Feature-Driven Natural Language Video Localization for Egocentric Video Queries

Project link on Github: https://github.com/xpuria/episodic-memory

Adel Aboutalebi Pirnaeimi
Politecnico di Torino
s324930@studenti.polito.it

Pouria Mohammadalipourahari
Politecnico di Torino
s327015@studenti.polito.it

Mohammad Saghali
Politecnico di Torino
s328134@studenti.polito.it

***Abstract*—Natural Language Querying (NLQ) in egocentric videos presents unique challenges due to the unstructured and lengthy nature of such content. In this project, we implemented and evaluated a two-step method to address Natural Language Video Localization (NLVL) tasks using the Ego4D dataset. We utilized pre-extracted features from EgoVLP and Omnivore to train two models: VSLBase and the enhanced VSLNet, which incorporates a Query-Guided Highlighting (QGH) mechanism to improve localization accuracy. The models were evaluated on the Ego4D NLQ benchmark to predict temporal video segments that correspond to natural language queries. After training and testing the models, we compared their performance to baseline results, achieving significant improvements in accuracy, particularly with VSLNet.**

I. Introduction

Egocentric vision, capturing first-person video through head-mounted cameras, has become an essential area of study in computer vision due to its ability to provide detailed insights into human activities. Unlike traditional third-person video, egocentric footage reveals interactions from the wearer's perspective, capturing fine details like hand-object interactions, attention shifts, and decision-making processes. This unique vantage point makes egocentric video analysis critical for applications such as robotics, human-computer interaction, and assistive technologies.

However, analyzing the vast amount of data generated by egocentric videos poses significant challenges, particularly when it comes to extracting meaningful information. One effective approach to tackle this challenge is through the use of NLQ, which allow users to interact with video data using everyday language. By querying specific events or objects, NLQ systems make video analysis more intuitive and accessible, removing the need for technical expertise.

The Ego4D dataset, which contains over 3,670 hours of egocentric video, is a foundational resource for this field. It provides a rich diversity of real-world scenarios and serves as the basis for evaluating NLQ systems. The Ego4D NLQ benchmark specifically measures how well models can localize video segments based on natural language queries, presenting challenges like identifying precise temporal segments and dealing with the variability of human language and activities. Overcoming these challenges can significantly improve the accessibility and usability of egocentric video data [1].

While the NLQ task effectively identifies relevant video segments, it only provides timestamps, requiring users to watch the video to find answers, which can be time-consuming. To address this, we further extended our work to enable the model to output a natural language answer directly from the corresponding video segment. By employing a Video Language Model (VLM), we can extract textual answers, enhancing the efficiency of video querying.

Despite the potential of NLQ systems, current research faces obstacles in temporal video localization and multi-modal integration—the ability to effectively link linguistic inputs with visual content. Models often struggle with understanding the complexity of human actions and localizing relevant video segments based on natural language. This project addresses these challenges by employing state-of-the-art video-language pretraining models such as EgoVLP and Omnivore. We focus on improving the performance of NLQ systems through feature extraction and fine-tuning techniques using the VSLNet architecture, aiming to enhance both accuracy and efficiency [2].

In summary, this report explores methodologies for improving NLQ-based video understanding by leveraging advanced models and comparing different feature extraction techniques. Our work contributes to the development of more effective solutions for understanding egocentric videos through natural language queries, with the potential to significantly impact applications in video analysis and interaction.

II. Related Works

The field of egocentric vision and NLVL has advanced significantly with the development of large-scale datasets and innovative methodologies. This section highlights key contributions relevant to our study.

Recent progress in egocentric vision is marked by the introduction of datasets like EPIC-KITCHENS-100 and Ego4D. EPIC-KITCHENS-100 captures detailed human interactions through head-mounted cameras, supporting various tasks such as action recognition and anticipation. This dataset enriches our understanding of everyday activities and aids research that integrates natural language queries for video comprehension [3].

NLVL has traditionally been approached as a ranking task, where multimodal matching architectures are used to identify the best video segment for a given language query [4], [5]. Some studies have reformulated NLVL as a sequential decision-making problem, employing reinforcement learning to address it [6].

