NaQ: Leveraging Narrations as Queries to Supervise Episodic Memory

Santhosh Kumar Ramakrishnan¹ Ziad Al-Halah² Kristen Grauman¹,³

¹UT Austin ²University of Utah ³FAIR, Meta AI

Abstract

Searching long egocentric videos with natural language queries (NLQ) has compelling applications in augmented reality and robotics, where a fluid index into everything that a person (agent) has seen before could augment human memory and surface relevant information on demand. However, the structured nature of the learning problem (free-form text query inputs, localized video temporal window outputs) and its needle-in-a-haystack nature makes it both technically challenging and expensive to supervise. We introduce Narrations-as-Queries (NaQ), a data augmentation strategy that transforms standard video-text narrations into training data for a video query localization model. Validating our idea on the Ego4D benchmark, we find it has tremendous impact in practice. NaQ improves multiple top models by substantial margins (even doubling their accuracy), and yields the very best results to date on the Ego4D NLQ challenge, soundly outperforming all challenge winners in the CVPR and ECCV 2022 competitions and topping the current public leaderboard. Beyond achieving the state-of-the-art for NLQ, we also demonstrate unique properties of our approach such as the ability to perform zero-shot and few-shot NLQ, and improved performance on queries about long-tail object categories. Code and models: http://vision.cs.utexas.edu/projects/naq.

1. Introduction

Human memory can fail us in day-to-day things in our visual experience. We misplace objects in the house (where is my passport?), we lose track of what tasks we have or have not done (did I add the salt already?), we forget where we did a given activity (where did I buy tickets last time?), we do not notice the state of an object in our environment (did I leave the garage door open?). First-person or “ego-centric” perception on a wearable camera could reduce that cognitive overload and provide us with a superhuman personal episodic memory—by seeing what we see, and indexing it in meaningful and easy-to-access ways.

This is the vision of the Natural Language Query (NLQ) task in Ego4D’s Episodic Memory benchmark [16]. Given a natural language question and a long egocentric video, the NLQ task requires identifying the precise temporal window in the camera wearer’s past video that reveals the answer. See Figure 1. Such functionality could transform the everyday experience of an augmented reality user with always-on AR glasses. It could similarly play a role for a mobile household robot, whom a user may wish to query about its visual history (have you seen my keys?).

The NLQ challenge has attracted substantial attention in the research community over the last year [23, 24, 40] as have related video-language efforts for question answering [31,34,35,37–39]. The technical challenges are striking. Queries are free-form natural language, response windows are tiny slivers (a few seconds or less) within a long stretch of video, and wearable camera video is notoriously noisy with its quick head motions and limited field of view.

Today’s most successful methods embrace the visual-language aspect of the problem. In particular, inspired by the growing success of visual-linguistic embeddings [22, 25,29,33,37], top competitors on NLQ perform large-scale pretraining on (video clip, text description) pairs mined from the Ego4D dataset’s provided narrations [23], which are timestamped play-by-play descriptions of the camera-
C turns on the tap with her right hand
C opens a drawer
C cracks an egg into the bowl
C opens the third refrigerator door

Figure 2. Narration examples. “C” refers to the camera-wearer.

wearer’s activity (see Figure 2). The result is a video backbone enhanced by the semantics of grounded language.

While it is important to have strong video and text representations, the downstream query localization models that search the video for a response are also crucial to NLQ, yet relatively starved for data. This is a direct consequence of the difficulty in annotating a query-response pair (which entails posing a creative question and scrolling the long video to mark the temporal response window) versus the relative ease in narrating a video (which entails pausing the video at regular intervals and describing what happened). For example, whereas Ego4D has 3,670 hours of data annotated with narrations—more than 3.85M sentences in total—it offers only 227 hours of NLQ query examples, for 19k total text queries. Accordingly, existing methods risk failing to learn about things that are poorly represented in training, such as queries about objects in the long-tail or complex queries involving interactions between multiple visual entities.

