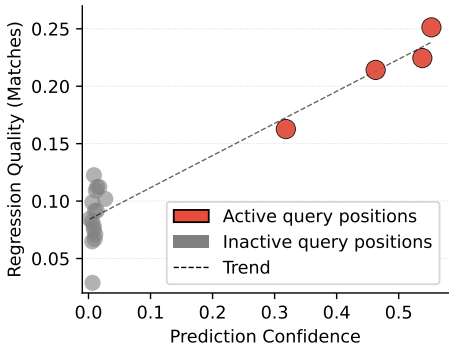


Figure 3.10: Correlation between the average ratio of matched predictions—i.e., predictions with an IOU of at least 0.1 with one of the GT moments—with respect to their respective confidence score. This highlights the tradeoff between regression quality and confidence score quality. These results correspond to HD-EPIC-S2 for CG-DETR. .

3.6.2 Qualitative study

3.6.2.1 Qualitative results of the generated search queries

Here we show various qualitative examples of the search queries generated by our proposed search-query pipeline. More concretely, for each of the benchmarks, we first showcase various examples of the under-specifications resulting from the original caption-based queries. Then, we include various instances of final search queries, showing all the original captions that resulting from the search-query pipeline, end up mapping to the same representative search query.

HD-EPIC As explained in Sec. 3.5.2, thanks the great detail of the captions of this dataset, we were able to extract 3 different levels of under-specified

Variant	R_m			mAP_m			active
	@0.1	**@0.3**	**@0.5**	**@0.1**	**@0.3**	**@0.5**	
base	24.71	15.52	7.89	32.15	20.1	10.29	3.64±1.18
+actionness[134]	26.04	16.05	8.11	32.26	19.96	10.23	3.79±0.98
+SFL[186]	21.83	11.22	4.82	27.54	14.48	6.31	3.05±0.99
-SA+QD (ours)	**26.17**	**17.00**	**9.40**	**35.38**	**23.39**	**13.04**	6.43±2.16

Table 3.9: Performance of alternative confidence calibration mechanisms for CG-DETR on HD-EPIC-S2.

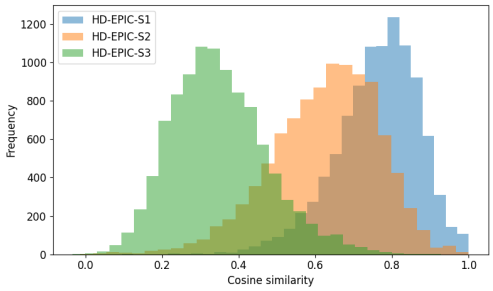


Figure 3.11: Histogram of the feature similarities of each of the test set of each of the levels of under-specification HD-EPIC-S1/S2/S3 with respect to the original caption-based queries from HD-EPIC

search queries—S1,S2 and S3. Figure 3.11 further depicts the effect of these under-specifications by visualizing the features similarities of the under-specified search queries with respect to the original caption-based ones.

In Tab. 3.10, moreover, we present multiple qualitative examples of how a caption-based query progressively under-specifies in each of the proposed levels of specificity. This is further shown in Tab. 3.11, Tab. 3.12 and Tab. 3.13 where we show various final search queries, including the final search query and the multiple original captions that match it.

YC2-S Below we repeat the same analysis for YC2-S benchmark. Concretely, find in Fig. 3.12 the histogram of the features similarities between the original caption-based queries and the corresponding search queries. Additionally, Tab. 3.14 provides various qualitative examples of under-specifications of original captions from YC2 [84] into their corresponding search queries, while Tab. 3.15 shows various final search queries.

ANC-S Finally, we perform the same analysis for ANC-S benchmark. Concretely, find in Fig. 3.13 the histogram of the features similarities

Caption: *"Holding the ball of chicken and potato mixture in my left hand while I get some flour in my right hand so that I can sprinkle over the ball"*

↓

S1: *"Hold the mixture and get some flour to sprinkle over it"*

↓

S2: *"Sprinkle flour on the mixture"*

↓

S3: *"Prepare food"*

Table 3.10: Qualitative results of the under-specified search queries for HD-EPIC-S1/S2/S3.

Representative: *"Tear the lettuce and place it on the plate"*

↓

Caption1: *"Tear the lettuce leaves in half again and place onto the plate that is on top of the weighing scale ."*

Caption2: *"Tear the lettuce leaves in half again and put the pieces onto the plate ."*

Caption3: *"Use both hands to tear the lettuce leaves in half again and place the pieces onto the plate ."*

Caption4: *"Use both hands to tear the lettuce leaves in half ."*

Caption5: *"Use both hands to tear the lettuce leaf in half"*

Caption6: *"Use the left hand to put the lettuce leaves into the bowl and then use both hands to tear the lettuce leaves in half again ."*

Table 3.11: Example of search queries and the captions that match it for HD-EPIC-S1.

between the original caption-based queries and the corresponding search queries. Additionally, Tab. 3.16 provides various qualitative examples of under-specifications of original captions from ANC [58] into their corresponding search queries, while Tab. 3.17 shows various final search queries.

3.6.2.2 Qualitative VMR results

Below we showcase various qualitative examples that compare the performance of our proposed modification (-SA+QD) with respect to its corresponding baseline, CG-DETR. Concretely, Fig. 3.14 and Fig. 3.15 show two different

Representative: *"Open a lid"*
↓
Caption1: *"Open the lid of the trash bin by flipping it ."*
Caption2: *"Open the trash bin 's lid ."*
Caption3: *"Open the lid of the trash bin . This action occurs off the screen ."*
Caption4: *"Open the lid of the food waste bin ."*

Table 3.12: Example of search queries and the captions that match it for HD-EPIC-S2.

Representative: *"Open a book"*
↓
Caption1: *"Push down in the center of the recipe book along the spine to try and make sure it stays open onto the countertop."*
Caption2: *"Pick up the recipe book using the right hand and flipping it over so that the recipe can be seen."*

Table 3.13: Example of search queries and the captions that match it for HD-EPIC-S3.

Caption: *"Add salt to the pan and mix"*
↓
S:*"Season food."*

Table 3.14: Qualitative results of the under-specified search queries for YC2-S.

examples for each of the scenarios included in our proposed benchmarks—i.e., HD-EPIC-S1/S2/S3, YC2-S and ANC-S. Observe that in numerous examples, the base CG-DETR is unable to activate sufficient predictions with a non-vanishing confidence, which hinders the capacity to detect multi-moment queries as the number of active queries is smaller than the number of GT moments to retrieve. This is considerably mitigated by (-SA+QD) that consistently activates more decoder queries, thus showing considerably better behavior in these scenarios.

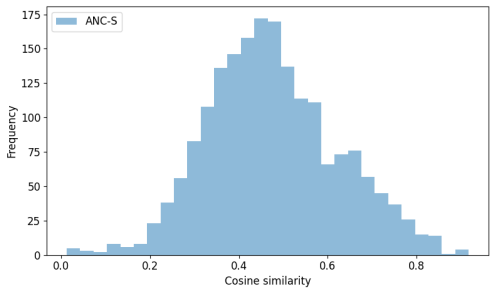


Figure 3.12: Histogram of the feature similarities of the test set of YC2-S with respect to the original caption-based queries from YC2

Representative: *"Add ingredients."*
↓
Caption1: *"Crack one egg into a bowl"*
Caption2: *"Add one table spoon of oil salt and cayenne pepper and baking powder and beat"*
Caption3: *"Add one cup of beer and mix"*
Caption4: *"Add one quarter cup of corn meal and one cup of flour"*
Caption5: *"Add onions into batter and drop into hot oil"*

Table 3.15: Example of search queries and the captions that match it for YC2-S.

Caption: *"The person then moves back and fourth on the machine while rowing his arms back and fourth."*
↓
S: *"A person uses a machine."*

Table 3.16: Qualitative results of the under-specified search queries for ANC-S.

3.7 Conclusions and future work

In this chapter, we revisited Video Moment Retrieval (VMR) from the perspective of linguistic OOD generalization, moving beyond the caption-based assumptions that dominate existing benchmarks. We introduced three search-query evaluation settings that expose a consistent degradation in performance when models trained on descriptive captions are evaluated on more realistic, under-specified user queries. Our analysis identified two key factors for this degradation: (1) a linguistic gap between detailed captions and

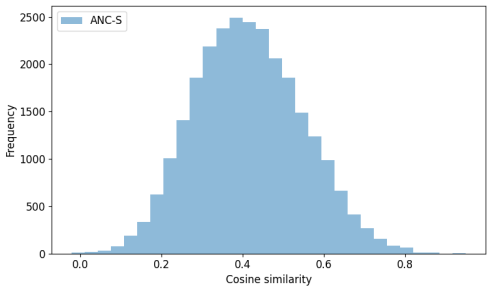


Figure 3.13: Histogram of the feature similarities of each of the test set of ANC-S with respect to the original caption-based queries from ANC

Representative: People perform household chores.
↓
Caption1: *"A young girl and boy are washing dishes in a kitchen."*
Caption2: *"They are doing the dishes in the sink."*
Caption3: *"the mother enters and adds some dishes from the rack back into the sink to be rinsed again and shows the boy what was wrong with the pot."*

Table 3.17: Example of search queries and the captions that match it for ANC-S.

concise search queries, and (2) a multi-moment gap induced by the shift from single-moment annotations to queries that may correspond to multiple relevant temporal segments. We further showed that this shift can trigger an active decoder-query collapse in proposal-free architectures, and proposed multiple architectural modifications that partially mitigate these effects. Overall, this chapter complements Chapter 2 by showing that generalization challenges in video understanding are not limited to visual domain shifts, but also arise from distribution shifts in the linguistic modality.

Future work: While this chapter focuses on under-specification as a key source of linguistic distribution shift, realistic search queries often exhibit additional linguistic aspects that remain largely unexplored in current VMR models and benchmarks, such as negation, redundancy, politeness markers, and conversational context. Addressing these aspects will likely require tighter integration between language modeling and video retrieval architectures. Additionally, note that in Chapters 2 and 3 we address generalization under OOD shifts in the visual and linguistic modalities, respectively, while deliberately remaining confined to the scope of an individual task (TAL

and VMR, respectively). A natural next step toward broader generalization is therefore going beyond such assumption, and investigating instead how the knowledge encoded in video-understanding models can be efficiently transferred and adapted across tasks. This perspective motivates the next chapter, which focuses on knowledge transferability and parameter-efficient adaptation.

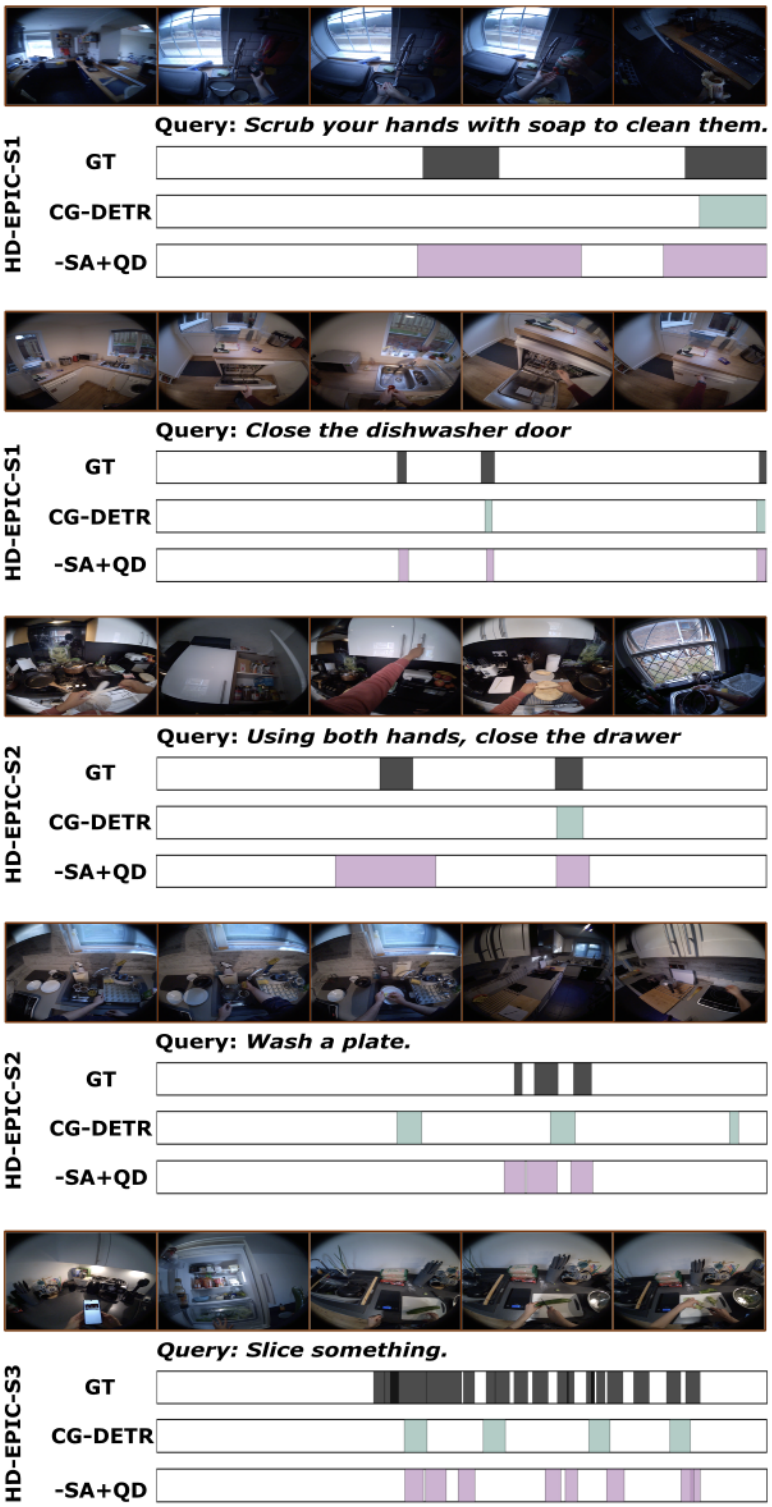


Figure 3.14: Qualitative results of the performance of CG-DETR and our proposed modification (-SA+QD) on HD-EPIC-S1/S2/S3.

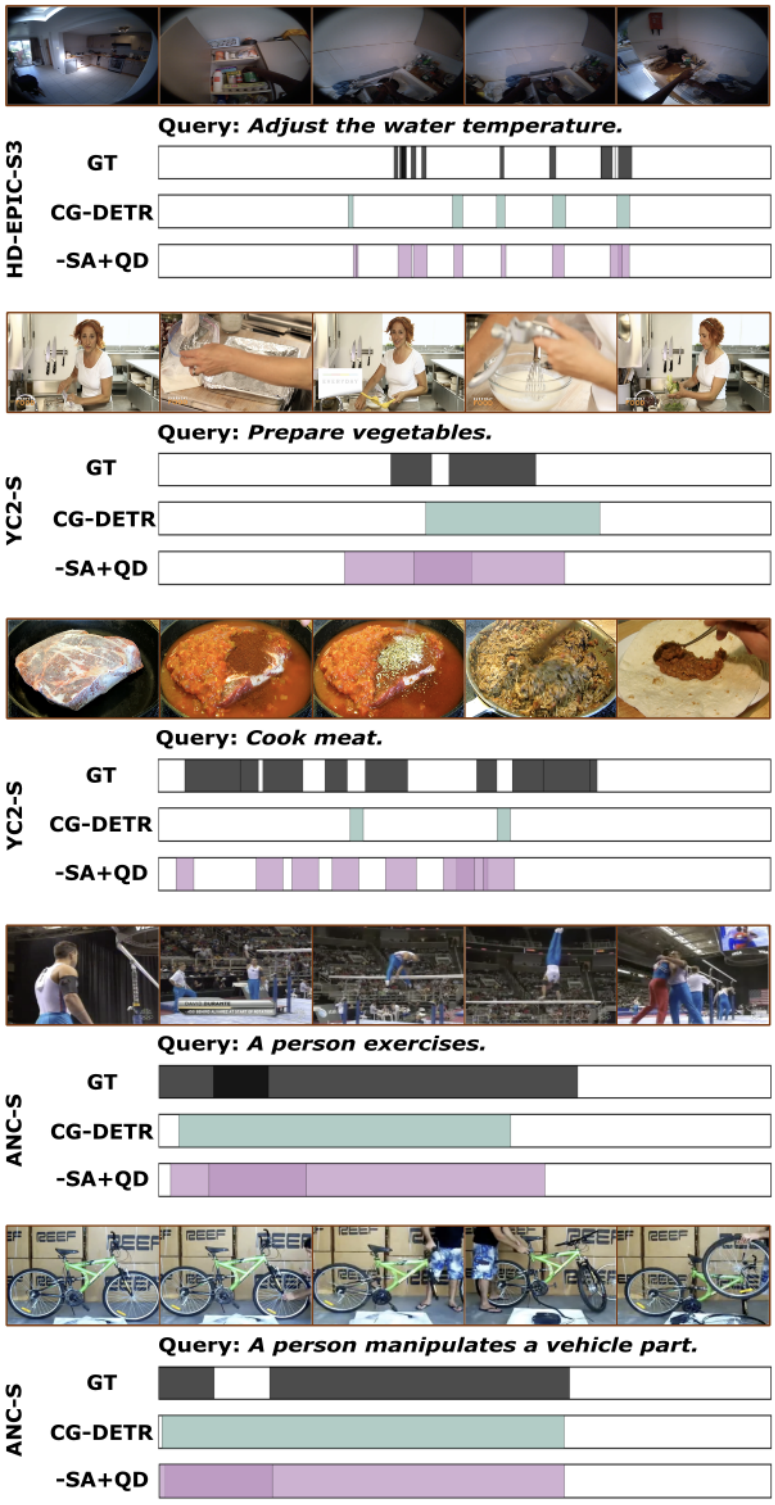


Figure 3.15: Qualitative results of the performance of CG-DETR and our proposed modification (-SA+QD) on HD-EPIC-S3, YC2-S and ANC-S.

