Egocentric Vision

Mustafa Kerem Köse
Politecnico di Torino
s339018@studenti.polito.it

Andrea Cauda
Politecnico di Torino
s343386@studenti.polito.it

Davide Tonetti
Politecnico di Torino
s334297@studenti.polito.it

Amirreza Karimipourkarian
Politecnico di Torino
s337174@studenti.polito.it

***Abstract*—In this project, enhancements to a baseline model for the Ego4D Natural Language Queries (NLQ) benchmark are presented, aimed at improving temporal localization performance within egocentric videos. Different video feature encoders, specifically *Omnivore* and *EgoVLP*, are systematically evaluated to analyze their impact on model accuracy. Moreover, the efficacy of different language embeddings (BERT and GloVe) is compared to determine their suitability for temporal localization tasks.**

To extend the analysis, 50 queries with correctly retrieved segments are manually annotated to define textual ground truths. The corresponding clips are extracted via *ffmpeg* and processed with the *Video-LLaVA* model to generate textual responses, which are then evaluated using different metrics. The complete source code for this project is available on GitHub: https://github.com/gitandrehub/Egocentric-vision.

I. Introduction

Visual temporal localization is the task of identifying precise time intervals in video streams that correspond to a given natural language query. Within egocentric videos, this problem is particularly challenging due to the dynamic viewpoint, unstructured environments, and complex activities typical of first-person recordings. The Ego4D Natural Language Queries (NLQ) benchmark addresses this issue by providing a framework and dataset specifically designed to evaluate models' abilities to localize segments temporally based on textual descriptions.

Temporal localization in egocentric video is critical for applications such as augmented reality, video summarization, assistive technologies, and human-computer interaction. However, the inherent variability and complexity of egocentric content pose significant hurdles, including ambiguity in actions, subtle visual cues, noisy backgrounds, and diverse activity contexts.

To tackle these challenges, this project enhances a baseline model through systematic experimentation and in-depth analysis. Initially, two distinct video encoders, Omnivore and EgoVLP, are assessed to understand their influence on performance. Additionally, two language embedding strategies, BERT and GloVe, are compared to evaluate their effectiveness in accurately interpreting natural language queries within the temporal localization context.

Further extending the scope of our investigation into the Natural Language Queries (NLQ) benchmark, we enhance the analysis pipeline with both visual grounding and textual question answering. Starting from improved retrieval models, we select 50 NLQ queries with high retrieval performance and manually annotate their correct textual ground truths. For each query, we extract the relevant video segments and apply the Video-LLaVA model to generate textual answers grounded in the visual content. This allows us to derive a coherent textual explanation of what occurs within that segment. We then evaluated our approach both qualitatively and quantitatively, using metrics such as ROUGE, BLEU and RoBERTa to measure the alignment between generated answers and manually annotated references.

II. Related Work

Temporal localization of natural language queries (NLQ) within egocentric videos has seen increasing attention in recent years, particularly with benchmarks such as Ego4D setting new standards for evaluating model effectiveness. Prominent baseline methods, such as VSLNet, have demonstrated the potential of using deep neural networks to correlate language queries with relevant video segments effectively. VSLNet employs context-query attention mechanisms and boundary prediction frameworks, establishing itself as a robust baseline for temporal localization tasks.

Despite these strengths, baseline models often rely heavily on pre-extracted video features that significantly influence their final performance. To address this issue, recent work has explored the use of advanced video encoders, such as Omnivore and EgoVLP. Omnivore [1] provides rich multimodal representations that capture spatial-temporal dynamics more effectively than traditional encoders. EgoVLP further specializes in egocentric perspectives, leveraging contrastive learning paradigms tailored explicitly to first-person video understanding [2].

Additionally, language embeddings play a crucial role in model performance, with embedding choices such as BERT and GloVe significantly impacting the accuracy of temporal localization systems. BERT embeddings offer contextually rich representations through transformer-based encoders, enabling models to better capture the nuances of natural language queries. In contrast, GloVe provides efficient word-level embeddings that are more lightweight and suitable for scenarios with computational constraints. However, empirical results in this study indicate that GloVe embeddings consistently underperform compared to BERT, particularly in tasks requiring a deeper understanding of temporal and contextual relationships within egocentric video content.