Feature extraction remains a cornerstone of video analysis. The SlowFast network, by capturing both slow and fast motion dynamics, provides a robust baseline for video recognition and enhances dynamic video content interpretation [7].

III. Methodology

In this section, we outline the methodology used to address the task of NLVL, which involves identifying specific temporal moments in untrimmed egocentric videos based on natural language queries. This task requires a deep understanding of both video content and language semantics. We treat NLVL as a span-based question answering (QA) problem, where a video is processed similarly to a text passage, and the goal is to predict the start and end boundaries of the relevant video segment (answer span). To accomplish this, we leverage two video-language models: VSLBase, which follows a standard span-based QA framework, and VSLNet, which incorporates query-guided attention for improved localization. Both models process pre-extracted visual features from EgoVLP and Omnivore to streamline training. Additionally, we extend the task by exploring natural language answer extraction from localized video moments, enhancing the informativeness of the system's outputs.

A. Model architecture

This part outlines the architectures of VSLBase and VSLNet, both designed for NLVL. VSLBase uses a standard span-based framework, while VSLNet enhances this with query-guided attention for better temporal localization. The following subsections detail their key components.

1) VSLBase Architecture: The VSLBase architecture consists of four key components: feature extractors, feature encoders, context-query attention, and a conditioned span predictor [2]. An illustration of the complete architecture can be found in Fig. 1.(a).

a) Feature Extraction: To reduce computational complexity, we utilize pre-extracted visual features from Omnivore and EgoVLP models, bypassing the need for direct video frame processing. This approach is particularly efficient for large-scale datasets, where end-to-end training is impractical due to computational constraints. The video features are represented as $V = \{v_1, v_2, \ldots, v_T\}$, where each v_i corresponds to a visual feature vector. For the text query, word embeddings are extracted using BERT, with GloVe embeddings as an alternative, represented as $Q = \{q_1, q_2, \ldots, q_m\}$.

b) Feature Encoding: The *Feature Encoder* is responsible for projecting both the video and text features into a unified feature space to enable cross-modal reasoning. Both visual and textual features are transformed via linear layers into a shared dimension d, yielding the representations $V' \in \mathbb{R}^{T\times d}$ and $Q' \in \mathbb{R}^{m\times d}$ for the video and query, respectively.

The encoding process can be summarized as:

$$\tilde{V} = \text{FeatureEncoder}(V')$$

$$\tilde{Q} = \text{FeatureEncoder}(Q')$$

The encoder typically employs convolutional layers followed by attention mechanisms, which capture spatial-temporal dependencies in the video and contextual relationships in the text. This process produces the encoded features $\tilde{V}$ and $\tilde{Q}$, which are ready for cross-modal interaction in the next stage.

c) Context-Query Attention: The *Context-Query Attention (CQA)* mechanism enables the interaction between the encoded video and query features. It first calculates similarity scores between visual features and query features, forming a similarity matrix S, where each entry S_{ij} represents the similarity between the i-th visual feature and the j-th query feature.

Using this similarity matrix, CQA computes two sets of attention weights:

- *Context-to-Query Attention (A):* Highlights query features relevant to each visual feature.
- *Query-to-Context Attention (B):* Highlights visual features that are contextually relevant to the query.

These attention weights are used to aggregate and align features from both modalities:

$$A = S_r \cdot \tilde{Q}$$

$$B = S_r \cdot S_c^T \cdot \tilde{V}$$

where S_r and S_c are row-wise and column-wise normalized versions of the similarity matrix. The output is a refined representation of the visual features, which incorporates information from both the video and the query:

$$V_q = \text{FFN}([\tilde{V}; A; \tilde{V} \odot A; \tilde{V} \odot B])$$

Here, $\odot$ denotes element-wise multiplication, and FFN represents a feed-forward neural network that processes the concatenated features. This enriched feature set is then passed to the span predictor.

d) Conditioned Span Predictor: The *Conditioned Span Predictor* is designed to predict the start and end boundaries of the target video segment. It employs two unidirectional LSTMs—one for predicting the start boundary and another for the end boundary, conditioned on the hidden state of the first LSTM. This sequential design ensures that the end boundary prediction is informed by the start boundary prediction, improving the temporal coherence of the output.