Chapter 4

Sparse-Dense Side-Tuner for efficient Video Temporal Grounding

In previous chapters, we addressed various important challenges related to OOD generalization both in the visual (Chapter 2) and the linguistic domain (Chapter 3). Crucially, these studies focus on generalization within the same task. An equally important dimension of generalization, however, is knowledge transferability—i.e., the capacity to efficiently reuse the powerful representations extracted from a source model for a related target task.

The relevance of this dimension is highlighted by the rise of modern large-scale video foundation models. These models encode immensely rich spatio-temporal representations acquired through highly expensive pre-training processes over massive datasets. Despite the high quality of these features, effectively adapting them to downstream tasks remains a challenge because it often requires full fine-tuning, which is resource-intensive and computationally wasteful. In this chapter, we explore this final, crucial aspect of generalization. We investigate **Parameter-Efficient Fine-Tuning (PEFT)** methods to reuse and adapt existing representations to new, related video grounding tasks. For this, we introduce a lightweight fine-tuning strategy that preserves the robust, underlying knowledge learned from large-scale pre-training, while enabling rapid and cost-effective specialization to novel tasks.

4.1 Introduction

Recently the field of Video Understanding has gained attention due to its potential in applications like search engines or recommendation systems. A key task in this field is VMR, which localizes moments within videos based on textual descriptions. Typically, this involves doing Moment Retrieval (MR) [53, 154, 110, 181, 176] (as also done in Chapter 3), even though other works approach this task from a Highlight Detection (HD) [106, 98,

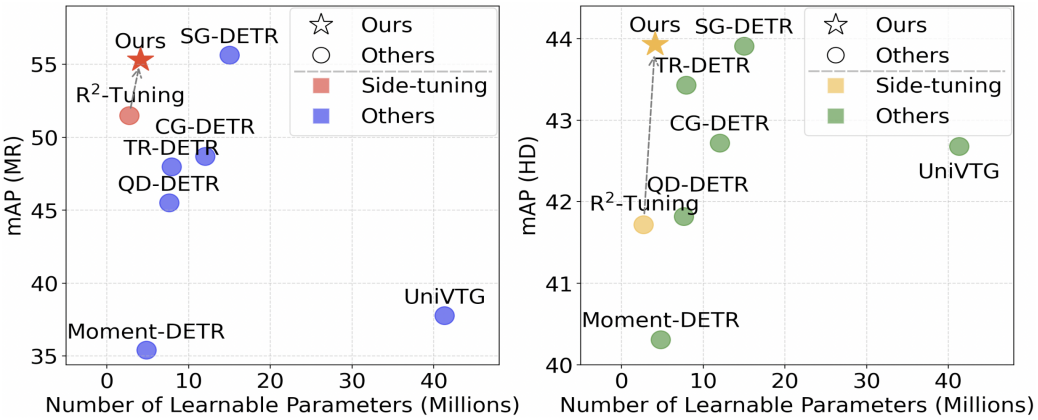


Figure 4.1: Comparison of our proposed method and the main MR (left) and HD (right) baselines. These are all evaluated on QVHighlights *val* split and use InternVideo2-1B features. These results show that our method improves existing side-tuning works and attains SOTA results while significantly reducing the number of trainable parameters.

29, 123, 184]. More specifically, MR predicts moment boundaries w.r.t. textual queries, while HD offers a more interpretable perspective, predicting frame-level saliency scores.

In the past, existing works focused on either MR or HD, given the lack of datasets that combined both sub-tasks. QVHighlights [110] shifted this paradigm, proposing the first dataset suitable for this multi-task setup. Since then, various works have been proposed following *anchor-based* [136, 160, 176, 187], *anchor-free* [110, 162, 181] and even LLM-based [188, 189] approaches. All of these, however, incur in a common critical limitation, which is relying on the final-layer features of frozen backbones. This is especially limiting when facing large distribution shifts between the pre-training distribution and that of the downstream task [207]. This issue is particularly pronounced in VMR, where an image-domain backbone [118] is transferred to the video domain.

A typical solution is fine-tuning; however, full fine-tuning is impractical due to its high computational cost. Doing this more efficiently has thus become critical, motivating the interest in *parameter-efficient fine-tuning (PEFT)* [185] methods—e.g., Prompting or Adapters– which optimize only a small subset of parameters. Unfortunately, these still require full back-propagation through the backbone, making it still memory-intensive. To address this, [138] introduced *side-tuning* (ST), an effective PEFT, and *memory-efficient fine-tuning* (MEFT) method. ST creates a parallel pathway to refine

intermediate features, thus back-propagating over a minimal set of parameters only. In this regard, we highlight the work of R^2-Tuning [187], the first ST approach for VMR, which recursively fuses multi-modal intermediate CLIP [118] embeddings. [187] also proposes adapting these frame-level embeddings for MR by generating a dense set of anchors. As shown in Fig. 4.1, this method proves ineffective for MR— a highly sparse task where videos may contain very few ground-truth action.

This motivates our dual-stream *Sparse-Dense Side-Tuner (SDST)*, the first proposal-free ST method, carefully designed for multi-task learning across sparse (MR) and dense (HD) tasks. For this, SDST jointly refines frame-level embeddings—suitable for HD– while learning what we call the *recurrent decoder queries* for MR. We also identify an implicit contextual limitation of the deformable attention [105] module of anchor-free architectures like ours, which in our task results in collapsing offsets around their initialization values. This also impedes the selection of keys outside of the current estimated moment boundaries, critical to potentially refine the boundaries of longer moments. Consequently, we propose the alternative *Reference-based Deformable Self-Attention (RDSA)*, which naturally addresses this issue by reformulating the CA into a SA-based mechanism. Finally, we tackle the performance degradation of ST methods derived from the use of image-based CLIP, over more advanced spatio-temporal VLMs [209], as noted in [181]. A key challenge of exploiting [209] is, however, defining an effective token pooling strategy, as naive strategies like CLS pooling yield significant performance drops. This motivates our proposed module re-utilization scheme, which results in the first successful integration of InternVideo2 [209] into an ST framework.

In summary, our main contributions are threefold: 1) We propose SDST, the first anchor-free ST architecture, specifically tailored for complex sparse-dense multi-task setups like VMR. Our method significantly outperforms existing ST architectures on QVHighlights [110], TACoS [28] and Charades-STA [53] (see Fig. 4.1). SDST also performs competitively and even surpasses existing SOTA while reducing its learnable parameter count by up to 73%, and incurring in a minimal memory overhead. 2) We identify a key contextual limitation of the deformable attention mechanism that, in our task, results in collapsing offsets. Consequently, we propose RDSA, which tackles this issue by allowing more complex key-selection strategies. 3) We address the limitations of transfer learning from an image-domain backbone to a video domain by integrating, for the first time, the more advanced spatio-temporal backbone [209] into an ST framework. This proves to be a non-trivial challenge with massive performance implications.

4.2 Related work

Video moment retrieval (VMR): One relevant task in Video Understanding is VMR [160] which aims to identify actions from text queries. Traditionally, this task was approached either from a MR [53, 154, 110, 163, 162, 181, 176] (as described in Chapter 3) or HD [106, 98, 29, 123, 184] perspective. MR focuses on predicting specific action proposals while HD aims to compute the saliency scores for each of the frames w.r.t. the query. Nevertheless, after the proposal of QVHighlights [110], the literature started approaching this from a multi-task learning perspective. In this regard, arguably the most notable family of methods is that based on DETR [110, 163, 162, 205, 181, 198] given the sparse nature of MR. These methods normally fuse both video and textual modality into a final embedding which is used for HD and as a memory of a Transformer decoder that refines a set of learnable queries. More in detail, Moment-DETR [110] constitutes the first baseline, which uses a standard Transformer encoder to process both modalities simultaneously. QD-DETR [163] proposes instead to inject the textual information to the video modality with cross attention (CA), followed by a temporal modeling module. CG-DETR [162] reduces the negative impact of irrelevant textual tokens by adding an Adaptive CA module that leverages dummy tokens to redirect attention weight. In this same line, SG-DETR [181] introduces a saliency-guided CA, weighting the contribution of irrelevant textual tokens based on the computed saliency scores. Despite the notable success of these proposal-free methods, one core limitation that they all share is that they rely solely on the last-layer features of a frozen CLIP [118] and/or Slowfast [88] backbones. [173] overcomes this limitation via the full fine-tuning of CLIP, even though this becomes nearly intractable given the resources that this requires. R^2-Tuning [187], the most similar method to our work, overcomes this limitation by proposing the first PEFT and MEFT method for VMR, which recursively applies a multi-modal fusion ST. In this work, we argue that one of its core drawbacks, however, is its anchor-based nature, remaining oblivious to the shown benefits of DETR-based architectures in very sparse tasks like MR [132, 129, 181]. This motivates our proposed SDST, which incorporates the best of both PEFT and MEFT.

Parameter-and-memory-efficient fine-tuning: Foundation models and VLMs [118, 140] have become the corner-stone of recent advances in Video Understanding applications. The use of pre-extracted features remains, however, an important limitation to its application on downstream tasks like VMR [160]. Fine-tuning these huge models is often simply infeasible given the large resources that these require. This has motivated the rise of PEFT [185],

which aims to democratize the fine-tuning process by restricting the tunable parameters to the bare minimum. In this regard, some of the most prominent approaches are Prompting [128, 147] and the use of Adapters [137, 180]. The first keeps the backbone frozen while learning prompts that are appended to the input, bridging the gap between the backbone's expected distribution and that of the downstream task. Adapters, in contrast, incorporate small trainable modules in the backbone while keeping the rest unchanged. These methods, however, still require full backpropagation through the frozen model, making them memory-inefficient. Recently, the new paradigm of *ST* [131, 138, 166]—and particularly [187] for VMR– has gained attraction creating a parallel pathway to exploit intermediate backbone representations while guaranteeing that backpropagation is only applied to this small parallel module. Interestingly, these methods normally rely on pre-extracted CLIP representations for tasks like HD/MR. As shown by [181], this is a very limiting aspect given the lack of temporal reasoning of CLIP or its difficulties in understanding textual queries beyond simple spatial descriptions. For this reason, in this work, we propose a novel ST architecture that, for the first time, relies on the more advanced InternVideo2 [209] backbone. Unlike CLIP, this backbone is trained on video-domain inputs, resulting in enhanced spatial and temporal modeling capabilities.

Deformable attention: A key component of most SOTA methods for VMR like SG-DETR [181] is their deformable attention module [105, 141]. This module addresses the well-studied slow convergence of DETR [94]. It does so by limiting the selectable keys based on the predictions of the learnable queries, completely decoupling both query and key spaces. Despite desirable in terms of efficiency, it has been previously noted in the literature [145] that the lack of context of the queries w.r.t. the key/value space can lead to suboptimal attendable key selection. VMR methods often mitigate this effect by partially or completely initializing the learnable queries based on the DETR-memory [145]—i.e., frame-level representations in our case. In this work, we empirically demonstrate the limitations of these initialization-based methods for VTG tasks, which motivates our proposed simple yet effective alternative, the *Reference-based Deformable Self-Attention (RDSA)*. This reformulates the deformable CA into a deformable SA, naturally solving the aforementioned issue.

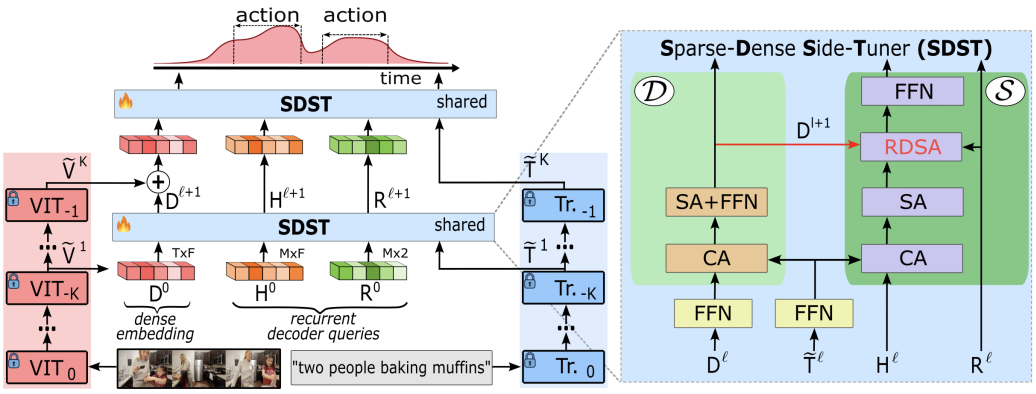


Figure 4.2: Our method (left) first processes the video and textual inputs using [209], and then recursively applies a shared SDST, a recurrent dual-stream model attached to the last K layers. This iteratively refines dense embeddings and learnable *recurrent decoder queries*. The SDST consists of a dense stream $\mathcal{D}$ for temporal and multi-modal dense refinement, and a sparse stream $\mathcal{S}$ that refines the recurrent queries conditioned on the dense signal using our proposed RDSA module.

4.3 Method

4.3.1 Problem definition

In this chapter, we address the problem of MR and HD based on textual descriptions. For this, we consider an arbitrary input-query pair $(\mathbf{X}^v, \mathbf{X}^t)$ where $\mathbf{X}^v \in \mathbb{R}^{T\times H\times W\times 3}$ and $\mathbf{X}^t \in \mathbb{R}^{L\times F_e}$. Here T and L are the number of frames and tokens, and F_e the textual encoding dimension. Our goal is to predict, for the given video-query pair, its respective frame-wise saliency scores $\mathbf{Y}^s \in \mathbb{R}^T$ for HD, and the M action moments $\mathbf{Y}^m \in \mathbb{R}^{M\times 2}$ for MR.

4.3.2 Overview

In our work we propose a dual-stream ST architecture that includes a dense (frame-level) and a sparse (segment-level) stream for MR and HD respectively (see Fig. 4.2). More specifically, we first leverage a frozen InternVideo2-1B [209] backbone, a model that possesses powerful spatio-temporal modeling capabilities, to extract K intermediate visual-textual representations. These are then processed by multiple weight-sharing SDST layers, which refine dense representations and recurrent decoder queries. More in detail, at each level, the model refines the dense embeddings via multi-modal textual conditioning as well as temporal modeling. This signal is then used to condition the sparse

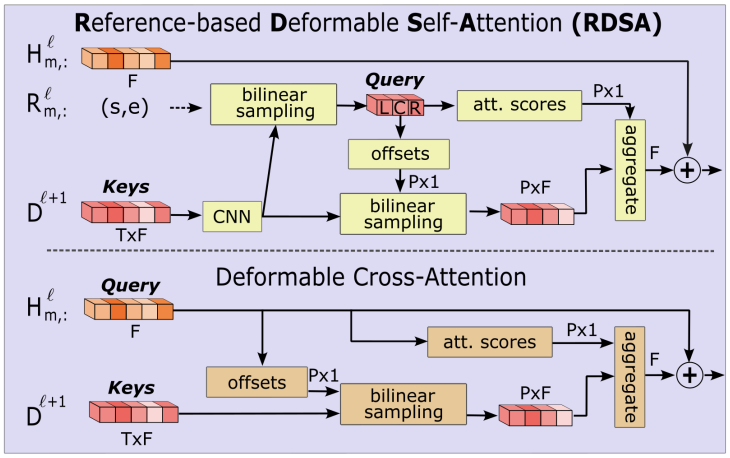


Figure 4.3: Comparison between RDSA and Def.CA[141] for a proposal m. The RDSA module is based on a context enhanced deformable attention mechanism that restricts the selectable keys based on the center, left-, and right-most action embeddings. The final dense embedding solves HD, while the recurrent queries address MR.

stream $\mathcal{S}$, that builds on [129] while incorporating our novel deformable attention mechanism, namely *Reference-based Deformable Self-Attention* (see Fig. 4.3). This mitigates the shortcomings of the deformable CA mechanism, enhancing the contextual information of the decoder queries and thus improving the quality of the selected keys. After unrolling this process K times, we apply the dense and the sparse prediction heads to compute the saliency scores and the predicted segment boundaries, respectively.

4.3.3 Sparse-Dense Side-Tuner (SDST)

4.3.3.1 Recurrent refinement with side-tuners

Our work builds on an ST framework, which first requires extracting K intermediate video and textual features $\tilde{\mathbf{V}} \in \mathbb{R}^{K \times T \times F_v}$ and $\tilde{\mathbf{T}} \in \mathbb{R}^{K \times L \times F_t}$ (find the details in Sec. 4.4). Here F_v and F_t are their respective dimensionalities. We then define a zero-initialized dense embedding $\mathbf{D}^0 \in \mathbb{R}^{T \times F}$ and a set of M learnable recurrent moment proposals inspired by [132], which we refer to as *recurrent decoder queries*. The queries include the learnable center-width moment references $\mathbf{R}^0 \in \mathbb{R}^{M \times 2}$, and their corresponding latent embeddings $\mathbf{H}^0 \in \mathbb{R}^{M \times F}$. We then define the recurrence for a given level $1 \leq \ell \leq K$ as

follows:

$$\mathbf{D}^{\ell+1}, \mathbf{R}^{\ell+1}, \mathbf{H}^{\ell+1} = SDST(\mathbf{D}^{\ell}, \mathbf{R}^{\ell}, \mathbf{H}^{\ell}, \tilde{\mathbf{V}}^{\ell}, \tilde{\mathbf{T}}^{\ell}), \tag{4.1}$$

where SDST represents the shared-weight layers of our model, progressively refining the representations across two streams: the *1) dense learning stream* $\mathcal{D}$ and the *2) sparse learning stream* $\mathcal{S}$, that we describe below.