Moreover, research has extended the evaluation of temporal localization models beyond standard NLQ benchmarks. Recent methodologies have combined localization outputs with downstream video question answering (VideoQA) models, such as Video-LLaVA. This combined approach validates temporal localization results and ensures comprehensive video content understanding by generating meaningful textual answers to queries based on localized video segments.

The integration of advanced video encoders, effective language embeddings, and complementary VideoQA tasks has contributed significantly to improving the robustness and accuracy of temporal localization frameworks.

Category	Template
Objects	Where is object X before / after event Y?
	Where is object X?
	What did I put in X?
	How many X's? (quantity question)
	What X did I Y?
	In what location did I see object X ?
	What X is Y?
	State of an object
	Where is my object X?
Place	Where did I put X?
People	Who did I interact with when I did activity X?
	Who did I talk to in location X?
	When did I interact with person with role X?

Fig. 1: Query template from the Ego4D NLQ benchmark

III. Methodology

This section outlines the training pipeline and model architecture used for the Ego4D Natural Language Queries (NLQ) task. We implement and compare two models, **VSLBase** and **VSLNet**, to perform temporal video grounding based on natural language queries.

A. *Dataset and annotations*

This project utilizes the Ego4D Natural Language Queries (NLQ) benchmark, a large-scale dataset comprising egocentric video clips annotated with natural language queries and temporally aligned answer segments. It is structured around real-world human activities across a diverse set of daily scenarios.

The dataset contains a total of approximately 19.000 annotated query–answer pairs, spanning diverse everyday scenarios from cooking and cleaning to mechanical tasks. The distribution of answer segment durations varies significantly across scenarios, with cooking and cleaning exhibiting longer average durations. Most query responses span approximately 20–40% of the full clip length, indicating that the model must accurately localize relevant moments within broader temporal contexts.

For each query, the annotation includes: the question text, a predefined query template (Fig. 1), and ground truth timestamps that indicate where the answer appears. In this work, we use version 1 of the Ego4D annotations and features.

Query template distribution: Analyzing the query template distribution, the frequency of the 13 query templates reveals a significant imbalance, as shown in Fig. 2. The most common templates are object-centric, such as Objects: Where is object X before / after event Y? and Place: Where did I put X?, each appearing over 2,000 times. Conversely, templates related to individuals, like People: When did I talk to or interact with person with role X?, are much rarer, with fewer than 150 instances. This imbalance suggests the model will be significantly more exposed to object and location-based queries during training.

Number of query per scenario: An analysis of the number of NLQs per scenario reveals that the most represented scenarios are *Cooking, Cleaning, Car mechanic, jobs related to construction/rennovation company, Carpenter, Scooter mechanic, Baker, Bike, Grocery shopping indoors*.

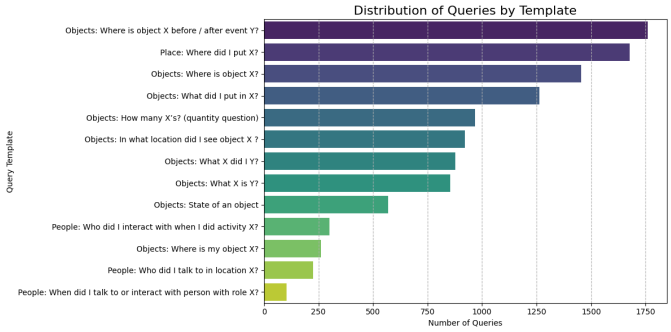

Fig. 2: Distribution of query templates in the dataset, showing the frequency of each question type.

B. *Feature Extractor*

As illustrated in Fig. 3, the first step of the pipeline consists of feature extraction.