Let h_{st} and h_{et} represent the hidden states for the start and end boundaries, respectively:

$$h_{st} = \text{UniLSTM}_{\text{start}}(v_{qt}, h_{st-1})$$

$$h_{et} = \text{UniLSTM}_{\text{end}}(h_{st}, h_{et-1})$$

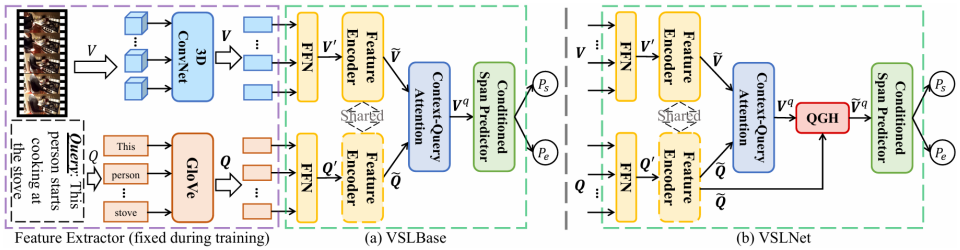

Fig. 1: An overview of the proposed architecture for NLVL. The feature extractor is fixed during training. Figure (a) depicts the adoption of the standard span-based QA framework, i.e., VSLBase. Figure (b) shows the structure of VSLNet.

The model then computes scores for the start and end positions using feed-forward layers:

$$S_{st} = W_s \cdot [h_{st}; v_{qt}] + b_s$$

$$S_{et} = W_e \cdot [h_{et}; v_{qt}] + b_e$$

where W_s, W_e are weight matrices, and b_s, b_e are bias terms. These scores are used to predict the likelihood of each frame being the start or end of the relevant video segment.

2) VSLNet Architecture: The VSLNet architecture builds upon the VSLBase framework with the inclusion of a QGH strategy, enhancing performance in NLVL. While VSLBase follows a span-based prediction approach, focusing on predicting start and end boundaries of the target moment, VSLNet introduces improvements that allow the model to better address the continuous nature of video data. A visual representation of the architecture is shown in Fig. 1.(b).

VSLBase effectively combines visual features extracted from video frames and natural language queries to predict the boundaries of the target video segment. However, it lacks the ability to adequately capture slight shifts in video frames that may affect localization performance. To address this limitation, VSLNet incorporates the **Query-Guided Highlighting (QGH)** strategy, which refines the search space by highlighting a broader region that includes the target moment and its surrounding context.

The Query-Guided Highlighting mechanism operates as follows: After the model predicts the start and end boundaries using the Conditioned Span Predictor, QGH extends these boundaries to include adjacent frames, thus providing additional context for improved accuracy. This highlighted region is divided into a **foreground**, containing the target moment and adjacent frames, and a **background**, consisting of less relevant video parts.

The model then applies an attention mechanism, focusing on the highlighted region to guide the prediction process. By leveraging this broader context, the model can better differentiate subtle visual cues that are crucial for more accurate temporal localization. The final representation of the video features integrates the query-guided contextual information, which is then passed to subsequent layers for precise span prediction.

In conclusion, the Query-Guided Highlighting strategy in VSLNet provides a focused search space by including relevant contextual frames around the predicted target moment. This allows VSLNet to perform more accurate localization compared to VSLBase by addressing challenges unique to video data and leveraging improved attention mechanisms.

B. From Video Interval to a Textual Answer

The proposed extension of this project aims to enhance NLVL by leveraging advanced Video-Language Models (VLMs), specifically Video-LLaVA, to generate textual answers from egocentric video segments. The methodology is structured as follows:

1) NLQ Query and Segment Selection: The process begins with the application of NLQ to generate initial predictions for relevant video segments. For each query, we select a specific number of video segments based on the highest Intersection over Union (IoU) scores. These scores are adapted from the Generalized IoU framework, which ensures that the segments closely align with the query's requirements. The selected video segments contain crucial information corresponding to each input query.

2) Video Segment Extraction: Following the selection of relevant video segments, we extract these segments using the FFmpeg tool to isolate the necessary intervals. This step is crucial for avoiding unnecessary computational overhead by processing only the pertinent video parts. The extracted video segments are then prepared for further analysis using the Video-LLaVA model.