4.3.3.2 Dense learning stream $\mathcal{D}$

Following [187], this stream first refines the dense embeddings $\mathbf{D}^{\ell}$ by conditioning it to the video-textual embeddings $\tilde{\mathbf{V}}^{\ell}$ and $\tilde{\mathbf{T}}^{\ell}$. For this, both of these multi-modal embeddings are projected to a shared F-dimensional space, using two MLPs ($\mathcal{F}_v : F_v \rightarrow F$ and $\mathcal{F}_t : F_t \rightarrow F$):

$$\mathbf{V}^{\ell} = \mathcal{F}_v(\tilde{\mathbf{V}}^{\ell}) \ , \ \mathbf{T}^{\ell} = \mathcal{F}_t(\tilde{\mathbf{T}}^{\ell}). \tag{4.2}$$

Then we incorporate the clip-wise visual information to $\mathbf{D}^{\ell}$ via a weighted sum modulated by $\beta^{\ell} \in [0, 1]$, a zero-initialized layer-dependent parameter that yields:

$$\mathbf{D}^{\ell} := \beta^{\ell}\mathbf{D}^{\ell} + (1 - \beta^{\ell})\mathbf{V}^{\ell}. \tag{4.3}$$

Thereafter, we inject the textual information with CA, where $\mathbf{T}^{\ell}$ does not include the CLS token. We also apply a temporal modeling module defined as a SA and a subsequent point-wise feed forward network (PFFN):

$$\mathbf{D}^{\ell+1} = PFFN(SA(CA(\mathbf{D}^{\ell}, \mathbf{T}^{\ell}, \mathbf{T}^{\ell}))), \tag{4.4}$$

4.3.3.3 Sparse learning stream $\mathcal{S}$

This stream can be seen as a recurrent DETR-based mechanism that exploits the complementarity of both sparse (MR) and dense (HD) tasks to refine the recurrent decoder queries. This is, refining the center-width references $\mathbf{R}^{\ell} \in \mathbb{R}^{M \times 2}$ and their latent embeddings $\mathbf{H}^{\ell} \in \mathbb{R}^{M \times F}$. Similar to the stream $\mathcal{D}$, we first incorporate a CA module that conditions $\mathbf{H}^{\ell}$ to the textual query. This is followed by a SA module to enable the information flow between different moment proposals:

$$\mathbf{H}^{\ell} = SA(CA(\mathbf{H}^{\ell}, \mathbf{T}^{\ell}, \mathbf{T}^{\ell})). \tag{4.5}$$

Another essential aspect of this stream is effectively conditioning the recurrent decoder queries to the video modality. Existing works [132, 129, 181] typically rely on the use of deformable attention due to its convergence and performance benefits [105, 141]. This mechanism leverages attention queries to predict offsets that define a limited set of selectable keys. However, in this task we observe that 1) these offsets collapse near their initialization and 2) are unable to *look* beyond the current estimated boundaries (see Sec. 4.6.2). We attribute this to contextual limitations, which motivates our proposed alternative *Reference-based Deformable Self-Attention* (RDSA), enabling an enhanced cross-modality injection. We also incorporate a simple PFFN to enhance the model's expressivity:

$$\mathbf{H}^{\ell} = PFFN(RDSA(\mathbf{R}^{\ell}, \mathbf{H}^{\ell}, \mathbf{D}^{\ell+1})). \tag{4.6}$$

Importantly, all of these modules are shared across the K levels for parameter efficiency, and include residual connections and DropPath [40] for stability and regularization.

4.3.3.4 Reference-based deformable self-attention (RDSA)

Deformable attention in the context of existing anchor-free methods: The Vanilla Attention mechanism is the core component of Transformers, one of the most popular architectures in the community at the time of this writing. The attention mechanism can be defined as:

$$\mathbf{Q} = \mathbf{X}_{\mathcal{Q}}\mathbf{W}_{\mathcal{Q}}\ ,\ \mathbf{K} = \mathbf{X}_{\mathcal{K}}\mathbf{W}_{\mathcal{K}},\ \ \mathbf{V} = \mathbf{X}_{\mathcal{V}}\mathbf{W}_{\mathcal{V}} \tag{4.7}$$

$$\mathbf{S} = \frac{\sigma(\mathbf{Q}\mathbf{K}^{T})}{\sqrt{d_k}}\mathbf{V} \tag{4.8}$$

Here σ is a softmax activation, and $\mathbf{X}_{\mathcal{Q}}, \mathbf{X}_{\mathcal{K}}$ and $\mathbf{X}_{\mathcal{V}}$ define the inputs to the query, keys and values projection matrices $\mathbf{W}_{\mathcal{Q}}, \mathbf{W}_{\mathcal{K}}, \mathbf{W}_{\mathcal{V}}$, respectively. In the self-attention case, $\mathbf{X}_{\mathcal{Q}} = \mathbf{X}_{\mathcal{K}}$ while for cross-attention, $\mathbf{X}_{\mathcal{Q}} \neq \mathbf{X}_{\mathcal{K}}$.

This mechanism, despite very extended in the community these days, presents several important pitfalls like its quadratic complexity or its slow convergance. his motivated the proposal of various efficient attention mechanisms to attain similar performance while improving its efficiency. In this chapter we focalize on the **deformable attention mechanism**, one of the keys to the success of recent anchor-free methods like DETR—including works tackling VMR [181]. This mechanism is known to provide faster convergence speed, reduced time

complexity, and as shown in Sec. 4.6.2, overall better performance than standard CA. As depicted in Fig. 4.3 (down), the key of this module is the lack of explicit interaction between queries and keys, which are defined as

$$\mathbf{Q} = \mathbf{X}_{\mathcal{Q}}\mathbf{W}_{\mathcal{Q}}^{def}\ ,\ \mathbf{K} = \mathbf{X}_{\mathcal{K}}\mathbf{W}_{\mathcal{K}}^{def}, \tag{4.9}$$

where $\mathbf{W}_{\mathcal{Q}}^{def}$, $\mathbf{W}_{\mathcal{K}}^{def}$ are two linear projections. The deformable attention thus avoids computing the $\mathbf{QK}^T$ similarity matrix, and instead employs an offset and attention-score predictors, $\mathcal{G}_{\Delta}$ and $\mathcal{G}_{\mathbf{A}}$ to select and weight—only based on the queries– a small subset P of selectable keys. The output of this mechanism, the weighted aggregation of the selected keys, namely $\mathbf{S}$, is computed as follows

$$\Delta = \mathcal{G}_{\Delta}(\mathbf{Q}) \in \mathbb{R}^{M\times P},\ \mathbf{A} = \mathcal{G}_{\mathbf{A}}(\mathbf{Q}) \in \mathbb{R}^{M\times P}, \tag{4.10}$$

$$\mathbf{S} = \sum_{p=1}^{P} (\mathbf{A}_{:,p}\ \mathbf{K}[\mathbf{c} + \mathbf{w}\odot\Delta_{:,p}]) \in \mathbb{R}^{M\times F} \tag{4.11}$$

where $\mathcal{G}_{\Delta}$, $\mathcal{G}_{\mathbf{A}}$ are often modeled as CNNs and $x[y]$ is the bilinear sampling of x on index(es) y. Notably, in our 2-dimensional setup, the offsets $\Delta_{:,p}$ are added to center reference $\mathbf{c} = \mathbf{R}_{:,0}^{\ell} \in \mathbb{R}^{M}$, and weighted over the action width $\mathbf{w} = \mathbf{R}_{:,1}^{\ell} \in \mathbb{R}^{M}$.

Limitations: Interestingly, in the literature we find that when it comes to the deformable attention, most works operate with a certain independence to whether they deal with a deformable self-attention or cross-attention, assuming that the benefits of this mechanism extrapolate to both scenarios. In this work, however, we find that in fact, the deformable attention is only naturally suited for self-attention scenarios as its effectiveness inherently relies on *implicit interactions* between $\mathbf{Q}$ and $\mathbf{K}$—not present in the cross-attention case. Concretely, in a self-attention setup, $\mathbf{X}_{\mathcal{Q}} = \mathbf{X}_{\mathcal{K}}$ (see Eq. 4.9). Thus, modeling $\mathcal{G}_{\Delta}$ and $\mathcal{G}_{\mathbf{A}}$ as CNNs naturally allows queries to gain context of the local neighborhood of queries, and consequently, of the key/value space. This breaks their independence assumption, giving the queries critical information to decide *where to look*.

Importantly, this is not the case with the deformable cross attention where we try to inject cross-modal information from the keys to the queries, which is our goal in this this work. In this scenario, $\mathbf{X}_{\mathcal{Q}} \neq \mathbf{X}_{\mathcal{K}}$, and thus, using CNN-based offset predictors does not inject knowledge of the key/value space, making its guesses *blind* or based on the *blind* predictions of previous iterations. Concretely, in Sec. 4.6.2 we show that in our task, the drawbacks of this naive

deformable cross-attention have various empirical implications such as the offset collapse near the offset initialization values, or the inability to select keys from frames beyond the estimated action boundaries, critical to refine segments into longer actions. This motivates our proposed *Reference-based Deformable Self-Attention*, an alternative mechanism that allows reformulating the deformable CA module into a deformable SA based on the references $\mathbf{R}^{\ell}$.

Our proposed alternative (RDSA): Our proposed RDSA (see Fig. 4.3) takes as input the center-width references $\mathbf{R}^{\ell} \in \mathbb{R}^{M\times 2}$ and the dense embeddings $\mathbf{D}^{\ell} \in \mathbb{R}^{T\times F}$. Then, differently from other methods that leverage the learnable queries $\mathbf{Q} = \mathbf{H}^{\ell}$ to compute the offsets and attention scores (see Eq. 4.10), we propose using an alternative query embedding. Concretely, we first refine the dense embedding $\mathbf{D}_l$ with a simple CNN, allowing frames to gain local context of their surroundings. Then, we use bilinear sampling to extract three key action embeddings: left-most (l), center (c), and right-most (r). The center provides highly informative action features, while the extremes help refine the moment boundaries. This yields,

$$\hat{\mathbf{Q}} = \hat{\mathbf{X}}_{\mathcal{Q}}\mathbf{W}_{\mathcal{Q}}^{def}\ ,\ \hat{\mathbf{X}}_{\mathcal{Q}} = CNN(\mathbf{D}^{\ell})[l,c,r]. \tag{4.12}$$

where $\hat{\mathbf{X}}_{\mathcal{Q}} \in \mathbb{R}^{M\times 3F}$. This rewrites Eq. 4.10 to

$$\hat{\Delta} = \mathcal{G}_{\Delta}(\hat{\mathbf{Q}}),\ \hat{\mathbf{A}} = \mathcal{G}_{\mathbf{A}}(\hat{\mathbf{Q}}). \tag{4.13}$$

Finally, we apply the deformable attention from Eq. 4.11 using the new offsets and attention scores $\hat{\Delta}$ and $\hat{\mathbf{A}}$. This can be viewed as *asking the key spots of the action where to look* based on the textual information that we previously injected in Eq. 4.5. This naturally solves the context limitation of the standard deformable CA given that $\hat{\mathbf{X}}_{\mathcal{Q}}$ and $\mathbf{X}_{\mathcal{K}}$ now live in the same latent space—as they are both derived from $\mathbf{D}^{\ell}$. This allows the model to make informed decisions about which keys to attend to while keeping all its core computational advantages.

4.3.4 Prediction heads and training objectives

After recursively applying the SDST over K intermediate layers, we compute the respective saliency scores and action segment proposals, as well as define their corresponding objective functions.

4.3.4.1 Highlight detection

To solve HD, we define the saliency scores as the cosine similarity between the dense embedding $\mathbf{D}^K$ and a pooled representation of $\mathbf{T}^K$. Concretely, given the dense visual embedding of the final refinement layer $\mathbf{D}^K \in \mathbb{R}^{T\times F}$, we first apply a learnable AdaptivePooling mechanism to produce a single aggregated textual embedding $\mathbf{T}^{pool} \in \mathbb{R}^F$ from the original textual representations $\mathbf{T}^K \in \mathbb{R}^{L\times F}$. We then define the per-frame saliency scores $\hat{\mathbf{Y}}^s \in \mathbb{R}^T$ as

$$\hat{\mathbf{Y}}^s = cos\_sim(\mathbf{D}^K, \mathbf{T}^{pool}) = \frac{\sum_{j=1}^{F} \mathbf{D}_j^K \mathbf{T}_j^{pool}}{||\mathbf{D}^K||\,|\mathbf{T}^{pool}||}. \tag{4.14}$$

where this cosine similarity is computed for each of the visual frames. To ensure that a higher score corresponds to a higher relevance of a given frame w.r.t. the textual embedding $\mathbf{T}^{pool}$—corresponding to the last layer K– we use a SampledNCE loss, which ranks the positive frames.

4.3.4.2 Moment Retrieval

To solve MR, following the standard DETR pipeline, we apply the Hungarian algorithm to obtain a one-to-one matching between the predicted moment boundaries $\mathbf{R}^\ell \in \mathbb{R}^{M\times 2}$ at each intermediate layer ℓ, and the ground-truth (GT) annotations $\mathbf{Y}^m \in \mathbb{R}^{M^*\times 2}$. Note that unless stated otherwise, we refer to the corresponding matches of the ground-truth $\mathbf{Y}_m$ as $\hat{\mathbf{Y}}^m \in \mathbb{R}^{M^*\times 2}$. Below we describe the different objective functions that we apply to these matching embeddings.

Classification loss: The classification loss takes the predicted action probabilities of the M different recurrent decoder queries $\hat{\mathbf{p}} \in \mathbb{R}^{M\times 1}$ and brings the probability of the unmatched proposals to 0, while the rest have a probability of 1. Given the imbalance between matched and unmatched queries, we leverage a FocalLoss

$$\mathcal{L}_{\text{cls}} = -\frac{1}{M}\sum_{m=1}^{M} \alpha(1-\hat{\mathbf{p}}_m)^\gamma \log(\hat{\mathbf{p}}_m), \tag{4.15}$$

where $\hat{\mathbf{p}}_m$ is the predicted probability for proposal m , and α and γ are the standard Focal Loss hyperparameters that help address the class imbalance.

Regression losses: We then focus our attention on the actual regression of the boundaries. For this, following previous DETR works [110] we first define

an L1 loss given by

$$\mathcal{L}_{\text{L1}} = \frac{1}{M^*} \sum_{i=1}^{M^*} \left| \hat{\mathbf{Y}}_i^m - \mathbf{Y}_i^m \right|, \tag{4.16}$$

minimizing the absolute error between the predicted and ground-truth segment boundaries. Additionally, we employ an IoU-based loss [134] to maximize the overlap between the predicted and GT action segments:

$$\mathcal{L}_{\text{IoU}} = 1 - \frac{\sum_{i=1}^{M^*} \text{IoU}(\hat{\mathbf{Y}}_i^m, \mathbf{Y}_i^m)}{M^*}, \tag{4.17}$$

where $\text{IoU}(\hat{\mathbf{Y}}_i^m, \mathbf{Y}_i^m)$ is the intersection-over-union between the predicted and ground-truth segments.

Actionness losses: As described in the Sec. 4.3.4, our NMS post-processing first considers the CLS score, which measures the probability of a predicted segment to be matched to a GT. As shown by [134], this is not enough as another pillar to an effective post-processing is having an estimate of the regression quality. For this, we also define the actionness scores $\hat{\mathbf{Y}}^a \in \mathcal{R}^M$ as the maximum overlap of every query with any of the GT. In other words, for each of the learnable recurrent query embeddings we compute the maximum IOU with any of the GT actions. Then, we apply an L1 loss to regress this score which we can then leverage during inference.

$$\mathcal{L}_{act} = \frac{1}{M} \sum_{i=1}^{M} \left| \hat{\mathbf{Y}}_i^a - max_j^{M^*}(IOU(\mathbf{R}_i^\ell, \mathbf{Y}_j^m)) \right| \tag{4.18}$$

where $\hat{\mathbf{Y}}_i^a$ is the predicted actionness score of the i-th recurrent decoder query, and $max_{j=1}^{M^*}(IOU(\mathbf{R}_i^\ell, \mathbf{Y}_j^m))$ is the maximum overlap of the i-th query w.r.t. any of the GT actions.

4.3.4.3 Multimodal alignment

One critical aspect to guarantee an effective side-tuning is the inclusion of alignment losses which bring the visual and textual latent space closer in a semantically meaningful way. This is particularly important because, although the backbone has been pre-trained to ensure some degree of alignment, adapting it to a new domain such as VMR inevitably introduces domain shifts and noise. Without this alignment losses, this would hence significantly degree

the quality of the features, and thus hinder the final performance. In our work we address this issue by introducing two contrastive losses [187] that enforce video-query consistency at different levels of intermediate representation: 1) video-level alignment 2) layer-wise alignment. Notably, these losses are applied to all the intermediate layers independently.

Video-level contrastive loss At a given level ℓ, this loss enforces similarity between action-relevant frames and their corresponding textual query embedding. Specifically, it takes the embeddings $\mathbf{V}^{\ell}$ and the pooled textual embedding $\mathbf{T}^{pool}$ of that level, and pulls the positive frames—i.e., belonging to the action– closer while pushing away the negatives. Interestingly, for a given j-th frame, we consider the negatives to be all the other j-th frames of the remaining batch elements, at the same refinement level ℓ. Thereafter, we enforce our objective via an InfoNCE loss:

$$\mathcal{L}_{video\_cal} = InfoNCE(\mathbf{V}^{\ell}, \mathbf{T}^{pool}) \tag{4.19}$$

where InfoNCE [79] maximizes the similarity between the correct video-text pairs while promoting its separation from unrelated samples.

Layer-wise contrastive loss This loss is similar to the previous one but operates across layers instead of the batch. This is, this ensures that the same frame-query pair learns different representations at two distinct levels ℓ and ℓ'. This promotes that these representations are not redundant, and thus add complementary information to the model. To be more specific, following [187] we define the negatives at level ℓ as the same frame-embedding but corresponding to a different intermediate layer ℓ'.

$$\mathcal{L}_{video\_cal} = InfoNCE(\mathbf{V}^{\ell}, \mathbf{T}^{pool}). \tag{4.20}$$

4.3.4.4 Final loss

Plugging together all the previous losses, we define the final loss as:

$$\mathcal{L} = \lambda_5 \mathcal{L}_{HD} + \lambda_6 \mathcal{L}_{MR} + \lambda_7 \mathcal{L}_{align}. \tag{4.21}$$

4.3.4.5 Inference

During inference, we apply a soft NMS post-processing to filter out redundant action predictions. This algorithm sorts the proposals based on a confidence score, which in our case we define as the square root of the product of the

class probabilities and actionness scores:

$$\hat{\mathbf{C}} = \sqrt{\hat{\mathbf{p}} \cdot \hat{\mathbf{Y}}^{a}}. \tag{4.22}$$

This prioritizes a high classification confidence together with a high localization confidence.