- Videos are encoded into segment-wise features V using pretrained models, such as Omnivore or EgoVLP. The difference between them lies in the nature of their training: Omnivore is trained on a mix of video and image data, while EgoVLP is specifically trained for egocentric video understanding.
- Queries are embedded into token-level features Q using either GloVe or BERT. The key difference is that GloVe provides static word embeddings, whereas BERT generates contextualized embeddings that depend on the surrounding words.

These features are then processed through feed-forward layers and a shared encoder to obtain refined representations $\tilde{V}$ and $\tilde{Q}$.

C. *VSLBase*

VSLBase is composed of two branches that extract visual and textual features from the input video and the corresponding textual query, followed by a shared encoder block

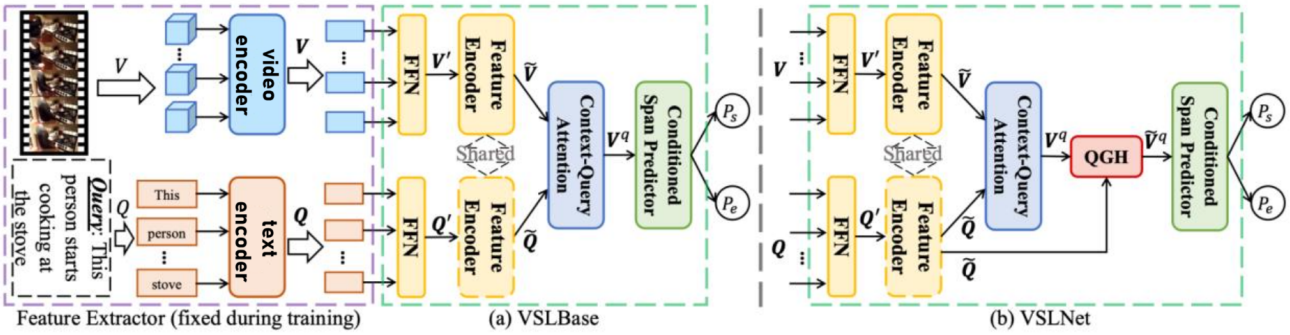

Fig. 3: Architecture of VSLBase and VSLNet.

that maps the features of both modalities into a common feature space. The Context-Query Attention (CQA) module computes the similarity scores between each visual feature and query feature using an attention-like mechanism. Finally, two unidirectional LSTMs are used to regress the temporal boundaries of the answer segment.

D. *VSLNet*

VSLNet extends VSLBase by incorporating a QueryGuided Highlighter (QGH) module, designed to direct the model's attention toward the most relevant parts of the video before span prediction. The QGH encodes the query into a sentence-level representation, which is then combined with the query-aware video features produced by the CQA module. A convolutional layer followed by a sigmoid activation generates highlighting scores that indicate each segment's relevance to the query. These scores are used to weight the video features, enhancing the model's focus on key moments. The final span predictor operates on these highlighted features, and an auxiliary loss encourages the model to identify not only the exact answer span but also the broader contextual region around it.

IV. Experiments and Results

Both models are trained using a cross-entropy loss on the annotated segment boundaries, and the training progress is monitored using TensorBoard.

To evaluate model performance on the Ego4D NLQ benchmark, we conducted a series of experiments using pre-extracted video features and natural language queries.

Training Configuration

Two distinct feature sets were utilized: Omnivore–SwinL FP16 features with a dimensionality of 1536, and EgoVLP features with a dimensionality of 256. The maximum sequence length is fixed at 128 tokens. Hyperparameters such as learning rate, batch size, and number of training epochs are not tuned empirically but set based on values commonly used in prior works for fair comparison. All models were trained using the AdamW optimizer with a fixed learning rate. The loss function combines multiple objectives, specifically localization loss and highlight loss, with the latter optionally modulated by a weighting factor λ.