3) Application of Video-LLaVA for Textual Answer Generation: The extracted video segments are input into the Video-LLaVA model along with the corresponding NLQ queries. Video-LLaVA is a powerful vision-language model designed to process and understand video content in conjunction with natural language queries.

Video-LLaVA operates by integrating and aligning visual information from video frames with textual input. This alignment enables the model to effectively interpret and generate textual responses based on the visual content of the video segments. By focusing on the relevant visual and textual information, Video-LLaVA produces accurate and contextually appropriate answers [8].

4) Evaluation Using NLP Metrics: To evaluate the quality of the generated textual answers, we use several natural language processing (NLP) metrics, including BLEU, ROUGE, BERTScore, SPICE, and Word Mover's Distance (WMD).

These metrics compare the responses generated by Video-LLaVA against manually constructed ground truth answers for each video segment. This comprehensive evaluation ensures the answers' accuracy, relevance, and fluency.

The inclusion of Video-LLaVA is a key component of this methodology. Its capability to align visual features with natural language queries allows for detailed and contextually relevant answers to be generated from video segments. Video-LLaVA's advanced processing of video data ensures that the answers accurately reflect the content and context of the visual input, enhancing the overall performance of the NLVL task.

IV. Experiments and Results

In this section, we evaluate the performance of our models on the Ego4D NLQ benchmark. We begin by providing details on the dataset and experimental setup, followed by a description of the baseline models and pre-extracted features we used. Finally, we present and discuss the quantitative and qualitative results of our experiments, as well as the extensions implemented in this study.

A. Dataset

The dataset utilized in this study is sourced from the Ego4D NLQ benchmark. The dataset spans a wide range of daily human activities, such as cooking, commuting, and object interactions, all captured from a first-person perspective. Each natural language query is temporally localized within its corresponding video, with annotated start and end timestamps indicating the relevant segments of interest [1].

The average clip length is approximately 522 seconds, with a median of 480 seconds, indicating a broad range of clip durations. The distribution of clip lengths is skewed toward shorter clips, reflecting that while most clips are relatively brief, there are also a substantial number of longer clips. This variability in clip lengths can impact how queries are mapped to video segments, suggesting the need for models that can effectively handle both short and long clips. In terms of query durations, a small fraction of queries (around 9%) last fewer than four frames, implying that filtering out these very short queries might improve model performance. Additionally, the distribution of query templates, as illustrated in Figure 2, reveals 13 distinct types, such as "Objects: How many X's?" and "Actions: What is X doing?". This diversity highlights the wide range of tasks presented by the NLQ benchmark. The temporal positions of the query answer segments are spread throughout the video clips, with many occurring centrally, as shown by the answer segment position distribution.

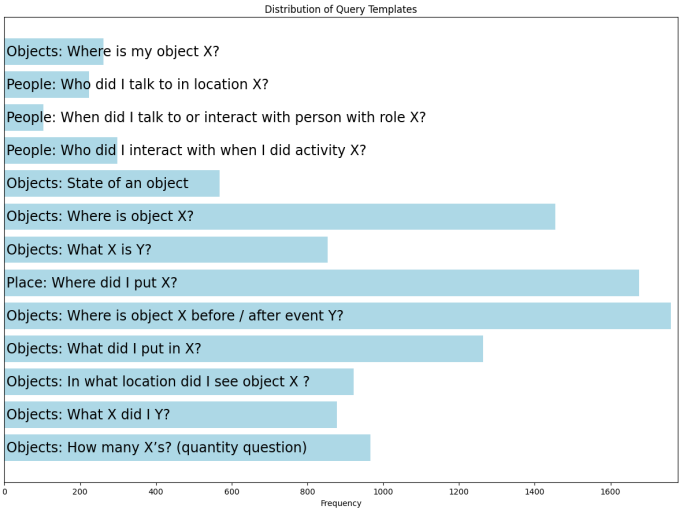

Fig. 2: Distribution of query templates: The analysis reveals 13 distinct types of queries, categorized into objects, places, people, and activities. Each query is defined by one of these 13 templates.