4.4 Extracting intermediate features with InternVideo2

The cornerstone of ST is leveraging intermediate multi-modal representations from frozen VLMs. While previous ST approaches predominantly rely on CLIP, we argue that CLIP has notable limitations in temporal modeling and aligning visual features with textual queries beyond simple static descriptions. To overcome these limitations, we incorporate InternVideo2 [209] into our framework, leveraging its advanced spatio-temporal modeling capabilities [181]. However, extracting intermediate representations from InternVideo2—particularly from its visual encoder $\mathcal{E}_v$– remains a challenging and underexplored task in the literature, despite its significant impact on performance.

In particular, the main challenge arises from how to pool the spatio-temporal token dimension L_v of $\hat{\mathbf{V}}^{\ell} \in \mathbb{R}^{T \times L_v \times F}$, the output embedding of the ℓ-th layer of $\mathcal{E}_v$, into the final intermediate features $\tilde{\mathbf{V}}^{\ell} \in \mathbb{R}^{T \times F}$ (see Eq. 4.1). Prior CLIP-based methods simply use the CLS token as "summaries" over L_v. However, we find this strategy to be suboptimal for InternVideo2 (see Sec. 4.6.1) as this limits its spatial-aggregation capabilities. Notably, [209] optimizes an AdaptivePool module that computes the final-layer embeddings, which are then aligned with the text. We conjecture that leveraging this enhanced multi-modal alignment is key to performance. Unfortunately, such pooled representation is only computed at the last layer, leaving the question of *how to pool the remaining intermediate layers.* Ideally, we would optimize layer-independent AdaptivePool modules, but this would require full back-propagation through the entire model. A more efficient alternative would be computing gradients of the AdaptivePool modules only. This is, however, still computationally infeasible, not for the gradients themselves, but because of the need to load L_v spatio-temporal tokens per frame. This increase in the input memory size translates, for instance, to a 15× memory increase for QVHighlights. We thus hypothesize that re-using this frozen

pooling module across the K intermediate layers allows a memory efficient training—no additional backpropagation nor memory requirements are needed– while leveraging its enhanced pooling capabilities, even despite their respective distribution shifts. Formally, we compute

$$\tilde{\mathbf{V}}^{\ell} = AdaptivePool\left(\hat{\mathbf{V}}^{\ell}\right) \in \mathbb{R}^{T\times F},\ 1 \leq \ell \leq K. \tag{4.23}$$

As shown in Sec. 4.6.1, this pooling strategy considerably improves other alternatives, demonstrating the benefits of this module re-utilization.

4.5 Experimentation

4.5.1 Experimental setup

We test our proposal on various datasets for MR and HD, namely QVHighlights [110], TACoS [28], and the Charades-STA [53] dataset, which we briefly describe below:

QVHighlights: QVHighlights is the only dataset among the three that provides annotations of both MR and HD tasks. Concretely, this comprises 10k YouTube videos of humanly-annotated NLP queries of a vast variety of topics, from daily activities. For convenience, these videos are trimmed to a maximum duration of 150 seconds.

TACoS: TACos is a widely used dataset for MR consisting of only 127 videos of cooking scenes with an average duration of 287 seconds. Overall, this includes 19k sentence-moment pairs. Notice that following previous works from the literature, we adapt this dataset to support our multi-task-based model by generating synthetic saliency annotations. For this, we consider a frame to have a saliency score of 1 if this belongs to the action, and 0 otherwise.

Charades-STA : Charades-STA extends the original Charades dataset, including 10k videos and 16k different sentence-moment annotations that capture a variety of indoor activities, making it a relevant benchmark to evaluate models in everyday human activity understanding.

Another important aspect of the experimental setup are the chosen evaluation metrics. For the task of MR on QVHighlights, we compute Recall@1 with two IoU thresholds, 0.5 and 0.7, and the mean average precision (mAP) at

IoU thresholds between 0.5 and 0.95 with a step of 0.05—i.e., [0.5:0.05:0.95]. For HD, we report mAP and HIT@1 over the positive frames—i.e., the most salient ones classified as "VeryGood". For TACoS and Charades-STA, we compute Recall@1 at 0.3, 0.5, and 0.7 IoU thresholds, as well as mIoU.

4.5.2 Implementation details

In this section, we describe the most relevant implementation details of our proposed SDST architecture, a summary of which is also provided in Tab. 4.1. Note that the reported hyperparameters correspond only to the best-performing model. These hyperparameters remain mostly fixed w.r.t. to existing ST works like [187] and only introduce a handful new hyperparameters which we set to 1.0 for simplicity, while we do perform grid search to optimize the learning rate.

Overall, our method is implemented using PyTorch2.0 and CUDA12.8 and runs on a single NVIDIA RTX 6000 GPU with a precision of fp16. Our models are optimized, unless stated otherwise, using AdamW with a learning rate of $1e-4$ and a weight decay of $1e-4$. This follows a step-based schedule, decaying every 20 epochs. We apply linear warmup for the first 2000 iterations with a warmup ratio of 0.001 and clip gradients to a max norm of 35.

Our model operates with a hidden dimension of 256 and leverages a sinusoidal positional encoding. The entire model relies on ReLU non-linearities to enhance the model expressivity. To improve the regularization we use a dropout of 0.5 and incorporate a droppath with a drop probability of 0.25. Our model incorporates various Transformer blocks for instance for cross-modality injection and temporal relation learning, all of which have 8-heads, an attention dropout of 0.0, and an attention output dropout of 0.0. The attention modules are initialized with Xavier. The feedforward modules of the transformer block use a hidden dimension ratio of 4 times the chosen hidden dimension and also leverage a dropout of 0.0 and a Kaiming initialization. Importantly, following standard practices, we always incorporate residual connections to improve the stability of the training and follow a PostNorm strategy [68] that normalizes the input based on a learnable LayerNorm module based on PostNorm.

Architecturally, our model consists of a dense and sparse stream, as well as their respective prediction heads. The dense stream incorporates various Transformer blocks for cross-modality injection and temporal relation learning, all of which have 8-heads, an attention dropout of 0.0, and an attention output dropout of 0.0. The attention modules are initialized with Xavier.

Pipeline component	Module	Field	Value
Architecture	General config	Dropout	0.5
		K	4
		PE	Sinusoidal
		Hidden dimension	256
		Droppath	0.25
		Non-Linearities	ReLU
		FFN ratio	4
		Attention dropout	0.0
		FFN dropout	0.0
		Attention output dropout	0.0
		FFN output dropout	0.0
		PreNorm	No
		Normalization type	LN
		Attention initialization	Xavier
		FFN initialization	Kaiming
	Sparse module	Deformable sampling points	4
	CLS head	Type	MLP
		Depth	1
	Regression head	Type	MLP
		Depth	3
	Actionness head	Type	MLP
		Depth	3
		Roi size	16
		Roi scale	0
Optimization	Optimizer		AdamW
	Learning Rate		1e-4
	Weight Decay		1e-4
	LR Schedule	Type	Step-based
		Decay rate	Every 20 epochs
	Warmup strategy	Type	Linear
		N. iterations	2000
		Ratio	0.001
	Gradient clipping	Max norm	35
Datasets	QVHighlights	Batch size	32
		FPS	0.5
		Min video len	5
		Epochs	60
		Num. queries	30
	Charades-STA	Batch size	32
		FPS	1.0
		Min video len	5
		Epochs	50
		Num. queries	30
		Learning rate	2.5e-4
		Learning rate schedule	30
	TACoS	Batch size	32
		FPS	0.5
		Min video len	5
		Epochs	150
		Num. queries	5

Table 4.1: Summary of the most relevant hyperparameters and implementation details of our model.

The feedforward modules of the transformer block use a hidden dimension ratio of 4 times the chosen hidden dimension and also leverage a dropout of 0.0 and a Kaiming initialization. Importantly, throughout the Transformer blocks, we normalize the input through a learnable LayerNorm module based on PostNorm. Additionally, one of the key components of our Sparse stream is the use of our novel deformable attention mechanism named RDSA. This first applies a context-enhancing CNN is defined as a 2-layer CNN with a hidden dimension of 256, a learnable LayerNorm, and a non-linearity. Then, after concatenating the left-most, center, and right-most tokens, it uses an MLP to project them to a 64-dimension latent space. This is then used to apply two simple linear projections to compute the 4 different sampled Keys, with their respective attention scores.

For the different prediction heads, we distinguish various different modules. On the one hand, the CLS and Regression heads are defined as a 1 and 3-layer MLPs, respectively. On the other hand, we define the actionness head following previous works like [134], which uses RoiPooling with a Roi size of 16. These roi features are then used for the actionness prediction, applying a 3-layer MLP.

Finally, we make several important training considerations. All the experiments across the different datasets use a batch size of 32 and a minimum video length of 5. FPS is set to 0.5 for QVHighlights and TACoS, and 1.0 for Charades-STA.. We train for 60 epochs on QVHighlights, 50 on Charades-STA, and 150 on TACoS. The number of queries per sample varies on the nature of the dataset. In QVHighlights and Charades-STA, for instance, we define 30 different queries, while for TACoS we use only 5. For Charades-STA, we use a slightly higher learning rate of 2.5×10^{-4} with a decay schedule of 30 epochs.

4.5.3 Main experimental results

Below we present the main results that demonstrate the improved performance of SDST with respect to various relevant baselines in several benchmarks. Notice that unless stated otherwise **bold** stands for the best and underline for the second-best.

4.5.3.1 Benchmarking with existing baselines

We begin by comparing the performance of SDST over various relevant baselines. Concretely, observe in Tab. 4.2 the evaluation of our method on the *test* and *val* splits of QVHighlights. These results indicate that our

method considerably outperforms R^2-Tuning. Similarly, SDST performs very competitively and even surpasses on several metrics the current SOTA method, namely SG-DETR [181]. Note however that our method attains this performance while having only a 27% of the parameter count of SG-DETR.

Method	test							val							#Params
	MR					HD		MR					HD		
	R1		mAP			≥ Very good		R1		mAP			≥ Very good		
	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1	
BeautyThumb	–	–	–	–	–	14.36	20.88	–	–	–	–	–	–	–	–
DVSE	–	–	–	–	–	18.75	21.79	–	–	–	–	–	–	–	–
MCN	11.41	2.71	24.94	8.22	10.67	–	–	–	–	–	–	–	–	–	–
CAL	25.49	11.54	23.40	7.65	9.89	–	–	–	–	–	–	–	–	–	–
XML+	46.69	33.46	47.89	34.67	34.90	35.38	55.06	–	–	–	–	–	–	–	–
Moment-DETR	52.89	33.02	54.82	29.40	30.73	35.69	55.60	53.94	34.84	–	–	32.20	35.65	35.65	4.8M
UMT	56.23	41.18	53.83	37.01	36.12	38.18	59.99	60.26	44.26	56.70	39.90	38.59	39.90	64.20	14.9M
MomentDiff	58.21	41.48	54.57	37.21	36.84	–	–	–	–	–	–	–	–	–	–
QD-DETR	62.40	44.98	62.52	39.88	39.86	38.94	62.40	62.68	46.66	62.23	41.82	41.22	39.13	63.03	7.6 M
MH-DETR	60.05	42.48	60.75	38.13	38.38	38.22	60.51	60.84	44.90	60.76	39.64	39.26	38.77	61.74	8.2 M
UniVTG	58.86	40.86	57.60	35.59	35.47	38.20	60.96	59.74	–	–	–	36.13	38.80	61.8	41.3M
TR-DETR	64.66	48.96	63.98	43.73	42.62	39.91	63.42	67.10	51.48	66.27	46.42	45.09	–	–	7.9 M
CG-DETR	65.43	48.38	64.51	42.77	42.86	40.33	66.21	67.35	52.06	65.57	45.73	44.93	40.80	66.70	12.0 M
BAM-DETR	62.71	48.64	64.57	46.33	45.36	–	–	65.10	51.61	65.41	48.56	47.61	–	–	–
EaTR	–	–	–	–	–	–	–	61.36	45.79	61.86	41.91	41.74	37.15	58.65	9.0 M
Mr. BLIP	74.77	60.51	68.12	53.38	–	–	–	76.13	63.35	69.39	55.78	–	–	–	19.0 M
LLaVA-MR	76.59	61.48	69.41	54.40	–	–	–	78.13	64.13	69.64	56.32	–	–	–	17.0 M
HL-CLIP	–	–	–	–	–	41.94	70.60	–	–	–	–	–	42.37	72.40	2.0 M
R^2-Tuning	68.03	49.35	69.04	47.56	46.17	40.75	64.20	68.71	52.06	–	–	47.59	40.59	64.32	2.7 M
SG-DETR†	**72.20**	**56.60**	**73.20**	**55.80**	**54.10**	**43.76**	**69.13**	–	–	**73.52**	**57.91**	**55.64**	43.91	71.47	15.0 M
Flash-VTG†	70.69	53.96	72.33	53.85	52.00	–	–	73.10	57.29	72.75	54.33	52.84	–	–	10.9 M
Ours†	70.82	56.23	71.31	54.99	53.31	43.40	**69.13**	**73.68**	**60.90**	**73.52**	57.42	55.60	**44.00**	**72.00**	4.1 M

Table 4.2: MR and HD results for QVHighlights *test* and *val* split.† indicates the use of InternVideo2 features. We mark in gray the baselines that do not support either MR or HD.

To complement our findings, in Tab. 4.3 we evaluate our method on the Charades-STA and TACoS benchmarks. Observe that our method obtains SOTA in both of these datasets. In Charades-STA, for instance, improving by 2.71% of R1@0.7 and 2.06% of mIoU, the existing SOTA. Similarly, in TACoS, SDST improves SG-DETR by 2.39% R1@0.7 and 1.27% mIoU.

4.5.3.2 Statistical analysis

In the previous section we showcase the improvement of our proposal, SDST, with respect to existing baselines on various benchmarks. In this section we assess if these gains are statistically significant. In other words, we investigate if performance of SDST significantly differs from the other relevant baselines. Importantly, for our main results we were unable to establish a fair comparison with the other baselines across various seeds, given that for

Method	Charades-ST			TACoS		
	R@0.5	R@0.7	mIoU	R@0.5	R@0.7	mIoU
M-DETR	53.6	31.4	-	24.7	12.0	25.5
UMT	48.3	29.3	-	-	-	-
UniVTG	58.0	35.7	50.1	35.0	17.4	33.6
QD-DETR	57.3	32.6	-	36.8	21.1	35.8
CG-DETR	58.4	36.3	50.1	39.6	22.2	36.5
BAM-DETR	60.0	39.4	52.3	41.5	26.8	39.3
TR-DETR	57.6	33.5	-	-	-	-
MR. BLIP	69.3	49.3	58.6	-	-	-
LLaVA-MR	70.6	49.6	59.8	-	-	-
R^2-Tuning	59.8	37.0	50.9	38.7	25.1	35.9
SG-DETR†	70.2	49.5	59.1	**44.7**	29.9	40.9
FlashVTG†	70.3	49.9	–	41.8	24.7	37.6
Ours†	**72.0**	**52.6**	**61.2**	44.5	**32.3**	**42.2**

Table 4.3: Comparison on Charades-STA and TACoS datasets. † indicates the use of InternVideo2 features.

instance QVHighlights *test* has a limited number of submissions. Consequently, we focus on the statistical study of the different model rankings across all the 3 studied datasets and their respective metrics. Concretely, we carry out two primary statistical tests: the Friedman test [2] and Nemenyi's test [5].

Friedman Test: The Friedman test is a non-parametric statistical test to test if k different variables are part of the same population. Concretely, we apply this test to study if given a set of various models, their rankings differ significantly across different datasets and metrics. We define the null hypothesis of the Friedman test as *all models perform similarly*, hence implying that there are no significant differences in the rankings across datasets/metrics. Mathematically, the Friedman statistic χ_F^2 is given by

$$\chi_F^2 = \frac{12}{N \cdot k \cdot (k+1)} \sum_{i=1}^{k} \left(R_i - \frac{N(k+1)}{2} \right)^2, \tag{4.24}$$

where N is the number of datasets and their respective metrics. K is the number of tested models, and R_i is the sum of ranks for model i across the different datasets and metrics.

In our case, the Friedman test yielded a statistic of $\chi_F^2 = 5.640$ with a p-value of 0.933. This is greater than 0.05, the threshold that is typically employed to determine the statistical significance. Hence, we can reject the null hypothesis, and conclude that there is no significant difference between the rankings of the models evaluated on all datasets. In other words, we observe that the

	QVHighlights(test)							Charades-ST			TACoS		
Method	MR					HD							
	R1		mAP			≥ Very good		R1			R1		
	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1	@0.5	@0.7	mIOU	@0.5	@0.7	mIOU
Moment-DETR	9	9	9	9	9	7	6	9	9	–	9	9	9
QD-DETR	7	7	7	7	7	5	4	8	8	–	7	7	7
UniVTG	8	8	8	8	8	6	5	7	7	6	8	8	8
CG-DETR	5	6	5	6	6	4	2	6	6	5	5	6	5
BAM-DETR	6	5	5	5	5	–	–	5	4	4	4	3	3
R2-Tuning	4	4	4	4	4	3	3	4	5	3	6	4	6
SG-DETR†	1	1	1	1	1	1	1	3	4	2	1	2	2
Flash-VTG†	3	3	2	3	3	–	–	2	2	–	3	5	4
Ours†	2	2	3	2	2	2	1	1	1	1	2	1	1

Table 4.4: Ranking position across baselines for each of the studied benchmarks. Here 1 means performing the best, and 9 the worst, respectively.

performance of the various evaluated baselines, including our SDST, perform consistently across different datasets and metrics.