To address this issue, we introduce Narrations-as-Queries (NaQ), a simple but exceptionally effective data augmentation strategy for NLQ. NaQ is a novel strategy that uses timestamped narrations to expand the supervision available for training query-localization modules within an episodic memory architecture. Our hypothesis is that narrations provide descriptive information that is localizable in long videos, and thus can benefit an episodic memory model when used as training queries. Specifically, we derive ⟨video, language query, temporal window response⟩ annotations from timestamped narrations, and augment the conventional query-response data with these pseudo-queries. Importantly, this allows us to influence the localization module—the workhorse responsible for finding a needle in a haystack—with multimodal data, as opposed to just the video and text encoders.

Empirically, our idea has tremendous impact. Demonstrating NaQ on the Ego4D Episodic Memory benchmark, we find our simple augmentation strategy successfully complements multiple existing state-of-the-art episodic memory methods, achieving sizeable improvements (e.g., 32% to 125% relative jumps in accuracy) across query types, metrics, and methods. Notably, our gains hold even compared to existing methods such as EgoVLP [23] that use the same (or even more) narration annotations as our model, meaning that NaQ’s success can be attributed to good modeling, not more data. Moreover, NaQ even benefits video-language grounding on exocentric videos, i.e., it is beneficial to augment its exocentric training with narrated egocentric videos. To our knowledge, NaQ yields the very best results to date on the NLQ challenge, strongly outperforming all the challenge winners from Ego4D CVPR’22 and ECCV’22, and topping the current public leaderboard. Beyond achieving state-of-the-art results, we perform a thorough analysis of the strengths and weaknesses of NaQ, and demonstrate useful properties such as benefits on long-tail object queries as well as zero-shot and few-shot NLQ.

2. Related work

Egocentric video understanding. Prior work has developed video datasets and methods for egocentric perception [7, 11, 13, 16, 19]. Egocentric video offers a camera wearer’s perspective of their activities over a long time horizon and raises challenging research problems such as human-object interactions [4, 8], activity recognition [19, 42], anticipation [1, 15], episodic memory [16], and video summarization [9, 21]. In this work, we tackle the episodic memory task. We leverage the Ego4D dataset [16], which consists of 3,670 hours of video of daily-life activity captured by 931 camera wearers around the world.

Vision-language pretraining (VLP). VLP methods rely on large-scale video-text datasets [3, 26] to learn transferable representations for video-language tasks such as retrieval [10, 17], question-answering [31, 35] and video captioning [20, 41]. VideoBert learns joint video-text embeddings by discretizing video frames and performing BERT-like pre-training [33]. HERO improves over this with a hierarchical encoding of multi-modal inputs [22]. MIL-NCE learns to match clips with temporally close captions to address video-text misalignment in HowTo100M [25, 26]. While these methods primarily focus on third-person videos, EgoVLP [23] adapts the InfoNCE objective to egocentric settings and uses video-narration annotations from Ego4D [16] to learn video-text backbones for ego-video understanding. Just-Ask [37] proposes a strategy to generate video question-answering data consisting of (short clips, questions, text answers) from narrated YouTube videos.

While we take inspiration from such methods, our idea is very different. Unlike prior work that learns representations or video-QA systems from short video clips and (possibly weakly) aligned text, we learn to temporally localize short text queries in long untrimmed videos egocentric videos.
Whereas Just-Ask’s data generation procedure [37] outputs questions with text responses for short video clips, ours outputs temporal windows in long videos. Rather than pretraining a video/text backbone [22, 23, 25, 33], our model injects multimodal supervision to train a query-localization module. Overall, our idea is complementary to prior video-text pretraining efforts, as we will demonstrate in the results.