B. Experimental Setup

In this part, we describe the experimental setup used to train and evaluate two variants of the VSLNet architecture, VSLBase and VSLNet, on the Ego4D NLQ benchmark. The experiments were conducted using pre-extracted features from two models, EgoVLP and Omnivore, along with two different text encoders, BERT and GloVe. These experiments aim to investigate the impact of different model architectures, pre-extracted features, and text encoders on performance. Additionally, we extend our NLVL methodology by integrating Video-LLaVA to generate textual answers from egocentric video segments.

For our experiments, we trained two architectures: VSLBase and VSLNet, which are designed for the NLQ task. The primary reason for comparing these two architectures is to analyze whether the Query Guided Highlighter improves performance in identifying the relevant video segments more accurately.

The models were trained using the Adam optimizer with a learning rate (LR) of 15e-4 and 25e-4. We experimented with different values for batch size (32 and 64) and number of epochs (10 and 20) to identify the optimal configuration. Tuning these parameters is critical because they influence the stability of training and the convergence rate. The batch size affects the gradients' stability, while the learning rate controls the step size during optimization.

Training on frozen features from pre-trained models (EgoVLP and Omnivore) allows us to leverage the rich feature representations learned from large-scale egocentric video datasets without requiring computationally expensive end-to-end training.

We conducted experiments using two sets of pre-extracted video features:

- **EgoVLP**: A video-language model pre-trained on a subset of the Ego4D dataset using free-form textual narrations in a contrastive learning framework. EgoVLP is specifically designed to capture egocentric video representations and thus provides strong video-text alignments that are particularly suited for natural language queries [9].

- **Omnivore**: A general-purpose vision model pre-trained on diverse datasets, including videos, images, and 3D data. Although Omnivore is not tailored specifically for egocentric video tasks, it provides robust representations that capture both local and global temporal dependencies [10].

The motivation behind using both EgoVLP and Omnivore features is to investigate whether features extracted from a domain-specific video-language model (EgoVLP) outperform those from a more general-purpose vision model (Omnivore) in the NLQ task. This allows us to assess the benefits of task-specific pre-training in egocentric video understanding.

We also experimented with two different text encoders for the natural language queries:

- **GloVe**: A static word embedding model that maps each word in the query to a fixed vector based on word co-occurrence statistics from large text corpora. While GloVe is computationally efficient, it lacks the ability to capture contextual nuances in language, as it assigns the same embedding to a word regardless of context.
- **BERT**: A contextual word embedding model pre-trained on large amounts of text using masked language modeling. BERT generates context-aware embeddings, meaning that it can capture the subtle differences in meaning based on the surrounding words, which is crucial for understanding complex queries [11].

The decision to use both GloVe and BERT was driven by the need to assess whether a more advanced, context-sensitive text encoder (BERT) improves the model's ability to match video segments with queries compared to a simpler, static embedding model (GloVe).

We evaluated the performance of our models using the Mean Intersection over Union (mIoU) and Rank@K metrics to assess the accuracy of predicted video segments relative to ground truth annotations. Specifically:

- **Mean Intersection over Union (mIoU)** measures the average overlap between the predicted segments and the ground truth segments across all instances. The mIoU scores reported include tIoU thresholds of 0.3 and 0.5, where a threshold of 0.3 means a 30% overlap is required, and 0.5 means a 50% overlap is necessary for a prediction to be considered correct.
- **Rank@K** evaluates the retrieval effectiveness by determining how often the correct video segment appears within the top K predictions made by the model. We report results for Rank@1 and Rank@5, where the metric checks if the correct segment is found among the top 1 or 5 predictions, respectively, thereby providing insights into the model's precision in ranking the relevant segments higher.

These metrics provide a robust evaluation framework to measure both the precision of the temporal localization and the overall performance of the models across different levels of tolerance for temporal misalignment.

we outline the experimental setup for evaluating the extension of our NLVL methodology using Video-LLaVA to generate textual answers from egocentric video segments.