Pairwise Nemenyi's Test: In this second statistical significance test we are interested in a more fine-grained analysis that might allow us to determine if our proposed method performs significantly better than the remaining considered baselines –especially of the R2-Tuning and the SG-DETR. For this, we proceed with a pairwise comparison using the Nemenyi's test. More in detail, for each pair of models, the Nemenyi test statistic is calculated based on their respective rank differences across the various datasets and metrics. We present the obtained p-values in Tab. 4.5. These results indicate that our method (SDST) performs statistically better than all the other baselines with the exception of SG-DETR which performs statistically on par. This matches our previous observations and certifies that our method attains statistically equivalent performance to SOTA while using only 27% of its respective parameter count.

These results indicate that for Ours vs SG-DETR, the p-value is 0.7926, meaning there is no statistically significant difference between the two methods. In contrast, for comparisons between Ours and the other models, the p-values are all below 0.05, suggesting that Ours is significantly better than the other models.

Comparison w.r.t. SDST	p-value	Statistically different
Moment-DETR	0.0012	✓
QD-DETR	0.0013	✓
UniVTG	0.0011	✓
CG-DETR	0.0013	✓
BAM-DETR	0.0013	✓
R2-Tuning	0.0013	✓
SG-DETR	0.7926	
Flash-VTG	0.0011	✓

Table 4.5: Nemenyi's significance test across the various pair-wise comparisons w.r.t. to our proposed SDST.

Method	MR-mAP			HD ≥ Very Good	
M-DETR	60.20 ± 0.55	34.43 ± 0.43	35.40 ± 0.41	40.31 ± 0.21	63.89 ± 0.62
UniVTG	63.51 ± 0.25	38.83 ± 0.26	37.78 ± 0.16	42.68 ± 0.09	69.34 ± 0.23
QD-DETR	67.78 ± 0.29	46.40 ± 0.26	45.52 ± 0.15	41.82 ± 0.07	68.06 ± 0.24
CG-DETR	69.86 ± 0.21	49.35 ± 0.28	48.69 ± 0.17	42.72 ± 0.07	69.87 ± 0.15
TR-DETR	70.08 ± 0.15	49.20 ± 0.50	47.99 ± 0.42	43.43 ± 0.16	71.13 ± 0.25
R2-Tuning*	71.40 ± 0.330	53.786 ± 0.684	51.49 ± 0.358	41.72 ± 0.085	69.52 ± 0.472
SG-DETR	**73.52 ±0.05**	**57.91 ±0.13**	**55.64 ±0.20**	43.91 ± 0.14	71.47 ± 0.73
Ours	73.20 ± 0.226	56.76 ± 0.53	55.31 ± 0.23	**43.93 ±0.063**	**71.62 ±0.348**

Table 4.6: Evaluation of a set of representative baselines when leveraging InternVideo2-1b features, evaluated on QVHighlights *val* split.

4.5.3.3 Experimentation using InternVideo2-1B features only

Despite the improvements shown in the previous sections, most of the presented methods in Tab. 4.2 rely on alternative feature backbones—e.g., CLIP or Slowfast–, difficulting a direct comparison with our work. To establish a more fair comparison, in Fig. 4.1 and Tab. 4.6 we follow the work of [181] and evaluate a set of relevant baselines on QVHighlights when using InternVideo2 features. Observe that our work performs on par with the SG-DETR and even more importantly to our work, the SDST very considerably outperforms R^2-Tuning, improving the MR performance by a 3.82% average mAP or the HD by a 2.21% mAP and 2.1% HIT@1.

Additionally, in Tab. 4.7 we also show the analysis of the two remaining considered datasets, these being Charades-STA and TACoS. Similarly, in these two scenarios, our method improves the second-best performing works—i.e., FlashVTG for Charades-STA and SG-DETR for TACoS— in all the metrics but R1@0.5 on TACoS where it incurs in a marginal degradation.

Method	Charades-ST			TACoS		
	R1@0.5	R1@0.7	mIoU	R1@0.5	R1@0.7	mIoU
R^2-Tuning	68.2	46.26	58.14	38.02	25.27	35.36
SG-DETR	70.2	49.5	59.1	**44.7**	<u>29.9</u>	<u>40.9</u>
FlashVTG	70.3	<u>49.9</u>	–	41.8	24.7	37.6
Ours	**72.0**	**52.6**	**61.2**	<u>44.5</u>	**32.3**	**42.2**

Table 4.7: Comparison of multiple representative baselines on Charades-STA and TACoS datasets when leveraging InternVideo2-1b features.

Method	#Params (M)	Memory (GB)	MR			HD	
			R1@0.5	R1@0.7	mAP	mAP	HIT@1
w/o Tuning	2.70	2.35	66.97	51.10	46.19	41.45	67.23
E^3VA [200]	2.57	2.96	68.97	53.16	47.68	41.04	68.13
LoSA [183]	6.40	**2.39**	72.13	58.32	53.73	41.82	68.19
LST [138]	**2.04**	2.49	70.32	55.55	50.59	41.53	69.48
R^2-Tuning[187]	2.70	2.44	70.84	55.35	51.30	41.64	69.74
Ours	4.10	3.40	**73.68**	**60.90**	**55.60**	**44.00**	**72.00**

Table 4.8: Performance and efficiency comparison of different tuning methods on QVHighlights *val* split.

4.5.3.4 Comparison with other PEFT and MEFT methods

In this section, we compare our proposed SDST against other relevant PEFT methods when leveraging InternVideo2-1B features for QVHighlights *val* split. Importantly, we note that we were unable to evaluate relevant methods based on Adapters, LORA, or Prompt-based, due to severe computational limitations. Specifically, these methods require full backpropagation through the frozen backbone, exceeding the memory capacity of our NVIDIA RTX 6000. This underscores the importance of MEFT methods like ST. Furthermore, [187] shows that these memory-expensive alternatives underperform over ST for VTG, allowing us to safely restrict the scope of this ablation to *w/o Tuning*, and to relevant ST baselines—i.e., E^3VA [200], LoSA [183], LST [138] and R^2-Tuning [187]. Among these, only R^2-Tuning is naturally suitable for a multi-modal setup like ours. Consequently, for a fair comparison, we made minimal modifications to adapt the other baselines to our setting.

Observe in Tab. 4.8 that all these baselines poses a comparable number of trainable parameters, with the exception of LoSA [183] which has a slightly higher count. Similarly, all these methods show a very efficient memory usage, which as mentioned before, contrasts with other PEFT alternatives. We find that while all these tested baselines considerably improve the *w/o Tuning*

Method	Charades-STA			TACoS		
	R1@0.5	R1@0.7	mIOU	R1@0.5	R1@0.7	mIOU
w/o Tuning	67.69	45.56	57.98	34.22	21.82	32.51
E^3VA [200]	66.13	45.11	56.23	38.77	26.02	36.15
LoSA [183]	67.69	45.16	57.42	38.54	24.49	35.71
LST [138]	68.2	46.26	58.14	38.02	25.57	35.26
R^2-Tuning[187]	69.25	46.67	58.69	39.54	27.37	36.27
Ours	**72.00**	**52.60**	**61.20**	**44.50**	**32.30**	**42.20**

Table 4.9: Performance comparison of different tuning methods for MR on Charades-STA and TACoS.

on MR, they perform quite similarly in terms of HD. Overall, our proposed SDST improves all these methods, with a especially significant boost on HD.

In Tab. 4.9 we include the homologous analysis for Charades-STA and TACoS datasets which shows similar results.

4.6 Ablation studies

In this section, we ablate over various relevant aspects of our main contributions. Note that unless states otherwise, we follow the literature and evaluate them on the *val* split of QVHighlights. Also, we write write in **bold** the best results.

4.6.1 Leveraging InternVideo2 features for ST

1) Does pooling matter? As argued in Sec. 4.4, one of the main challenges in using the InternVideo2 backbone for ST is pooling the spatio-temporal clip-wise intermediate representations. In Tab. 4.10 we compare different pooling strategies. Concretely, we compare the standard CLS-pooling, average-pooling, and our proposed re-utilization of the frozen AdaptivePool (Sec. 4.4). Observe that CLS-pooling results in a considerable performance degradation of up to 5.07% average mAP on MR or 7.74% HIT@1 on HD w.r.t. the adaptive pooling strategy. The average pooling partially mitigates this degradation, but still obtains a considerably decreased performance. This confirms the significant impact of a carefully chosen pooling strategy.

2) Feature refinement vs. feature sampling: One of the most critical decisions of ST is determining the number of intermediate levels that should be used. Normally, doing so involves evaluating the performance when refining

Pool strat.	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
CLS	66.45	52.65	67.96	51.32	50.53	41.01	64.26
Avg. pool	70.65	56.9	70.43	54.58	53.44	43.06	69.68
Adapt. pool	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.00**	**72.00**

Table 4.10: Effect of using different pooling strategies.

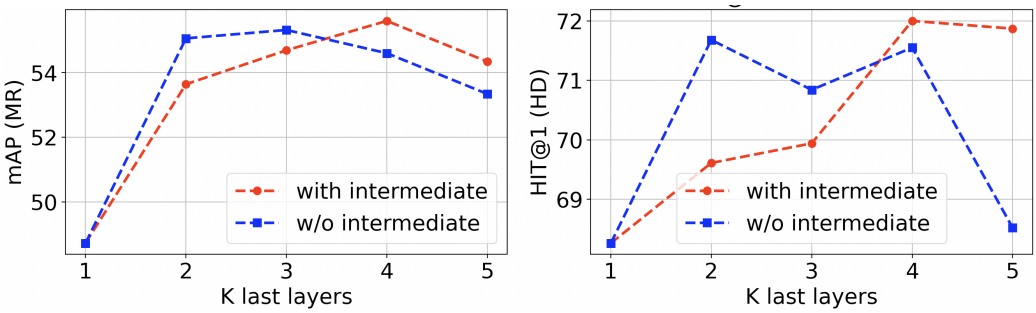


Figure 4.4: Ablation of the number of refinement levels and of their use of intermediate or last-layer features only.

the last K intermediate layers [187] . As shown in Fig. 4.4 (red) this indicates that in our case, the best performing K is 4. The literature often suggests this argument to be compelling enough to claim that using intermediate representations is beneficial over using only last-layer features. In this work, however, we find it reasonable to wonder: *Does this analysis depict the importance of intermediate features, or is it simply evidencing the need to do multiple refinement steps, regardless of the chosen features?*

To shed light on this issue, in Fig. 4.4 (blue) we evaluate the performance when doing K different refinement steps, but importantly, we always use the last-layer features only. Observe that for $K = 2$ and $K = 3$, actually, this improves the homologous experiments with intermediate features. Differently, for $K = 4$ and $K = 5$, the use of last-layer features harms the performance. These experiments thus suggest that proving the usefulness of intermediate features is not as straightforward as previously thought. In fact, these indicate that the advantages of using intermediate features arise only as we consider shallower layers.

3) Why not sampling features from even shallower layers? As shown in Tab. 4.11, this is not necessarily helpful because of what we call, *depth-pooling trade-off*. We conjecture that sampling from shallower layers does in fact provide additional complementary information. Nevertheless,

Levels	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
[37, 38, 39, 40]	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.0**	**72.00**
[25, 30, 35, 40]	71.48	59.48	71.45	56.37	54.83	43.42	70.19
[10, 20, 30, 40]	69.03	55.35	69.56	53.37	52.47	42.73	68.39

Table 4.11: Ablation of different sampling strategies of intermediate features. The first column indicates the $K = 4$ sampled layers—of a total of 40 layers.

Att. strat.	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
Stand. CA [68]	70.77	51.74	66.78	45.19	42.72	43.16	69.87
Def. CA [105]	72.58	58.19	71.82	55.80	54.27	43.26	70.58
Def. CA +PureInit [145]	72.13	57.87	70.68	54.85	52.92	43.19	70.32
Def. CA +MixedInit [145]	72.26	58.71	71.94	56.43	54.94	43.10	69.86
Ours	**73.68**	**60.90**	**73.53**	**57.42**	**55.60**	**44.0**	**72.00**

Table 4.12: Comparison across different attention strategies as well as various decoder query initializations.

this inevitably results in a distribution shift w.r.t. the last-layer feature distribution, hindering the effectiveness of the frozen AdaptivePool (see Eq. 4.23). The infeasibility of retraining this module for computational reasons, thus, leaves us with a delicate trade-off between the quality of the pooling and the depth of the features that must be accounted for.

4.6.2 Study of deformable attention

1) Comparison w.r.t. other baselines: Here we empirically evaluate the benefits of our proposed RDSA. Concretely, in Tab. 4.12 we empirically compare the performance of the standard CA [68] and Def. CA [105], as well as different decoder-query initialization strategies [145] for the latter. Notice the prominent performance degradation derived from the use of the CA, or how the query initialization techniques—far from improving the performance of the Def. CA– in some cases are even harmful. In contrast, our method consistently outperforms all these tested baselines, improving by 2.71% R1@0.7 or 1.62% mAP w.r.t. the original Def. CA.

2) Effect of the CNN and the chosen sampling points: An important aspect of RDSA are the points that are sampled to form the alternative query embeddings as well as the additional CNN module to gain local context (see Eq. 4.12). For this, in Tab. 4.13 we ablate over three possible sampling

Sampling points	CNN	MR					HD	
		R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
c		71.94	58.39	71.37	55.35	53.57	43.59	71.23
	✓	70.97	57.03	71.45	55.18	54.01	43.44	70.58
l-r		72.26	57.16	71.77	55.03	53.55	43.26	70.71
	✓	72.32	57.74	72.31	55.91	54.36	43.28	71.81
l-c-r		72.13	58.77	71.54	54.81	53.94	43.25	71.68
	✓	**73.68**	**60.90**	**73.53**	**57.42**	**55.60**	**44.00**	**72.00**

Table 4.13: Ablation of the effect of different sampling strategies like center-sampling (c), left-most and right-most action-boundary sampling (l) and (r), respectively. We also quantify the importance of our additional CNN module for enhanced context learning. Results correspond to QVHighlights *val* split.

strategies. The first samples only the center frame of the action, the second samples both the left and right-extremum of the action boundaries, and the later samples all these 3 embeddings. Note that as described in Sec. 4.3.3.4 , these embeddings are sampled based on the predicted action reference, and after concatenation, are used as alternative query embeddings for a Deformable Self-Attention mechanism.

This ablation indicates that the RDSA benefits the most from the extremum embeddings when also incorporating a CNN module. This indicates that the CNN is effectively gathering context of the neighborhood of the current action boundaries, providing critical information for the offset prediction, and thus, of *where the model should look* to further refine the predicted segments. Moreover, observe that center embeddings are necessary even though they seem to play a lesser role in the overall performance. Interestingly, the use of a CNN in fact harms the effectiveness of these embeddings. We conjecture that by definition, the center embeddings tend to be *surrounded* by very action-like embeddings. Thus, the local neighborhood does not necessarily provide useful information, and might even cause learning instabilities or aggravate the overfitting.

3) Where are the offsets pointing to? In Fig. 4.5 we depict the refinement of the weighted offset of a given query q, defined as $d_q = \sum_{p=1}^{P} A_{q,p}\Delta_{q,p}$, across the K different refinement steps. Note that as depicted in Eq. 4.11, these offsets are relative to the estimated moment widths, with -1 and +1 indicating the estimated left- and right-most boundaries. The Def. CA lacks context awareness, causing its offset predictions to stay near the offset initializations. In contrast, RDSA leverages an enhanced contextualization, thus reducing its

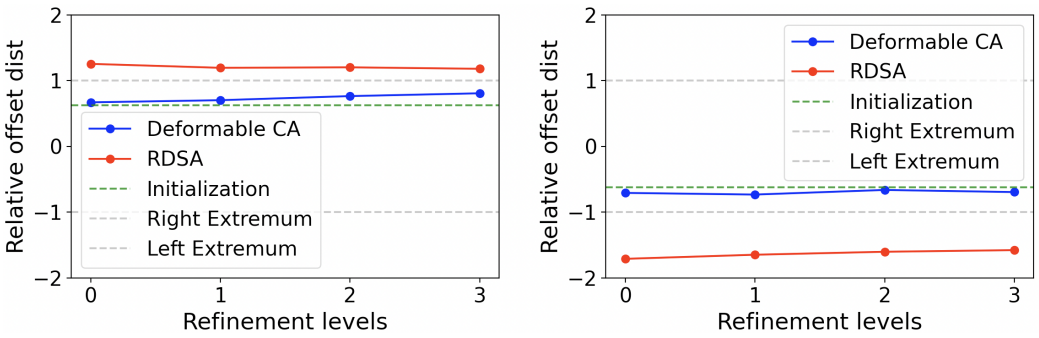


Figure 4.5: Average of the weighted offsets across M decoder queries and N batch elements for the $K = 4$ refinement levels. Here head 0 (left) is initialized near the left boundary, and head 1 (right) near the right boundary.

Att. strat.	**mAP short**	**mAP middle**	**mAP long**	**mAP**
Stand. CA	3.31	45.92	51.17	42.72
Def.CA	17.64	57.68	56.92	54.27
Ours	**18.38** (+0.74)	**58.15** (+0.47)	**59.76** (+2.84)	**55.60** (+1.33)

Table 4.14: Performance comparison of different attention strategies across different video lengths from the QVHighlights *val* split. We include the absolute difference between our method and the second-best performing baseline—i.e., Def. CA.

bias towards the offset initialization. Notably, RDSA learns to *look* beyond the current action boundaries (offsets < -1 and $> +1$) capturing near-boundary information, crucial for refining moment boundaries. This contrasts with [105] which remains constrained within the predicted moment boundaries.

4) Which actions benefit the most? In Tab. 4.14 we disentangle the MR performance according to the action length—i.e., short, middle, and long actions. This comparison indicates that one of the core limitations of the standard CA module is its almost complete inability to correctly identify short actions. Observe that its performance over short actions degrades by 14.33% and 15.07 w.r.t..the Deformable CA and RDSA, respectively. We also observe that our method is especially effective at predicting long actions, improving by 2.84% mAP w.r.t. the Deformable CA.