Model + feature	IoU=0.3 (%)		IoU=0.5 (%)	
	R@1	R@5	R@1	R@5
VSLNet + SlowFast [3]	5.45	10.74	3.12	6.63
VSLNet + Omnivore	6.09	12.65	3.33	7.80
VSLNet + EgoVlp	**8.03**	**15.75**	**4.90**	**10.40**
VSLBase + Omnivore	0.44	5.14	0.15	3.15

TABLE 1: Comparison of Validation Results on the Ego4D NLQ Benchmark for Omnivore features

*1) **Model Architecture comparison**:* We compare the performance of VSLBase and VSLNet using Omnivore and EgoVLP features against the official SlowFast baseline [3]. The results, using a BERT text encoder, are shown in Table 1.

VSLNet surpasses VSLBase in all evaluated metrics; this performance gain is directly attributable to the Query-Guided Highlighter (QGH) module, which is absent in VSLBase.

*2) **Visual feature comparison**:* The proposed VSLNet architecture achieves the best overall performance, yielding the highest scores at almost all IoU thresholds. In particular, VSLNet + EgoVlp reaches a Rank@1 of 8.03% at IoU=0.3 and 4.90% at IoU=0.5, outperforming the original VSLNet baseline.

Our VSLNet's boost in accuracy compared to baseline VSLNet mostly comes down to swapping out the feature extractor. EgoVLP's features, learned from a large corpus of egocentric videos and narrations, are highly specialized to capture the unique nuances of the first-person perspective. Omnivore, a generalist vision transformer, performs better than the CNN-based SlowFast, demonstrating the power of modern transformer architectures. However, its features are less specialized than EgoVLP's, resulting in intermediate performance.

Feature Setting	IoU=0.3 (%)		IoU=0.5 (%)	
	R@1	R@5	R@1	R@5
EgoVlp + GLOVE	3.90	9.24	2.32	5.96
EgoVlp + BERT	**8.03**	**15.75**	**4.90**	**10.40**

TABLE 2: Comparison of language encoders for VSLNet on the Ego4D NLQ benchmark.

*3) **Textual Encoder Comparison**:* On the other hand, we further analyze the impact of different textual encoders on the VSLNet model.

The EgoVLP + BERT combination achieves the strongest results across all evaluation metrics, with a Rank@1 of 8.03% at IoU=0.3 and 4.90% at IoU=0.5, as shown in Table 2.

In conclusion, BERT's capability to generate dynamic, context-aware embeddings enables it to effectively disambiguate meaning and accurately capture the true intent of a query.

V. Extension: Video-Based Question Answering

To enhance the temporal localization capabilities of our model, we extended the project by integrating a video-based question answering (VideoQA) component. The objective was to generate natural language answers from the predicted video segments, thus deepening the interpretability and utility of the localization results. By transforming temporal predictions into coherent textual responses, we bridge the gap between low-level model outputs and high-level human understanding.

A. Segment Selection Strategy

To ensure high-quality input for the VideoQA stage, we implemented a careful filtering process to select the most accurate localization predictions from our model.

Specifically, we began by computing an error metric for each prediction, as the sum of the absolute differences between the predicted and the ground truth start and end times. This metric captured the temporal misalignment of each segment in a simple yet effective manner. Using this metric, we ranked all predicted segments and selected the top 50 with the smallest differences. These segments represented the most temporally precise localizations produced by the model, and thus were the most reliable for further semantic analysis.

This selection strategy allowed us to focus the VideoQA stage on segments where the model's temporal localization was sufficiently accurate, ensuring that any errors in the generated textual answers could be more confidently attributed to limitations in the VideoQA component itself, rather than to misaligned or incorrect segment retrieval.

B. Clip Extraction

To prepare the video input for the VideoQA stage, we needed to isolate only the portions of the full length Ego4D videos that corresponded to the selected prediction. To do so we trimmed each video based on the selected prediction intervals; we employed *ffmpeg*, a powerful multimedia processing tool, to extract these segments from the corresponding full video files.

C. Integration of Video-LLaVa

*1) **Model Overview**:* For the question answering task, we employed the Video-LLaVA-7B-hf [4] model from the PKU-YuanGroup [5]. Video-LLaVA is a multimodal generative model designed for open-ended video-based question answering. It extends the original LLaVA architecture by incorporating a video encoder that handles sequences of video frames. This design enables it to process dynamic visual information over time, making it particularly suited for complex, real-world videos such as those in the Ego4D dataset.