For each NLQ, we first generate initial predictions for video segments. We then select the top 50 video segments based on the highest Intersection over Union (IoU) scores, ensuring these segments are the most relevant and accurately aligned with the query requirements. The identified video segments are extracted using the FFmpeg tool, which isolates the necessary intervals to manage computational resources effectively by processing only the pertinent video parts. These extracted video segments are then processed by the Video-LLaVA model, utilizing the pretrained weights LanguageBind-Video-LLaVA-7B-hf. This model generates textual answers based on the visual content of the video segments and the associated NLQ queries.

For accuracy evaluation, each video segment is manually reviewed and annotated with the correct answer to its associated query. This manual annotation process involves deriving the answer directly from the visual content of the video segment. To illustrate this process, refer to Figure 3, which demonstrates the annotation assigned to a video segment and compares it with the answer generated by Video-LLaVA. This ensures that the ground truth answers used for evaluation are accurate and directly derived from the video content.

Fig. 3: As shown in the figure, for each video and query, responses are manually provided as a ground truth, and these are compared with the responses generated by Video-LLaVA.

C. Results

In this section, we present and analyze the results of our experiments, focusing on the performance of VSLNet and VSLBase architectures using EgoVLP and Omnivore feature extractors. We also provide a comparison against the official SlowFast baseline results.

1) VSLNet vs. VSLBase: VSLNet consistently outperformed VSLBase across all configurations, confirming the superiority of its architecture for the NLQ task (see Table I). VSLNet's query-guided attention mechanism enabled more precise alignment between video segments and language queries specialy by using 64 epochs, 20 batch sizes and

Model	Batch Size, Epochs, LR	IoU@0.3 Rank@1	IoU@0.3 Rank@5	IoU@0.5 Rank@1	IoU@0.5 Rank@5
VSLNet	32, 10, 0.0025	6.81	14.40	4.52	9.34
	32, 10, 0.0015	7.74	15.85	4.54	10.09
	64, 20, 0.0025	9.06	17.19	5.34	11.38
	64, 20, 0.0015	**9.34**	**17.71**	**5.86**	**11.85**
VSLBase	32, 10, 0.0025	8.05	15.41	5.24	10.48
	32, 10, 0.0015	7.12	14.30	4.65	9.60
	64, 20, 0.0025	9.03	16.52	5.34	10.76
	64, 20, 0.0015	8.96	17.68	5.09	11.38

TABLE I: Performance comparison of VSLNet and VSLBase models based on various configurations of batch size, epochs, and learning rate.

Pre-Extracred Features	Batch Size, Epochs, LR	IoU@0.3 Rank@1	IoU@0.3 Rank@5	IoU@0.5 Rank@1	IoU@0.5 Rank@5
Omnivore	32, 10, 0.0025	6.43	12.96	3.64	8.18
	32, 10, 0.0015	5.09	12.34	2.63	7.07
	64, 20, 0.0025	6.45	13.66	3.54	8.67
	64, 20, 0.0015	6.97	13.63	3.56	8.49
EgoVLP	32, 10, 0.0025	6.81	14.40	4.52	9.34
	32, 10, 0.0015	7.74	15.85	4.54	10.09
	64, 20, 0.0025	9.06	17.19	5.34	11.38
	64, 20, 0.0015	**9.34**	**17.71**	**5.86**	**11.85**

TABLE II: Performance comparison of different pre-extracted feature types with VSLNet models with BERT encoder.

0.0025 learning rate, leading to improved performance in both Rank@k and mIoU. In contrast, VSLBase, while performing reasonably well, demonstrated limitations in handling more complex queries, particularly in scenarios requiring fine-grained temporal localization.

2) Pre-extracted Features: EgoVLP vs. Omnivore: The comparison between EgoVLP and Omnivore feature extraction methods showed that EgoVLP consistently delivered superior performance across all evaluation metrics(see Table II). EgoVLP's ability to capture egocentric video dynamics allowed for better results in first-person video tasks, which are critical for the Ego4D dataset. On the other hand, Omnivore, designed for general video tasks, underperformed relative to EgoVLP, likely due to its broader, non-specialized approach, which limited its effectiveness in handling egocentric perspectives.