4.6.3 Study of the dual stream architecture

1) Effect of each module of the sparse stream $\mathcal{S}$ In this ablation, we empirically evaluate the contribution of each of the components of $\mathcal{S}$—i.e.,

CA	SA	RDSA	FFN	MR					HD	
				R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
	✓	✓	✓	71.42	57.74	71.73	56.18	54.47	43.53	71.35
✓		✓	✓	71.87	58.71	72.13	55.99	54.32	43.28	69.55
✓	✓		✓	59.68	30.71	59.02	24.6	29.09	43.58	70.39
✓	✓	✓		72.84	59.29	73.13	56.91	55.72	43.85	72.58
✓	✓	✓	✓	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.0**	**72.00**

Table 4.15: Importance of the modules of the sparse stream $\mathcal{S}$.

Type of features	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
Raw. rep	72.58	56.37	70.12	50.99	50.74	44.13	71.37
Post CA	72.26	58.39	72.20	55.92	54.83	43.64	70.65
Post CA+SA	72.13	58.32	72.00	56.57	54.94	43.78	70.06
Post CA+SA+FFN	**73.68**	**60.90**	**73.53**	**57.42**	**55.60**	**44.00**	**72.00**

Table 4.16: Evaluation of various conditioning signals.

the influence of the CA, the SA, the RDSA, and the final PFFN. Observe in Tab. 4.15 that the initial textual injection CA seems to be particularly relevant for the R1 metric, while the reasoning SA module is important for the mAP. Moreover, as expected, the RDSA module is critical. This is to be expected, as without it, the queries become oblivious to the input video, leaving no other option but that of randomly guessing. Again, the additional expressivity that the final PFFN brings via its non-linearities proves to be beneficial, despite affecting less than the other modules.

2) Conditioning signal for the sparse stream $\mathcal{S}$: To effectively integrate RDSA, it is imperative to obtain an expressive video-based conditioning signal—i.e., the red arrow in Fig. 4.2. As shown in Tab. 4.16, using the raw video representation $\mathbf{V}^\ell$ significantly degrades performance, highlighting the need for a richer signal incorporating textual and temporal information. While conditioning on the outputs of the CA or SA modules of the dense stream $\mathcal{D}$ improves results, these lack non-linearities. Instead, using the non-linear PFFN output provides greater flexibility, leading to superior expressiveness. Notably, this choice also enhances HD performance, underscoring the benefits of task interaction between MR and HD.

3) Parameter sharing in SDST Tab. 4.17 compares the performance of SDST with and without parameter sharing. This is, we evaluate if creating independent SD side-tuners for each of the $K = 4$ different intermediate layers results in an improved performance on the MR and HD tasks. Observe that

Shared	MR					HD		Params
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1	
✓	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.0**	**72.00**	4.10 M
	71.23	57.1	71.62	55.44	54.1	43.82	70.45	12.43 M

Table 4.17: Ablation of the effect of using shared vs unshared parameters on QVHighlights *val* split.

Mod. permutation	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
SA-CA-Def-FFN	70.77	56.13	71.54	54.66	53.81	43.36	70.39
SA-Def-CA-FFN	70.97	57.29	71.30	54.72	53.74	43.06	69.87
Def-SA-CA-FFN	71.81	57.61	71.61	55.05	53.91	43.16	70.19
Def-CA-SA-FFN	72.77	58.71	72.78	56.46	54.98	44.04	71.35
CA-Def-SA-FFN	72.58	59.42	72.78	57.26	54.94	43.46	70.77
CA-SA-Def-FFN	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.00**	**72.00**

Table 4.18: Ablation on the importance of the ordering of the components in the sparse stream when evaluated on QVHighlights *val* split.

this is not the case. Despite the additional 8.32M parameters, unsharing the different side-tuning modules in fact results in a performance degradation in all the tested metrics. We hypothesize that sharing the same alignment module (see Eq.2) with the subsequent $\mathcal{L}_{align}$ loss, promotes that 1) embeddings share a unique latent space while 2) different layers focus on different semantics. This allows the sharing of the remaining modules, which we observed contributes to stabilizing the optimization, and thus, improve performance.

4) Study of the ordering of the different modules of the sparse stream: One important aspect to consider is the ordering of the 4 different modules of the sparse stream. For this, in Tab. 4.18 we evaluate multiple relevant combinations. Note that we avoid ablating over the final FFN module due to computational limitations. In this regard, Tab. 4.18 indicates that it is beneficial to include the CA module to gain textual context as early as possible.

4.6.4 Studying the efficiency and optimization stability

1) Additional efficiency analysis: In this chapter, we normally study the efficiency of our model w.r.t. existing related works based only on the number of parameters. In Tab. 4.19 we extend this analysis to the training memory and the running time. For simplicity purposes, we limit this study to the QVHighlights dataset.

	# Params (M)	Memory (GB)	Runtime (it/s)
Moment-DETR	4.8	1.54	7.45
R2-Tuning	2.7	2.4	5.55
TR-DETR	7.9	1.76	4.75
HL-CLIP	2.0	22.98	0.64
Llava-MR	17.0	$\approx 80 \times 8$	-
MR.Blip	19.0	$\approx 80 \times 8$	-
SG-DETR	15.0	-	-
Flash-VTG	10.9	2.3	5.2
Ours	4.1	3.4	4.16

Table 4.19: Efficiency summary of a set of representative models evaluated on QVHighlight with InternVideo2-1b features, and a batch size of 32.

$\mathcal{L}_1$	$\mathcal{L}_{IOU}$	$\mathcal{L}_{align}$	$\mathcal{L}_{act}$	$\mathcal{L}_{cls}$	MR					HD	
					R1@0.5	R1@0.7	mAP@0.5	mAP@0.75	mAP	mAP	HIT@1
	✓	✓	✓	✓	71.68	58.58	72.28	56.28	55.0	43.52	71.35
✓		✓	✓	✓	73.94	59.55	72.07	55.21	54.25	43.35	71.26
✓	✓		✓	✓	71.55	58.13	71.55	55.21	54.25	43.68	69.48
✓	✓	✓		✓	73.1	59.61	73.87	57.01	55.6	43.29	71.00
✓	✓	✓	✓		70.52	57.23	69.88	54.82	52.95	42.28	68.65
✓	✓	✓	✓	✓	**73.68**	**60.90**	**73.52**	**57.42**	**55.60**	**44.00**	**72.00**

Table 4.20: Importance of the main losses of our model when evaluated on QVHighlight val with InternVideo2-1b features.

2) Study of the inherent optimization difficulty: Undeniably, existing SOTA models—e.g., [187, 181]– accumulate a considerable number of losses and components. And unfortunately, ours is no exception. While this represents a considerable opportunity for future studies trying to create more compact models, in this section we aim to extend the ablation from Tab. 4.15 where we justify the necessity of the various model components of the sparse stream $\mathcal{S}$. Concretely, in Tab. 4.20 we show the need for each of the different proposed losses—with the exception of those that are indispensable to solve the inherent task, like the saliency-related losses.

In terms of difficulty of optimization and parameter search, we highlight the considerable robustness in terms of hyperparameter choice. Concretely, as specified in Sec. 4.5.2, the vast majority of the hyperparameters are kept consistent with previous relevant works like [187]. The newly introduced hyperparameters were set to 1.0 for simplicity. Nevertheless, this does not guarantee the robustness to this hyperparameter choice. Consequently we propose the following experiment: We randomly sample 4 additional different

Perm. #	MR					HD	
	R1@0.5	R1@0.7	mAP@0.5	mAP@0.7	mAP	mAP	HIT@1
1	73.03	59.10	72.78	56.31	55.17	43.70	70.58
2	73.94	59.35	73.92	57.26	55.87	44.30	72.58
3	72.19	57.87	72.43	56.22	54.72	44.03	71.74
4	72.97	58.77	73.13	56.84	55.58	44.52	71.68
Chosen	73.68	60.90	73.52	57.42	55.60	44.00	72.00
Mean±Std	73.16 ± 0.61	59.19 ± 0.98	73.15 ± 0.52	56.81 ± 0.48	55.38 ± 0.40	44.11 ± 0.27	71.71 ± 0.65

Table 4.21: Performance across different permutations with main retrieval and detection metrics.

Perm #	λ_{L1}	λ_{IOU}	λ_{sal}	λ_{align_video}	λ_{align_layer}	λ_{act}	λ_{cls}
1	1.47	1.91	0.11	0.18	0.42	0.84	1.27
2	1.43	0.41	0.17	0.33	0.37	0.3	0.29
3	1.81	0.92	0.36	0.42	0.23	0.63	0.64
4	0.4	1.24	0.31	0.1	0.16	1.13	0.64
Chosen	1	1	0.1	0.1	0.1	1	1

Table 4.22: The randomly chosen 4 different loss weight configurations and the final chosen configuration.

configurations—i.e., all the loss weights– defining a range of $[0.25, 2]$ for $\lambda_{L1}, \lambda_{IOU}, \lambda_{act}$ and λ_{cls} and from $[0.1, 0.5]$ for $\lambda_{sal}, \lambda_{align\_video}$ and λ_{align_layer}. Observe in Tab. 4.21 the results of each of these configurations—defined in Tab. 4.22– and observe that our model does not deviate significantly given these new random permutations. In fact, these more robust performance metrics—not requiring cherry picking the best configuration– would still rank equally in the overall ranking from Tab. 4.2.

4.7 Conclusions and future work

In this chapter, we introduced the *Sparse-Dense Side-Tuner*, the first proposal-free side-tuning method for VMR. Our dual-stream architecture jointly optimizes dense embeddings and *recurrent decoder queries*. We also identify an inherent context limitation of the Deformable CA—a key component of proposal-free architectures like ours— which motivates our proposed alternative, the *Reference-based Deformable Self-Attention*. Finally, we provide the first effective integration of InternVideo2 to a side-tuning framework, leading to a substantial performance boost. Overall, these contributions advance the thesis goal by enabling an efficient

knowledge transfer across video understanding tasks, showing how pretrained representations can be adapted with minimal additional parameters.

Future work: While this approach improves adaptation efficiency, an important avenue for future research is evaluating and enhancing cross-domain generalization, particularly in scenarios with diverse visual and linguistic distributions. Additionally, achieving end-to-end efficiency—i.e., in terms of compute, parameters, amount of training data, etc— remains an open challenge. This avenue of research could involve exploring techniques that reduce the computational cost of large vision-language model forward passes, including sparse temporal processing, dynamic token pruning, and early-exit strategies. Such developments would further enable scalable and practical deployment of VMR systems in real-world applications.

Chapter 5

Conclusions

This thesis has addressed one of the most pressing challenges in modern deep learning for video understanding: achieving robust generalization across environments and tasks. While recent state-of-the-art models have achieved impressive performance on controlled benchmarks, their reliance on dataset-specific correlations severely limits their applicability in realistic deployment scenarios. Consequently, rather than targeting peak performance on existing benchmarks, this work emphasizes the generalization of state-of-the-art video search and retrieval models across diverse environments, as well as the transferability of their knowledge across related tasks. Together, these properties constitute first-class objectives for making video understanding systems reliable and deployable in real-world applications.

The contributions of this thesis are organized around three axes: visual out-of-distribution (OOD) generalization, linguistic OOD generalization, and efficient knowledge transfer across tasks. First, we investigated visual OOD generalization in Temporal Action Localization. Through the proposed SADA framework (Chapter 2), we demonstrated that models trained on a single visual domain often fail catastrophically when transferred to unseen environments, largely due to their reliance on spurious contextual correlations rather than invariant temporal dynamics. By introducing semantics-informed adversarial alignment, SADA promotes the learning of domain-invariant spatio-temporal representations, showing that robust generalization requires explicitly disentangling action semantics from visual context.

Second, this thesis addressed linguistic OOD generalization in Video Moment Retrieval (VMR). In Chapter 3, we identified a critical limitation in current VMR evaluation protocols: the widespread reliance on highly descriptive, visually grounded caption-based queries. We showed that such formulations poorly reflect real user behavior and induce significant linguistic bias, leading to severe performance degradation when models are exposed to ambiguous or under-specified queries. By formalizing and measuring this linguistic generalization gap, this work revealed systematic failure modes, particularly in DETR-style retrieval architectures, and provided a foundation for more realistic and informative evaluation of video search systems.

Third, we examined generalization from the perspective of efficient knowledge transfer across related but distinct tasks. We argue that such transferability is a necessary condition for truly scalable and generalizable video understanding systems. In practice, this challenge is amplified by the rise of increasingly powerful vision–language pre-trained models, whose adoption is often hindered by the high computational cost of task adaptation. In Chapter 4, we introduced SDST, a parameter-efficient fine-tuning framework for video grounding that enables effective reuse of pretrained knowledge at a fraction of the adaptation cost. This approach demonstrates that robust generalization must be accompanied by efficient mechanisms for knowledge transfer, ensuring that powerful models remain accessible and deployable.

Beyond methodological contributions, this thesis also advances evaluation practices in video understanding. For TAL, we introduced new cross-dataset evaluation protocols that move beyond in-domain testing and explicitly assess domain generalization. Moreover, for VMR, we proposed the first benchmarks designed to quantify linguistic generalization by simulating realistic user queries. Together, these benchmarking efforts enable more rigorous, transparent, and deployment-relevant evaluation of video understanding systems.

Overall, this thesis shifts the focus of video understanding research from optimizing performance on fixed benchmarks toward designing models that are robust, flexible, and scalable under realistic conditions. By addressing visual and linguistic distribution shifts alongside efficient knowledge transfer, this work contributes foundational insights and practical tools for advancing video search and retrieval systems capable of operating reliably in real-world environments.

5.1 Limitations

This section highlights several cross-cutting challenges that affect not only the individual works presented in this thesis, but also the broader progress of the field toward genuinely deployable video understanding (search and retrieval) systems.

5.1.1 Limited Cross-Dataset Evaluation

This limitation is particularly evident in TAL, as studied in Chapter 2(SADA). Existing datasets rarely share compatible action taxonomies, as annotations tend to be highly domain-specific or overly fine-grained. As a result, evaluating

true cross-dataset transfer often requires simplifying assumptions or proxy mappings, which inevitably restrict the strength of the conclusions that can be drawn about domain invariance.

A similar issue arises in VMR, as investigated in Chapters 3 and 4. Here, datasets differ substantially in video source, style, duration, annotation density, and query formulation. Performance drops observed across datasets may therefore stem not only from a lack of semantic generalization, but also from structural incompatibilities in annotation or data format. This ambiguity complicates the interpretation of empirical results and limits our ability to isolate genuine failures of representation invariance.

Overall, the absence of standardized, interchangeable benchmarks fundamentally constrains progress in generalization research. Addressing this limitation will require coordinated community efforts toward shared taxonomies, cross-dataset protocols, and evaluation settings explicitly designed to disentangle semantic generalization from dataset-specific artifacts.

5.1.2 Restricted Real-World Scope of Generalization and Invariance

Although the methods proposed in this thesis demonstrate strong performance both on existing and the proposed benchmarks, they have not yet been exhaustively validated in unconstrained real-world environments, where sources of variability extend well beyond those captured by current datasets.

In Chapter 2, SADA addresses several major forms of visual domain shift, such as changes in camera viewpoint and scene context. However, this coverage remains incomplete. Achieving true visual invariance would require robustness to a much broader spectrum of conditions, including extreme weather, severe motion blur, adversarial perturbations, sensor noise, and transitions between simulated and real-world environments. These factors remain largely underexplored in current video understanding benchmarks.

Similarly, while Chapter 3 demonstrated that a significant source of linguistic generalization failure arises from the mismatch between descriptive caption-based queries and realistic user queries, this analysis does not fully capture the complexity of human language. Important aspects such as negations, relative reasoning—e.g.,*"shorter than the previous action"*—, conversational context, implicit intent, and other linguistic resources such as sarcasm were not explicitly modeled. As a result, the linguistic robustness

of current systems remains limited when faced with the full spectrum of real-world user interactions.

These limitations highlight that benchmark-driven generalization, while necessary, remains only a partial proxy for real-world robustness. Bridging this gap will require both richer evaluation settings and models capable of reasoning under far less controlled conditions.

5.1.3 Computational Efficiency Across the Model Life Cycle

While this thesis significantly improves the efficiency of knowledge transfer through parameter-efficient fine-tuning, computational cost remains a limiting factor when considering the full model life cycle.

In Chapter 4, SDST reduces adaptation costs by minimizing the number of trainable parameters. However, inference still requires full forward passes through large video–language models, often over long video sequences. Consequently, the proposed solutions do not fully address the substantial computational burden associated with large-scale inference, which remains a critical bottleneck for deployment. Truly efficient video understanding systems must therefore propose more holistic approaches that guarantee efficiency in all the different stages of modeling, including adaptation, but also other aspects like inference-time efficiency. From my point of view, promising directions include sparse temporal sampling [139], dynamic token pruning [149], early-exit mechanisms [45, 191] as well as adaptive computation strategies that adapt to the complexity of the input [170, 179]—avoiding to assign a fixed computation budget regardless of the complexity of the input (a harder query should allocate more compute than a simpler one). Nevertheless, integrating such mechanisms still remains an open challenge today, marking a promising direction for future research.

5.2 Directions for Future Research

5.2.1 Physics-Aware Invariance and Causal Learning

Moving beyond statistical domain adaptation, a promising direction for future research lies in incorporating physics-aware and causal reasoning into video understanding models [107, 169]. Instead of relying solely on correlational patterns, I believe the scientific community should focus on devising models that explicitly distinguish causal [119, 135], invariant aspects of actions from

spurious, domain-specific cues. While this is very much an open challenge, I foresee that research in line of causal interventions on temporal sequences, counterfactual reasoning, or the integration of physical constraints related to gravity, motion, and kinematics, will play a key role. Combining video representations with physics-informed inductive biases may provide a solid path toward truly domain-agnostic generalization.

5.2.2 Scaling Transferability with Modular and Sparse Architectures

Building on the efficiency gains demonstrated by SDST, future work should explore how parameter-efficient fine-tuning can be combined with modular and sparse architectures, particularly in large Video–Language Models (VLMs). Approaches such as Mixture-of-Experts (MoE) architectures [148, 127, 197] offer the possibility of activating only task-relevant sub-modules during adaptation and inference. Such designs would not only reduce adaptation cost, but also significantly lower inference-time computation by avoiding the processing of modules that are considered unnecessary for a given downstream task. This would further enhance scalability and make high-capacity models more accessible in real-world settings.