Specifically, the 7B version is built on top of a 7-billion parameter LLaMA-based language model and has been pretrained and aligned for multimodal understanding using large-scale instructional video datasets; this encoder captures temporal dependencies and visual dynamics across the sampled frames from a video.

The model leverages both visual and textual modalities to produce coherent answers and is capable of interpreting complex visual cues and aligning them with language prompts, making it an ideal candidate for our use case. The output is a sequence of dense visual embeddings that represent motion, object appearance, and scene context.

At the heart of Video-LLaVA is a LLaMA-7B language model, which serves as the text decoder. This model has been pretrained on massive text datasets to learn grammar, reasoning, and general knowledge. In Video-LLaVA, it has been further fine-tuned to follow natural language instructions, allowing it to answer questions in a conversational format.

*2) **Application in our Pipeline**:* To conform to the model's token limits and computational requirements, we downsampled each clip to and extracted 8 uniformly spaced frames using the PyAV library.

We used the Hugging Face transformers interface to load the model and processor. Each query was formatted into a prompt and the video frames were paired with this prompt using the VideoLlavaProcessor, and the combined input was fed into the model for inference.

We limited the output to a maximum of 80 tokens to ensure concise responses, while inference was performed on an A100 GPU to accommodate the GPU memory demands of the model, which can peak at over 30GB during generation. This setup allowed us to efficiently process all selected clips and generate coherent, semantically relevant answers for each query.

VI. Evaluation

To evaluate the quality of the generated answers, we used a combination of textual overlap metrics via BLEU and ROUGE and semantic inference via the roberta-large-mnli

model. This allowed us to assess both surface-level similarity and deeper semantic alignment between the model's output and the reference answers.

A. *Evaluation Metrics*

*1) **ROUGE and BLEU**:* Initially, we evaluated the generated answers using BLEU (Bilingual Evaluation Understudy) and ROUGE (Recall-Oriented Understudy for Gisting Evaluation), two widely used metrics for assessing textual overlap; these are standard metrics used to evaluate the quality of generated text by comparing it to one or more reference texts. BLEU focuses on precision, measuring how many n-grams in the generated output appear in the reference and it is commonly used in machine translation and rewards exact word matches.

ROUGE, on the other hand, emphasizes recall and is widely used in summarization tasks. Variants like ROUGE-1, ROUGE-2 and ROUGE-L capture unigram, bigram, and longest common subsequence overlap, respectively.

Both ROUGE and BLEU scores are typically expressed as values between 0 and 1, where:

- 0 = no overlap or similarity between generated text and reference
- 1 = perfect match or complete overlap

Together, these metrics offer a quantitative measure of how much the generated text overlaps with the expected output.

*2) **RoBERTa based NLI**:* As a second evaluation approach we used the pretrained Roberta-large-mnli [6] model from Hugging Face, which is trained to determine whether a hypothesis logically follows from a premise. It is based on RoBERTa, which stands for Robustly Optimized BERT Approach, an improved version of BERT (Bidirectional Encoder Representations from Transformers).

Instead of using string matching or token overlap (like ROUGE or BLEU), Roberta-large-mnli evaluates whether two sentences are logically aligned. That makes it very effective for evaluating answers that are phrased differently than the reference, use synonyms or paraphrasing, or vary in structure but retain the correct meaning.

The model takes as input a text pair: a premise (e.g., the reference answer) and a hypothesis (e.g., the generated answer) and returns a tensor of three logits; these logits are raw scores, one for each of the three possible logical relationships between the input sentences:

- Contradiction: the hypothesis contradicts the premise.
- Neutral: the hypothesis is neither clearly true nor false.
- Entailment: the hypothesis logically follows from the premise.

To make these scores interpretable, you apply a softmax function: This transforms the logits into a probability distribution, where the sum of the three output values equals 1 and each value represents the confidence of the model in that relationship. In our evaluation, we extracted the probability associated with the entailment, which served as a scalar similarity score between 0 and 1.