3) Comparison with the Official Baseline Results: When compared with the SlowFast baseline with paper used 2D Temporal adjacent Networks(2D-TAN) and VSLNet for this, both EgoVLP and Omnivore achieved better results, with EgoVLP showing the greatest improvement. The SlowFast models, while are effective in action recognition tasks, struggles with the fine-grained temporal localization required for the NLQ task in the Ego4D challenge. Our results showed that EgoVLP features significantly outperformed the SlowFast baseline,as it's shown by **Best Model*** in Table III which is the VSLNet model with EgoVLP pre-extracted features and BERT encoder, further validating the need for specialized feature extraction in egocentric video analysis [12].

4) Text Encoders: BERT vs. GloVe: Finally, our experiments with text encoders demonstrated that models using

Model	IoU@0.3 Rank@1	IoU@0.3 Rank@5	IoU@0.5 Rank@1	IoU@0.5 Rank@5
2D-TAN	5.04	12.89	2.02	5.88
VSLNet	5.45	10.74	3.12	6.63
Best Model*	**9.34**	**17.71**	**5.86**	**11.85**

TABLE III: Performance comparison of SlowFast features(2d-TAN and VSLNet) and our best-resulted model

Word Embedding Methods	Batch Size, Epochs, LR	IoU@0.3 Rank@1	IoU@0.3 Rank@5	IoU@0.5 Rank@1	IoU@0.5 Rank@5
BERT	32, 10, 0.0025	6.81	14.40	4.52	9.34
	32, 10, 0.0015	7.74	15.85	4.54	10.09
	64, 20, 0.0025	9.06	17.19	5.34	11.38
	64, 20, 0.0015	**9.34**	**17.71**	**5.86**	**11.85**
GloVe	32, 10, 0.0025	3.98	9.86	2.25	6.07
	32, 10, 0.0015	3.20	8.78	2.27	5.58
	64, 20, 0.0025	6.61	13.68	4.16	8.62
	64, 20, 0.0015	7.87	15.44	4.44	10.56

TABLE IV: Results for VSLNet with EgoVLP features and BERT/GloVe embeddings.

BERT consistently outperformed those using GloVe (see Table IV). BERT's dynamic, context-aware embeddings allowed for better alignment between queries and video content, resulting in more accurate localization. GloVe, with its static word embeddings, was less effective, particularly in handling complex or context-sensitive queries.

These results collectively demonstrate the effectiveness of task-specific feature extraction and advanced text encoders in addressing the challenges of query-based video localization within the Ego4D dataset. The combination of EgoVLP features and the BERT text encoder proved to be the most effective configuration, maximizing performance in the NLQ task.

5) Evaluating Video-LLaVA: We present the results of applying the Video-LLaVA model to the selected video segments. The evaluation of the model's performance was conducted using a range of natural language processing (NLP) metrics to assess how well the generated textual answers aligned with the manually annotated ground truth.

BLEU Score: The Video-LLaVA model achieved a mean BLEU score of 0.3615. This score reflects a moderate level of fluency and adequacy in the generated answers when compared to the reference answers. While higher BLEU scores are generally more desirable, a score above 0.3 indicates acceptable performance, particularly in scenarios where precise word order is less critical.

ROUGE Score: The mean ROUGE score of 0.6411 demonstrates a strong overlap in n-gram structures between the generated answers and the reference answers. This indicates that the model effectively captured key phrases and structures necessary to convey the correct information.

BERTScore (F1): With a mean BERTScore F1 of 0.6868, the results suggest that the generated answers align closely with the semantic content of the reference answers. BERTScore evaluates contextual similarity, and a score above 0.6 generally signifies strong semantic correspondence be-

tween the generated and reference answers.

SPICE Score: The mean SPICE score of 0.5139 reflects the model's ability to capture the underlying relationships and attributes within the generated responses. SPICE evaluates the semantic structure of the content, and a score above 0.5 indicates a good level of detail and accuracy in the generated answers.

WMD (Word Mover's Distance): The WMD score of 0.3418 reveals that while the model produced relevant answers, there is room for improvement in reducing the semantic distance between the generated and reference answers. WMD measures the distance between word distributions of the generated and reference answers, highlighting areas where semantic alignment can be enhanced.

In summary, the Video-LLaVA model demonstrated effective performance across various metrics, showing a good alignment with the reference answers. However, there is potential for further refinement, particularly in improving the semantic proximity of the generated answers to the reference answers. This evaluation highlights the strengths and areas for improvement in our approach to generating textual answers from egocentric video segments.