5.2.3 Generalizing Retrieval Toward Human-Centered Interaction

Extending the work of Chapter 3, I believe that future research should focalize on moving beyond single-shot query grounding, toward more interactive and conversational retrieval paradigms, in line with a few recent works like [204, 159, 182]. This is, devising retrieval models that do not operate on a single query only. Instead, these models could rely on conversations in which ambiguity can be resolved iteratively through dialogue, clarifications, and user feedback. For instance, for a query *"find the moment when a person shoots a ball"*, a model may require clarifications on whether the player should be from the team dressed in red, the team in blue, or both of them indistinctively. Such refinement process is analogous to how existing LLMs work, where users prompt a question, and iteratively improve and guide the response to resolve unspecified details, confusions, etc. until the desirable outcome (hopefully) is reached.

Supporting such interactions would require models capable of maintaining conversational memory, resolving references, and refining predictions based on follow-up instructions. Additionally, retrieval systems could be extended

to handle more complex reasoning queries, such as procedural or step-based instructions. These capabilities would significantly broaden the applicability of video understanding systems in real-world human–AI interaction scenarios.

5.3 Ethical and Societal Implications

5.3.1 Bias Amplification, Fairness

While this thesis improves the generalization capabilities of video understanding models, this improvement also raises important ethical questions related to bias amplification and mitigation. Large foundation models are often trained on data that reflects societal imbalances, for instance, in terms of gender, race, or sexual orientation. It is thus paramount to study if stronger generalization can unintentionally propagate these biases across domains, or if in contrast, these works offer opportunities to mitigate them, yielding fairer algorithms.

In this regard, we believe that the domain-robustness techniques explored in Chapter 2—specifically through the SADA framework—offer key opportunities. By viewing demographic attributes such as race or gender as distinct domains, we can apply the principles of domain adaptation to ensure that model performance remains consistent across diverse social groups. This conceptualization treats the shift between different demographic populations as a domain shift, where the goal is to achieve invariance to sensitive attributes while maintaining high task accuracy.

Other advances proposed in this thesis, specifically in terms of robustness to under-specified queries (see Chapter 3), also pose a promising step toward effective bias mitigation. By encouraging models to perform well on under-specified queries—where protected attributes such as gender, race, or appearance are intentionally omitted—the model is forced to rely on task-relevant semantics rather than demographic cues. For example, grounding *"a person running"* instead of *"a man running"* explicitly removes sensitive attributes from the inference process. Ensuring robust performance under such under-specification allows models to operate effectively even when potentially bias-inducing information is removed.

In this sense, under-specification is not a weakness but a powerful tool for fairness-aware design, enabling models to generalize while respecting ethical constraints. This perspective, combined with the domain-robustness

strategies of Chapter 2, opens new avenues for research at the intersection of generalization and responsible AI.

5.3.2 Privacy, Surveillance, and Responsible Use

Improved robustness in TAL and VMR enhances the effectiveness of systems used in sensitive contexts such as monitoring and surveillance. While such capabilities can have positive applications—e.g., safety, anomaly detection—, they also raise serious concerns regarding privacy or their misuse. As models become more invariant to viewpoint and environment, they also become more effective in surveillance scenarios. This dual-use nature underscores the importance of responsible deployment, transparency, and appropriate regulation. Personally, I believe that advancing technical robustness must go hand in hand with societal debate and governance mechanisms to ensure that these technologies are used in ways that respect fundamental rights.

5.3.3 Social and economic opportunities

More robust and generalizable video understanding systems have the potential for substantial social and economic impact. From a social perspective, advances in video understanding—and particularly in search and retrieval—can help mitigate several important societal challenges. For example, more effective retrieval systems can significantly improve automatic content moderation by more accurately localizing inappropriate or harmful material. This capability is especially relevant for protecting vulnerable populations, such as children, from exposure to explicit or unsafe content [116].

Another key social benefit concerns accessibility. As shown in Chapter 3, ensuring reliable performance under under-specified queries enables interaction through shorter, simpler, and less linguistically demanding inputs. This is particularly beneficial for users with dyslexia, cognitive impairments, or limited motor control—such as individuals affected by neurodegenerative diseases like ALS— who may find it difficult to produce long or precise textual descriptions. In this sense, under-specification directly lowers the barrier to interaction, fostering more inclusive and equitable human–AI interfaces.

From an economic standpoint, generalizable video understanding systems—especially for video search and retrieval— offer significant potential across a wide range of industries. In surveillance and security, such systems can substantially improve productivity by shifting human operators from continuous manual monitoring toward higher-level supervisory roles. In content moderation, robust automated systems can reduce the need for

human exposure to disturbing material, mitigating well-documented mental health risks [116, 168] while improving overall efficiency.

More fundamentally, improved generalization reduces the need for frequent retraining and domain-specific customization. Models that perform reliably across environments and tasks are more cost-effective to deploy and maintain, enabling smaller companies and organizations to access advanced video understanding capabilities without prohibitive computational resources. This contributes to the democratization of AI, reducing reliance on large-scale infrastructure, lowering barriers to innovation and reducing overall CO2 emissions.

5.3.4 Social and ethical concerns

Despite these positive prospects, the adoption of such technologies also introduces important challenges and ethical concerns. One prominent issue is the preservation of individual rights. As already observed in related AI applications, advances in automated understanding and generation have facilitated the spread of misinformation and manipulated content, including highly realistic fabricated videos. Although this thesis does not directly address misinformation detection, the increased effectiveness of video understanding systems points to the urgency of developing safeguards against misuse.

Another unavoidable societal question concerns the impact of automation on the workforce. As AI systems are increasingly adopted across industries, certain roles may become partially or fully automated. While this technological shift may create in the short term new opportunities in highly skilled domains such as science, engineering, and technology, I believe that in the long term, this may result in a substantial decrease of labor demand. Thus, unlike previous industrial revolutions, we may find ourselves in the first technological disruption that significantly diminishes the overall need for human involvement in certain tasks, rather than augmenting it. Addressing this challenge will require continued societal dialogue and forward-looking policy design, for example, by exploring reduced working hours, workforce re-skilling, or new economic models that ensure the benefits of automation are shared across all layers of society.

Overall, by enabling consistent and accurate performance under diverse visual conditions, linguistic under-specification, and efficient knowledge transfer, the methods explored in this thesis contribute not only to technical progress but also to broader social, economic and environmental objectives. While

I believe that these advances hold substantial promise, they will inevitably introduce new challenges that society must address through ongoing debate, regulation, and adaptation.

Bibliography

[1] Nils Reimers, I Sentence-BERT Gurevych, et al. "Sentence embeddings using siamese BERT-networks. arXiv 2019". In: *arXiv preprint arXiv:1908.10084* 10 (1908).

[2] Milton Friedman. "The use of ranks to avoid the assumption of normality implicit in the analysis of variance". In: *Journal of the american statistical association* 32.200 (1937), pp. 675–701.

[3] John McCarthy. "A basis for a mathematical theory of computation". In: *Studies in Logic and the Foundations of Mathematics*. Vol. 26. Elsevier, 1959, pp. 33–70.

[4] David H Hubel and Torsten N Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex". In: *The Journal of physiology* 160.1 (1962), p. 106.

[5] Peter J Huber. "Fifth Berkeley Symposium on Mathematical Statistics and Probability". In: *University of California* (1967).

[6] Leslie G Ungerleider. "Two cortical visual systems". In: *Analysis of visual behavior* 549 (1982), chapter–18.

[7] Irving Biederman. "Recognition-by-components: a theory of human image understanding." In: *Psychological review* 94.2 (1987), p. 115.

[8] Roger N Shepard. "Toward a universal law of generalization for psychological science". In: *Science* 237.4820 (1987), pp. 1317–1323.

[9] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. "Learning long-term dependencies with gradient descent is difficult". In: *IEEE transactions on neural networks* 5.2 (1994), pp. 157–166.

[10] Nikos K Logothetis, Jon Pauls, and Tomaso Poggio. "Shape representation in the inferior temporal cortex of monkeys". In: *Current biology* 5.5 (1995), pp. 552–563.

[11] Yann LeCun and Yoshua Bengio. "Convolutional networks for images, speech, and time series". In: *The handbook of brain theory and neural networks* (1998).

[12] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. prentice hall professional technical reference, 2002.

[13] Yves Grandvalet and Yoshua Bengio. "Semi-supervised learning by entropy minimization". In: *Advances in neural information processing systems* 17 (2004).

[14] Per Linell. *The written language bias in linguistics: Its nature, origins and transformations*. Routledge, 2004.

[15] James J DiCarlo and David D Cox. "Untangling invariant object recognition". In: *Trends in cognitive sciences* 11.8 (2007), pp. 333–341.

[16] Joaquin Quiñonero-Candela et al. *Dataset shift in machine learning*. Mit Press, 2008.

[17] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[18] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning". In: *IEEE Transactions on knowledge and data engineering* 22.10 (2009), pp. 1345–1359.

[19] Mark Everingham et al. "The pascal visual object classes (voc) challenge". In: *International journal of computer vision* 88.2 (2010), pp. 303–338.

[20] Nancy Kanwisher. "Functional specificity in the human brain: a window into the functional architecture of the mind". In: *Proceedings of the national academy of sciences* 107.25 (2010), pp. 11163–11170.

[21] Alireza Fathi, Xiaofeng Ren, and James M Rehg. "Learning to recognize objects in egocentric activities". In: *CVPR 2011*. IEEE. 2011, pp. 3281–3288.

[22] Antonio Torralba and Alexei A Efros. "Unbiased look at dataset bias". In: *CVPR 2011*. IEEE. 2011, pp. 1521–1528.

[23] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. "How does the brain solve visual object recognition?" In: *Neuron* 73.3 (2012), pp. 415–434.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012).

[25] Tomaso Poggio. "The levels of understanding framework, revised". In: *Perception* 41.9 (2012), pp. 1017–1023.

[26] Hassan Sajjad, Patrick Pantel, and Michael Gamon. "Underspecified query refinement via natural language question generation". In: *Proceedings of COLING 2012*. 2012, pp. 2341–2356.

[27] H. Kuehne, A. B. Arslan, and T. Serre. "The Language of Actions: Recovering the Syntax and Semantics of Goal-Directed Human Activities". In: *Proceedings of Computer Vision and Pattern Recognition Conference (CVPR)*. 2014.

[28] Anna Rohrbach et al. "Coherent multi-sentence video description with variable level of detail". In: *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*. Springer. 2014, pp. 184–195.

[29] Min Sun, Ali Farhadi, and Steve Seitz. "Ranking domain-specific highlights by analyzing edited videos". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer. 2014, pp. 787–802.

[30] Daniel LK Yamins et al. "Performance-optimized hierarchical models predict neural responses in higher visual cortex". In: *Proceedings of the national academy of sciences* 111.23 (2014), pp. 8619–8624.

[31] Jason Yosinski et al. "How transferable are features in deep neural networks?" In: *Advances in neural information processing systems* 27 (2014).

[32] Fabian Caba Heilbron et al. "Activitynet: A large-scale video benchmark for human activity understanding". In: *Proceedings of the ieee conference on computer vision and pattern recognition*. 2015, pp. 961–970.

[33] Jeffrey Donahue et al. "Long-term recurrent convolutional networks for visual recognition and description". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 2625–2634.

[34] Yaroslav Ganin and Victor Lempitsky. "Unsupervised domain adaptation by backpropagation". In: *International conference on machine learning*. PMLR. 2015, pp. 1180–1189.

[35] Andrej Karpathy and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3128–3137.

[36] Mingsheng Long et al. "Learning transferable features with deep adaptation networks". In: *International conference on machine learning*. PMLR. 2015, pp. 97–105.

[37] Subhashini Venugopalan et al. “Translating videos to natural language using deep recurrent neural networks”. In: *Proceedings of the 2015 conference of the north american chapter of the association for computational linguistics: human language technologies*. 2015, pp. 1494–1504.

[38] Jean-Baptiste Alayrac et al. “Unsupervised learning from narrated instruction videos”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4575–4583.

[39] Yaroslav Ganin et al. “Domain-adversarial training of neural networks”. In: *The journal of machine learning research* 17.1 (2016), pp. 2096–2030.

[40] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. “Fractalnet: Ultra-deep neural networks without residuals”. In: *arXiv preprint arXiv:1605.07648* (2016).

[41] Sergey Levine et al. “End-to-end training of deep visuomotor policies”. In: *Journal of Machine Learning Research* 17.39 (2016), pp. 1–40.

[42] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “" Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

[43] Zheng Shou, Dongang Wang, and Shih-Fu Chang. “Temporal action localization in untrimmed videos via multi-stage cnns”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1049–1058.

[44] Gunnar A Sigurdsson et al. “Hollywood in homes: Crowdsourcing data collection for activity understanding”. In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer. 2016, pp. 510–526.

[45] Surat Teerapittayanon, Bradley McDanel, and Hsiang-Tsung Kung. “Branchynet: Fast inference via early exiting from deep neural networks”. In: *2016 23rd international conference on pattern recognition (ICPR)*. IEEE. 2016, pp. 2464–2469.

[46] Daniel LK Yamins and James J DiCarlo. “Using goal-driven deep learning models to understand sensory cortex”. In: *Nature neuroscience* 19.3 (2016), pp. 356–365.

[47] Lisa Anne Hendricks et al. “Localizing moments in video with natural language”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5803–5812.

[48] Navaneeth Bodla et al. “Soft-NMS–improving object detection with one line of code”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5561–5569.

[49] Shyamal Buch et al. “Sst: Single-stream temporal action proposals”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 2911–2920.

[50] 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*. 2017, pp. 6299–6308.

[51] Gabriela Csurka. “A comprehensive survey on domain adaptation for visual applications”. In: *Domain adaptation in computer vision applications* (2017), pp. 1–35.

[52] Gabriela Csurka. “Domain adaptation for visual applications: A comprehensive survey”. In: *arXiv preprint arXiv:1702.05374* (2017).

[53] Jiyang Gao et al. “Tall: Temporal activity localization via language query”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5267–5275.

[54] Philip Haeusser et al. “Associative domain adaptation”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2765–2773.

[55] Haroon Idrees et al. “The thumos challenge on action recognition for videos “in the wild””. In: *Computer Vision and Image Understanding* 155 (2017), pp. 1–23.

[56] Will Kay et al. “The kinetics human action video dataset”. In: *arXiv preprint arXiv:1705.06950* (2017).

[57] Piotr Koniusz, Yusuf Tas, and Fatih Porikli. “Domain adaptation by mixture of alignments of second-or higher-order scatter tensors”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4478–4487.

[58] Ranjay Krishna et al. “Dense-captioning events in videos”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 706–715.

[59] Colin Lea et al. “Temporal convolutional networks for action segmentation and detection”. In: *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 156–165.

[60] Tianwei Lin, Xu Zhao, and Zheng Shou. “Single shot temporal action detection”. In: *Proceedings of the 25th ACM international conference on Multimedia*. 2017, pp. 988–996.

[61] Tsung-Yi Lin et al. “Focal loss for dense object detection”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[62] Ilya Loshchilov and Frank Hutter. “Decoupled weight decay regularization”. In: *arXiv preprint arXiv:1711.05101* (2017).

[63] Saeid Motiian et al. “Unified deep supervised domain adaptation and generalization”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5715–5725.

[64] John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. “Deeper attention to abusive user content moderation”. In: *Proceedings of the 2017 conference on empirical methods in natural language processing*. 2017, pp. 1125–1135.

[65] Zheng Shou et al. “Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5734–5743.

[66] Eric Tzeng et al. “Adversarial discriminative domain adaptation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7167–7176.

[67] Gül Varol, Ivan Laptev, and Cordelia Schmid. “Long-term temporal convolutions for action recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.6 (2017), pp. 1510–1517.

[68] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[69] Zehuan Yuan et al. “Temporal action localization by structured maximal sums”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3684–3692.

[70] Yue Zhao et al. “Temporal action detection with structured segment networks”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2914–2923.

[71] Oleksandr Bogdan et al. “DeepCalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras”. In: *Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*. 2018, pp. 1–10.

[72] Yu-Wei Chao et al. “Rethinking the faster r-cnn architecture for temporal action localization”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 1130–1139.

[73] Jingyuan Chen et al. “Temporally grounding natural sentence in video”. In: *Proceedings of the 2018 conference on empirical methods in natural language processing*. 2018, pp. 162–171.

[74] Yuhua Chen et al. “Domain adaptive faster r-cnn for object detection in the wild”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 3339–3348.

[75] Dima Damen et al. “Scaling Egocentric Vision: The EPIC-KITCHENS Dataset”. In: *European Conference on Computer Vision (ECCV)*. 2018.

[76] Robert Geirhos et al. “Generalisation in humans and deep neural networks”. In: *Advances in neural information processing systems* 31 (2018).

[77] Arshad Jamal et al. “Deep Domain Adaptation in Action Space.” In: *BMVC*. Vol. 2. 3. 2018, p. 5.

[78] Tianwei Lin et al. “Bsn: Boundary sensitive network for temporal action proposal generation”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 3–19.

[79] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation learning with contrastive predictive coding”. In: *arXiv preprint arXiv:1807.03748* (2018).

[80] Jian Shen et al. “Wasserstein distance guided representation learning for domain adaptation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[81] Gunnar A Sigurdsson et al. “Charades-ego: A large-scale dataset of paired third and first person videos”. In: *arXiv preprint arXiv:1804.09626* (2018).

[82] Waqas Sultani, Chen Chen, and Mubarak Shah. “Real-world anomaly detection in surveillance videos”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 6479–6488.

[83] Shaoan Xie et al. “Learning semantic representations for unsupervised domain adaptation”. In: *International conference on machine learning*. PMLR. 2018, pp. 5423–5432.

[84] Luowei Zhou, Chenliang Xu, and Jason Corso. “Towards automatic learning of procedures from web instructional videos”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1. 2018.

[85] Martin Arjovsky et al. “Invariant risk minimization”. In: *arXiv preprint arXiv:1907.02893* (2019).

[86] Min-Hung Chen et al. “Temporal attentive alignment for large-scale video domain adaptation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 6321–6330.