Each sample was also labeled with a general question type (e.g., color identification, object recognition, action description). We used these metadata to analyze how many samples belonged to each category and calculate the average entailment score per question type.

VideoQA results	ROUGE-L	BLEU	RoBERTa
Average output	0.165	0.024	0.359

TABLE 3: Comparison of average results for ROUGE-L, Blue and RoBERTa

B. *Observations and Results*

The average BLEU score was 0.024, as shown in Table 3, indicating that exact n-gram matches between predictions and references were extremely rare, likely due to the brevity and variability of the answers.

ROUGE metrics showed slightly higher alignment, whilst still being relatively low as ROUGE-L averaged 0.165. This value suggest that, while some answers shared key words or fragments with the ground truth, overall overlap remained limited, motivating the need for a deeper semantic evaluation.

The entailment scores obtained from the NLI-based evaluation show a wide variance, ranging from values close to 1.0 to near zero. A subset of predictions (15/50) achieved a strong semantic alignment with the reference answers, with entailment scores exceeding 0.8.

These cases indicate that Video-LLaVA was capable of generating responses that closely matched the intended meaning of the ground truth, even if phrased differently.

However, a considerable portion of the predictions (34/50) received low entailment scores, below 0.6, some dropping even below 0.01. This suggests that, in those instances, the generated answers were either irrelevant, contradictory, or lacked sufficient semantic overlap with the expected response.

The bimodal nature of the distribution, with many scores clustered at both the high and low extremes, implies that the model's performance tends to be confidently right or confidently wrong, rather than consistently mediocre. In particular, we tried to analyze the performance of the model based on the type of question, as shown in Table 4.

The model performed best on questions related to human-to-human interactions, achieving average entailment scores of 0.96 and 0.97. These categories typically required scene interpretation and object-level understanding, both of which Video-LLaVA handled reliably. On the other hand, performance dropped noticeably in more complex or abstract categories: for example, questions like 'What did I put in X?' and 'What X(adjective) is Y(object)?' achieved average scores of 0.42 and 0.48 respectively, indicating less consistent semantic alignment.

Object tracking before or after events averaged a score of 0.33, suggesting that Video-LLaVA may struggle with reasoning over temporal changes or ambiguous visual sequences. The lowest scores were found in object counting tasks, with many

Question Type	BLEU	ROUGE-L	RoBERTa Entile
What color is X?	0.000000	0.000000	0.009355
Objects: How many X's? (quantity question)	0.003600	0.027778	0.247708
Objects: State of an object	0.006433	0.164583	0.473916
Objects: What X is Y?	0.008578	0.066667	0.414865
Objects: What X did I Y?	0.022125	0.166667	0.573914
People: Who did I interact with when I did activity X?	0.023457	**0.333333**	**0.974677**
Objects: Where is object X before / after event Y?	0.024043	0.145308	0.174484
Objects: What did I put in X?	0.025194	0.153617	0.389142
Action: What X did I Y?	0.030602	0.250000	0.007303
Objects: Where is object X?	0.037218	0.240754	0.338998
Place: Where did I put X?	0.037686	0.245348	0.211847
People: Who did I talk to in location X?	**0.055069**	0.261897	0.965693

TABLE 4: Results by question type; best scores are shown in bold.

entailment scores falling below 0.1, suggesting that the model struggles significantly with quantitative reasoning and visual numerosity.

It is important to note that these results are based on a total of 50 samples, with a minimum of 1 example and a maximum of 9 examples per question type. As such, the averages reported here should be interpreted with caution due to the limited statistical significance of each category.

VII. Conclusion and Future Work

This project successfully enhanced and evaluated a temporal localization framework for the Ego4D Natural Language Queries benchmark. Through systematic experimentation, we demonstrated that the choice of both video and text encoders has a profound impact on model performance. Our findings show that the VSLNet architecture, augmented with the Query-Guided Highlighting (QGH) module, significantly outperforms the simpler VSLBase model.