V. CONCLUSION

In conclusion, this project explored the challenge of NLVL in egocentric videos using state-of-the-art models and feature extraction techniques. By implementing and comparing two models—VSLBase and VSLNet—on the Ego4D NLQ benchmark, we achieved notable improvements in temporal localization accuracy. VSLNet, with its QGH strategy, outperformed VSLBase, demonstrating the importance of refined contextual attention in egocentric video tasks. Additionally, our integration of pre-extracted features from EgoVLP and Omnivore revealed the superiority of EgoVLP in handling egocentric perspectives, further supported by the advanced contextual text encoding provided by BERT.

We extended this approach by incorporating Video-LLaVA to generate natural language answers directly from localized video segments, adding an extra layer of informativeness to the querying process. This extension enhances the practicality of the system by offering more than just temporal localization, making the results more useful for users.

Overall, the project demonstrates practical advancements in video localization and query response, offering a strong foundation for further development and real-world applications.

REFERENCES

[1] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S. K. Ramakrishnan, F. Ryan, J. Sharma, M. Wray, M. Xu, E. Z. Xu, C. Zhao, S. Bansal, D. Batra, V. Cartillier, S. Crane, T. Do, M. Doulaty, A. Erapalli, C. Feichtenhofer, A. Fragomeni, Q. Fu, C. Fuegen, A. Gebreselasie, C. González, J. Hillis, X. Huang, Y. Huang, W. Jia, W. Khoo, J. Kolár, S. Kottur, A. Kumar, F. Landini, C. Li, Y. Li, Z. Li, K. Mangalam, R. Modhugu, J. Munro, T. Murrell, T. Nishiyasu, W. Price, P. R. Puentes, M. Ramazanova, L. Sari, K. Somasundaram, A. Southerland, Y. Sugano, R. Tao, M. Vo, Y. Wang, X. Wu, T. Yagi, Y. Zhu, P. Arbeláez, D. Crandall, D. Damen, G. M. Farinella, B. Ghanem, V. K. Ithapu, C. V. Jawahar, H. Joo, K. Kitani, H. Li, R. A. Newcombe, A. Oliva, H. S. Park, J. M. Rehg, Y. Sato, J. Shi, M. Z. Shou, A. Torralba, L. Torresani, M. Yan, and J. Malik, "Ego4d: Around the world in 3, 000 hours of egocentric video," *CoRR*, vol. abs/2110.07058, 2021.

[2] H. Zhang, A. Sun, W. Jing, and J. T. Zhou, "Span-based localizing network for natural language video localization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, eds.), (Online), pp. 6543–6554, Association for Computational Linguistics, July 2020.

[3] D. Damen, H. Doughty, G. Farinella, A. Furnari, E. Kazakos, J. Ma, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100," *International Journal of Computer Vision*, vol. 130, pp. 1–23, 01 2022.

[4] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "Tall: Temporal activity localization via language query," 2017.

[5] L. A. Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with natural language," 2017.

[6] W. Wang, Y. Huang, and L. Wang, "Language-driven temporal activity localization: A semantic matching reinforcement learning model," pp. 334–343, 06 2019.

[7] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[8] B. Lin, Y. Ye, B. Zhu, J. Cui, M. Ning, P. Jin, and L. Yuan, "Video-llava: Learning united visual representation by alignment before projection," 2023.

[9] K. Q. Lin, A. J. Wang, M. Soldan, M. Wray, R. Yan, E. Z. Xu, D. Gao, R. Tu, W. Zhao, W. Kong, C. Cai, H. Wang, D. Damen, B. Ghanem, W. Liu, and M. Z. Shou, "Egocentric video-language pretraining," 2022.

[10] R. Girdhar, M. Singh, N. Ravi, L. van der Maaten, A. Joulin, and I. Misra, "Omnivore: A single model for many visual modalities," *CoRR*, vol. abs/2201.08377, 2022.

[11] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018.

[12] S. Zhang, H. Peng, J. Fu, and J. Luo, "Learning 2d temporal adjacent networks for moment localization with natural language," *CoRR*, vol. abs/1912.03590, 2019.