[87] Andre Esteva et al. “A guide to deep learning in healthcare”. In: *Nature medicine* 25.1 (2019), pp. 24–29.

[88] Christoph Feichtenhofer et al. “Slowfast networks for video recognition”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6202–6211.

[89] Ji Lin, Chuang Gan, and Song Han. “Tsm: Temporal shift module for efficient video understanding”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 7083–7093.

[90] Zhi Tian et al. “Fcos: Fully convolutional one-stage object detection”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 9627–9636.

[91] Huijuan Xu et al. “Multilevel language and vision integration for text-to-clip retrieval”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 9062–9069.

[92] Hang Zhao et al. “Hacs: Human action clips and segments dataset for recognition and temporal localization”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 8668–8678.

[93] Nakul Agarwal et al. “Unsupervised domain adaptation for spatio-temporal action localization”. In: *arXiv preprint arXiv:2010.09211* (2020).

[94] Nicolas Carion et al. “End-to-end object detection with transformers”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer. 2020, pp. 213–229.

[95] Min-Hung Chen et al. “Action segmentation with joint self-supervised temporal domain adaptation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 9454–9463.

[96] Shuhao Cui et al. “Gradually vanishing bridge for adversarial domain adaptation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 12455–12464.

[97] Ishaan Gulrajani and David Lopez-Paz. “In search of lost domain generalization”. In: *arXiv preprint arXiv:2007.01434* (2020).

[98] Fa-Ting Hong et al. “Mini-net: Multiple instance ranking network for video highlight detection”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*. Springer. 2020, pp. 345–360.

[99] Paul Pu Liang et al. “Towards debiasing sentence representations”. In: *arXiv preprint arXiv:2007.08100* (2020).

[100] Jonathan Munro and Dima Damen. “Multi-modal domain adaptation for fine-grained action recognition”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 122–132.

[101] Mayu Otani et al. “Uncovering hidden challenges in query-based video moment retrieval”. In: *arXiv preprint arXiv:2009.00325* (2020).

[102] Boxiao Pan et al. “Adversarial cross-domain action recognition with co-attention”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 07. 2020, pp. 11815–11822.

[103] Sinno Jialin Pan. “Transfer learning”. In: *Learning* 21 (2020), pp. 1–2.

[104] Peisen Zhao et al. “Bottom-up temporal action localization with mutual regularization”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*. Springer. 2020, pp. 539–555.

[105] Xizhou Zhu et al. “Deformable detr: Deformable transformers for end-to-end object detection”. In: *arXiv preprint arXiv:2010.04159* (2020).

[106] Taivanbat Badamdorj et al. “Joint visual and audio learning for video highlight detection”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 8127–8137.

[107] Daniel M Bear et al. "Physion: Evaluating physical prediction from vision in humans and machines". In: *arXiv preprint arXiv:2106.08261* (2021).

[108] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. "Is space-time attention all you need for video understanding?" In: *Icml*. Vol. 2. 3. 2021, p. 4.

[109] Zhiqiang Gao et al. "Gradient distribution alignment certificates better adversarial domain adaptation". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 8937–8946.

[110] Jie Lei, Tamara L Berg, and Mohit Bansal. "Detecting moments and highlights in videos via natural language queries". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 11846–11858.

[111] Zhihui Li and Lina Yao. "Three birds with one stone: Multi-task temporal action detection via recycling temporal annotations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 4751–4760.

[112] Chuming Lin et al. "Learning salient boundary feature for anchor-free temporal action localization". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 3320–3329.

[113] Jiashuo Liu et al. "Towards out-of-distribution generalization: A survey". In: *arXiv preprint arXiv:2108.13624* (2021).

[114] Xiaolong Liu et al. "End-to-end temporal action detection with transformer". In: *arXiv preprint arXiv:2106.10271* (2021).

[115] Depu Meng et al. "Conditional detr for fast training convergence". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 3651–3660.

[116] Leilasadat Mirghaderi. *Behind the screen: content moderation in the shadows of social media: by Sarah T. Roberts, New Haven and London, Yale University Press, 2019, 280 pp.,£ 14.00 (hardback), ISBN 978-0300235883*. 2021.

[117] Zhiwu Qing et al. "Temporal context aggregation network for temporal action proposal refinement". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 485–494.

[118] Alec Radford et al. "Learning transferable visual models from natural language supervision". In: *International conference on machine learning*. PMLR. 2021, pp. 8748–8763.

[119] Bernhard Schölkopf et al. "Toward causal representation learning". In: *Proceedings of the IEEE* 109.5 (2021), pp. 612–634.

[120] Deepak Sridhar et al. "Class semantics-based attention for action detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13739–13748.

[121] Jing Tan et al. "Relaxed transformer decoders for direct action proposal generation". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 13526–13535.

[122] Yingming Wang et al. "Anchor DETR: Query design for transformer-based object detection". In: *arXiv preprint arXiv:2109.07107* (2021).

[123] Minghao Xu et al. "Cross-category video highlight detection via set-based learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 7970–7979.

[124] Yuecong Xu et al. "Partial video domain adaptation with partial adversarial temporal attentive network". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 9332–9341.

[125] Xun Yang et al. "Deconfounded video moment retrieval with causal intervention". In: *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2021, pp. 1–10.

[126] Dima Damen et al. "Rescaling Egocentric Vision: Collection, Pipeline and Challenges for EPIC-KITCHENS-100". In: *International Journal of Computer Vision (IJCV)* 130 (2022), pp. 33–55. URL: https://doi.org/10.1007/s11263-021-01531-2.

[127] William Fedus, Barret Zoph, and Noam Shazeer. "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity". In: *Journal of Machine Learning Research* 23.120 (2022), pp. 1–39.

[128] Menglin Jia et al. "Visual prompt tuning". In: *European Conference on Computer Vision*. Springer. 2022, pp. 709–727.

[129] Feng Li et al. "Dn-detr: Accelerate detr training by introducing query denoising". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 13619–13627.

[130] Junnan Li et al. “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation”. In: *International conference on machine learning*. PMLR. 2022, pp. 12888–12900.

[131] Ziyi Lin et al. “Frozen clip models are efficient video learners”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 388–404.

[132] Shilong Liu et al. “Dab-detr: Dynamic anchor boxes are better queries for detr”. In: *arXiv preprint arXiv:2201.12329* (2022).

[133] Xiaolong Liu, Song Bai, and Xiang Bai. “An empirical study of end-to-end temporal action detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20010–20019.

[134] Xiaolong Liu et al. “End-to-end temporal action detection with transformer”. In: *IEEE Transactions on Image Processing* 31 (2022), pp. 5427–5441.

[135] Yang Liu et al. “Causal reasoning meets visual representation learning: A prospective study”. In: *Machine Intelligence Research* 19.6 (2022), pp. 485–511.

[136] Ye Liu et al. “Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 3042–3051.

[137] Junting Pan et al. “St-adapter: Parameter-efficient image-to-video transfer learning”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 26462–26477.

[138] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. “Lst: Ladder side-tuning for parameter and memory efficient transfer learning”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 12991–13005.

[139] Junke Wang et al. “Efficient video transformers with spatial-temporal token selection”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 69–86.

[140] Yi Wang et al. “Internvideo: General video foundation models via generative and discriminative learning”. In: *arXiv preprint arXiv:2212.03191* (2022).

[141] Zhuofan Xia et al. “Vision transformer with deformable attention”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 4794–4803.

[142] Yuecong Xu et al. “Aligning correlation information for domain adaptation in action recognition”. In: *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[143] Sunjae Yoon et al. “Selective query-guided debiasing for video corpus moment retrieval”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 185–200.

[144] Chen-Lin Zhang, Jianxin Wu, and Yin Li. “Actionformer: Localizing moments of actions with transformers”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 492–510.

[145] Hao Zhang et al. “Dino: Detr with improved denoising anchor boxes for end-to-end object detection”. In: *arXiv preprint arXiv:2203.03605* (2022).

[146] Yunhua Zhang et al. “Audio-adaptive activity recognition across video domains”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13791–13800.

[147] Kaiyang Zhou et al. “Conditional prompt learning for vision-language models”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 16816–16825.

[148] Barret Zoph et al. “St-moe: Designing stable and transferable sparse expert models”. In: *arXiv preprint arXiv:2202.08906* (2022).

[149] Shuning Chang et al. “Making vision transformers efficient from a token sparsification view”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 6195–6205.

[150] Qiang Chen et al. “Group detr: Fast detr training with group-wise one-to-many assignment”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2023, pp. 6633–6642.

[151] Mahta HassanPour Zonoozi and Vahid Seydi. “A survey on adversarial domain adaptation”. In: *Neural Processing Letters* 55.3 (2023), pp. 2429–2469.

[152] Zhengdong Hu et al. “DAC-DETR: Divide the attention layers and conquer”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 75189–75200.

[153] Zenan Huang et al. “Discriminative radial domain adaptation”. In: *IEEE Transactions on Image Processing* 32 (2023), pp. 1419–1431.

[154] Jinhyun Jang et al. “Knowing where to focus: Event-aware transformer for video grounding”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 13846–13856.

[155] Ding Jia et al. “Detrs with hybrid matching”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 19702–19712.

[156] Jihwan Kim, Miso Lee, and Jae-Pil Heo. “Self-feedback detr for temporal action detection”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 10286–10296.

[157] Pandeng Li et al. “Momentdiff: Generative video moment retrieval from random to real”. In: *Advances in neural information processing systems* 36 (2023), pp. 65948–65966.

[158] Yundong Li, Longxia Guo, and Yizheng Ge. “Pseudo Labels for Unsupervised Domain Adaptation: A Review”. In: *Electronics* 12.15 (2023), p. 3325.

[159] Kaiqu Liang and Samuel Albanie. “Simple baselines for interactive video retrieval with questions and answers”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 11091–11101.

[160] Kevin Qinghong Lin et al. “Univtg: Towards unified video-language temporal grounding”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2794–2804.

[161] Yifan Lu et al. “Exploiting Instance-based Mixed Sampling via Auxiliary Source Domain Supervision for Domain-adaptive Action Detection”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 4145–4156.

[162] WonJun Moon et al. “Correlation-guided query-dependency calibration in video representation learning for temporal grounding”. In: *CoRR* (2023).

[163] WonJun Moon et al. “Query-dependent video representation for moment retrieval and highlight detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 23023–23033.

[164] Poojan Oza et al. “Unsupervised domain adaptation of object detectors: A survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

[165] Sandro Pezzelle. “Dealing with semantic underspecification in multimodal NLP”. In: *arXiv preprint arXiv:2306.05240* (2023).

[166] Zhiwu Qing et al. “Disentangling spatial and temporal learning for efficient image-to-video transfer learning”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 13934–13944.

[167] Dingfeng Shi et al. “Tridet: Temporal action detection with relative boundary modeling”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 18857–18866.

[168] Ruth Spence et al. “The psychological impacts of content moderation on content moderators: A qualitative study”. In: *Cyberpsychology: Journal of Psychosocial Research on Cyberspace* 17.4 (2023).

[169] Hsiao-Yu Tung et al. “Physion++: Evaluating physical scene understanding that requires online inference of different physical properties”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 67048–67068.

[170] Burak Uzkent et al. “Dynamic inference with grounding based vision and language models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 2624–2633.

[171] Binglu Wang et al. “Temporal action localization in the deep learning era: A survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.4 (2023), pp. 2171–2190.

[172] Pengfei Wei et al. “Unsupervised Video Domain Adaptation for Action Recognition: A Disentanglement Perspective”. In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023.

[173] Shen Yan et al. “Unloc: A unified framework for video localization tasks”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 13623–13633.

[174] Benjin Zhu et al. “Conquer: Query contrast voxel-detr for 3d object detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 9296–9305.

[175] Peijun Bao et al. “Vid-Morp: Video Moment Retrieval Pretraining from Unlabeled Videos in the Wild”. In: *arXiv preprint arXiv:2412.00811* (2024).

[176] Zhuo Cao et al. “FlashVTG: Feature Layering and Adaptive Score Handling Network for Video Temporal Grounding”. In: *arXiv preprint arXiv:2412.13441* (2024).

[177] Xiaoxue Cheng et al. “Small agent can also rock! empowering small language models as hallucination detector”. In: *arXiv preprint arXiv:2406.11277* (2024).

[178] Rongyao Fang et al. “Feataug-detr: Enriching one-to-many matching for detrs with feature augmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.9 (2024), pp. 6402–6415.

[179] Zhanzhou Feng et al. “Efficient video transformers via spatial-temporal token merging for action recognition”. In: *ACM Transactions on Multimedia Computing, Communications and Applications* 20.4 (2024), pp. 1–21.

[180] Peng Gao et al. “Clip-adapter: Better vision-language models with feature adapters”. In: *International Journal of Computer Vision* 132.2 (2024), pp. 581–595.

[181] Aleksandr Gordeev et al. “Saliency-guided detr for moment retrieval and highlight detection”. In: *arXiv preprint arXiv:2410.01615* (2024).

[182] Guihe Gu et al. “Talksee: interactive video retrieval engine using large language model”. In: *International Conference on Multimedia Modeling*. Springer. 2024, pp. 387–393.

[183] Akshita Gupta et al. “LoSA: long-short-range adapter for scaling end-to-end temporal action localization”. In: *arXiv preprint arXiv:2404.01282* (2024).

[184] Donghoon Han et al. “Unleash the Potential of CLIP for Video Highlight Detection”. In: *arXiv preprint arXiv:2404.01745* (2024).

[185] Zeyu Han et al. “Parameter-efficient fine-tuning for large models: A comprehensive survey”. In: *arXiv preprint arXiv:2403.14608* (2024).

[186] Viacheslav Komisarenko and Meelis Kull. “Improving calibration by relating focal loss, temperature scaling, and properness”. In: *arXiv preprint arXiv:2408.11598* (2024).

[187] Ye Liu et al. “R^2-Tuning: Efficient Image-to-Video Transfer Learning for Video Temporal Grounding”. In: *arXiv preprint arXiv:2404.00801* (2024).

[188] Weiheng Lu et al. “LLaVA-MR: Large Language-and-Vision Assistant for Video Moment Retrieval”. In: *arXiv preprint arXiv:2411.14505* (2024).

[189] Boris Meinardus et al. “The surprising effectiveness of multimodal large language models for video moment retrieval”. In: *arXiv preprint arXiv:2406.18113* (2024).

[190] Yunus Emre Özköse and Pınar Duygulu. "Automatic Data Augmentation for Cooking Videos". In: *2024 32nd Signal Processing and Communications Applications Conference (SIU)*. IEEE. 2024, pp. 1–4.

[191] Haseena Rahmath P et al. "Early-exit deep neural network-a comprehensive survey". In: *ACM Computing Surveys* 57.3 (2024), pp. 1–37.

[192] Kate Sanders et al. "Grounding partially-defined events in multimodal data". In: *arXiv preprint arXiv:2410.05267* (2024).

[193] Hao Sun et al. "Tr-detr: Task-reciprocal transformer for joint moment retrieval and highlight detection". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. 5. 2024, pp. 4998–5007.

[194] Gemma Team et al. "Gemma: Open models based on gemini research and technology". In: *arXiv preprint arXiv:2403.08295* (2024).

[195] Yi Wang et al. "Internvideo2: Scaling foundation models for multimodal video understanding". In: *European Conference on Computer Vision*. Springer. 2024, pp. 396–416.

[196] Pengfei Wei et al. "Unsupervised Video Domain Adaptation for Action Recognition: A Disentanglement Perspective". In: *Advances in Neural Information Processing Systems* 36 (2024).

[197] Zhiyu Wu et al. "Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding". In: *arXiv preprint arXiv:2412.10302* (2024).

[198] Yicheng Xiao et al. "Bridging the gap: A unified video comprehension framework for moment retrieval and highlight detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 18709–18719.

[199] Lizhen Xu et al. "Redundant Queries in DETR-Based 3D Detection Methods: Unnecessary and Prunable". In: *arXiv preprint arXiv:2412.02054* (2024).

[200] Dongshuo Yin et al. "Parameter-efficient is not sufficient: Exploring parameter, memory, and time efficient adapter tuning for dense predictions". In: *Proceedings of the 32nd ACM International Conference on Multimedia*. 2024, pp. 1398–1406.

[201] Chuyang Zhao et al. "Ms-detr: Efficient detr training with mixed supervision". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2024, pp. 17027–17036.

[202] Zhuo Cao et al. “When One Moment Isn’t Enough: Multi-Moment Retrieval with Cross-Moment Interactions”. In: *arXiv preprint arXiv:2510.17218* (2025).

[203] Kevin Flanagan, Dima Damen, and Michael Wray. “Moment of Untruth: Dealing with Negative Queries in Video Moment Retrieval”. In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2025, pp. 5336–5345.

[204] Guihe Gu et al. “InstructSee: Instruction-Aware and Feedback-Driven Multimodal Retrieval with Dynamic Query Generation”. In: *Sensors* 25.16 (2025), p. 5195.

[205] Pilhyeon Lee and Hyeran Byun. “Bam-detr: Boundary-aligned moment detection transformer for temporal sentence grounding in videos”. In: *European Conference on Computer Vision*. Springer. 2025, pp. 220–238.

[206] Toby Perrett et al. “Hd-epic: A highly-detailed egocentric video dataset”. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025, pp. 23901–23913.

[207] David Pujol-Perich, Albert Clapés, and Sergio Escalera. “SADA: Semantic adversarial unsupervised domain adaptation for Temporal Action Localization”. In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2025, pp. 9237–9247.

[208] Jiajin Tang et al. “Sim-DETR: Unlock DETR for Temporal Sentence Grounding”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2025, pp. 22760–22771.

[209] Yi Wang et al. “Internvideo2: Scaling foundation models for multimodal video understanding”. In: *European Conference on Computer Vision*. Springer. 2025, pp. 396–416.

[210] Yifang Xu et al. “Zero-shot video moment retrieval via off-the-shelf multimodal large language models”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 39. 9. 2025, pp. 8978–8986.

[211] Pengcheng Zhao et al. “Ld-detr: Loop decoder detection transformer for video moment retrieval and highlight detection”. In: *arXiv preprint arXiv:2501.10787* (2025).