The most effective combination of features was EgoVLP for video and BERT for language achieved a Rank@1 score of 4.90% at an IoU of 0.5. This highlights the distinct advantage of using video encoders pre-trained on egocentric data and language embeddings that capture rich contextual semantics. Conversely, the use of non-contextual GloVe embeddings led to a notable drop in accuracy, confirming the necessity of powerful language models for this task.

Furthermore, we extended the standard evaluation by integrating a Video Question Answering (VideoQA) pipeline using the Video-LLaVA model. This allowed for a deeper, qualitative analysis of the model's understanding of localized video segments. The evaluation of generated answers using a ROBERTa-based NLI model revealed that performance varies significantly by question type. The model excelled at identifying human interactions, achieving entailment scores over 0.96 , but struggled with tasks requiring quantitative reasoning, such as counting objects. This two-stage approach not only validates the temporal localization but also provides a path toward more interpretable and useful egocentric vision systems.

Building on the findings of this project, several promising avenues for future research can be pursued to further advance egocentric video understanding.

- **End-to-End Fine-Tuning:** In this work, pre-trained video encoders were used as fixed feature extractors. A significant next step would be to fine-tune the video encoder (e.g., EgoVLP) and the language model (BERT) jointly with the VSLNet architecture on the NLQ task. This end-to-end training could allow the encoders to adapt their representations to the specific nuances of temporal localization, potentially yielding further performance gains.
- **Expansion of VideoQA Evaluation:** The VideoQA analysis was conducted on a limited set of 50 high-quality predictions. To gain more statistically significant insights, this evaluation should be expanded to a much larger and more diverse set of queries [7]. This would involve analyzing hundreds of examples, including cases where the initial localization was less accurate, to better understand the full spectrum of failure modes.
- **Advanced VideoQA Models and Reasoning:** The struggles of Video-LLaVA with counting and temporal reasoning point to areas for improvement. Future work could explore more advanced multimodal Large Language Models (LLMs) that have demonstrated stronger capabilities in visual counting [8], spatial awareness, and temporal reasoning.
- **Systematic Hyperparameter Optimization:** The current study did not perform empirical hyperparameter tuning to ensure a fair comparison with prior works. A comprehensive hyperparameter search (e.g., for learning rate, batch size, and loss weights) would be a valuable step to optimize the models and likely unlock additional performance.

References

[1] R. Girdhar, M. Singh, N. Ravi, L. Van Der Maaten, A. Joulin, and I. Misra, "Omnivore: A single model for many visual modalities," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16102–16112, 2022.

[2] K. Q. Lin, C. Feichtenhofer, M. Zolfaghari, R. Yang, H. Fan, C. Xu, H. Wu, J. Li, Q. Zhang, S. Xie, *et al.*, "Egocentric video-language pretraining," *Advances in Neural Information Processing Systems*, vol. 35, pp. 7575–7586, 2022.

[3] G. Kristen *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," 2022.

[4] B. Lin, B. Zhu, Y. Ye, M. Ning, P. Jin, and L. Yuan, "Video-llava: Learning united visual representation by alignment before projection.," *arXiv preprint arXiv:2311.10122.*, 2023.
[5] YuanGroup, "Video-llava," *GitHub https://github.com/PKU-YuanGroup/Video-LLaVA*, 2024.
[6] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, and V. S. Mike Lewis, Luke Zettlemoyer, "Roberta: A robustly optimized bert pretraining approach," *arXiv:1907.11692v1*, 2019.
[7] Y. Zhong, W. Ji, J. Xiao, Y. Li, W. Deng, and T.-S. Chua, "Video question answering: Datasets, algorithms and challenges," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing* (Y. Goldberg, Z. Kozareva, and Y. Zhang, eds.), (Abu Dhabi, United Arab Emirates), pp. 6439–6455, Association for Computational Linguistics, Dec. 2022.
[8] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan, "Video-chatgpt: Towards detailed video understanding via large vision and language models," 2023.