**Video-language grounding.** Prior work performs video-language grounding (VLG) in exocentric videos [14, 20, 30, 38, 39]. The Ego4D episodic memory benchmark first introduced NLQ, a new VLG task requiring temporal query localization in long egocentric videos [16]. Existing VLG methods like 2D-TAN [39] and VSLNet [38] have been adapted to perform NLQ, while the recent ReLER [24] model achieves state-of-the-art NLQ using a multi-scale and cross-modal transformer with video-level data augmentation. Our goal is to improve such methods via large-scale data augmentation with narration-based queries. In addition, our proposed strategy performs query-level augmentation and is complementary to the video-level data augmentation from [24]. Recent work uses point-wise (aka “glance”) annotations to reduce annotation costs for VLG training [6, 36]. However, these are limited to exocentric videos and assume task-specific point annotations, whereas the Ego4D narrations are not specific to the NLQ task. As we will demonstrate in experiments, our approach stacks well when combined with prior NLQ methods [23, 24, 38], and can even benefit exocentric VLG via an ego-exo transfer of the egocentric narrations.

### 3. Approach

Our key insight is to leverage narrations as an additional data source to improve a model’s ability to localize answers in a long video when prompted with a natural language query. To do this, we propose a strategy to convert narrations and their timestamps into NLQ annotations. Our strategy is automatic and simple which allows us to scale the training data for episodic memory search by two orders of magnitude.

We first define the episodic memory task (Sec. 3.1), then our Narrations-as-Queries approach to convert narrations into NLQ annotations (Sec. 3.2), and finally our training strategy (Sec. 3.3).

#### 3.1. Episodic memory with natural language query

The goal of episodic memory is to perform query-driven reasoning about long-form egocentric videos. First introduced in Ego4D [16], it is well-motivated by applications discussed above, such as augmented reality assistants that enable superhuman memory. The NLQ task has attracted significant attention in the research community, with 10+ teams around the world competing on the benchmark over the last year [23, 24, 40], organized challenges at CVPR’22 and ECCV’22, and an active public leaderboard.1

More formally, given an egocentric video \( V \) capturing a camera wearer’s past experiences and a natural language query \( Q \) in the form of a question, the task requires temporally localizing where the answer can be seen in the video, i.e., a response window \( R = [t_s, t_e] \). For example, the query could be \( Q = \text{“What vegetables did I put in the soup the last time I made it?”} \), and the model needs to search a given video \( V \) to identify the time window \([t_s, t_e]\) that contains the answer, i.e., the type of vegetables in the soup. A data sample for this task is of the form \( \langle \text{video}, \text{query}, \text{response} \rangle \). The video can be several minutes long, and the response to the query can appear in a time window that is shorter than a second, making this a very challenging task.

#### 3.2. Narrations-as-Queries

Prior NLQ methods are limited in performance due to the lack of large-scale NLQ annotations of the form \( \langle \text{video}, \text{query}, \text{response} \rangle \). We address this limitation by proposing a method to automatically transform narrations associated with egocentric videos to a compatible form for NLQ. Narrations are free-form sentences describing the current activity performed by the camera-wearer (see Fig. 2). They are time-stamped and temporally dense (e.g., there are 13.2 sentences per minute of video on average in Ego4D [16]).

These annotations are substantially cheaper to obtain than NLQ annotations. For narrations, the annotators needs to simply describe the activity that is seen in the video; whereas for NLQ, first a meaningful, unambiguous question needs to be formulated and then the annotator needs to manually search the video back and forth to identify the time window that shows the answer. Hence, narrations can be annotated at a much larger scale compared to NLQ (e.g., Ego4D has 3.85M narrations vs. 19k NLQ samples). Moreover, narrations have several applications beyond NLQ [2, 5, 23, 28], and are likely to be invested in on a large-scale.

Our idea is to leverage this massive data source to aid learning for the NLQ task. We achieve this by first generating a temporal window associated with each narration that approximately captures when the activity described by the narration started and ended. Then, we use these samples (narrations coupled with temporal windows) as additional supervision to train an NLQ localization model to identify where these narrations happen in the video (see Fig. 3). Next, we formally describe our approach in detail.

1. **Generating temporal windows for narrations.** Each video narration consists of a textual sentence \( T \), and a single timestamp \( t \) marking the correspondence to the underlying video (see Fig. 3, left). However, this is incompatible with

1Ego4D NLQ challenge: https://eval.ai/web/challenges/challenge-page/1629/overview
NLQ task architectures which require queries and temporal response windows as supervision. To address this, we propose temporal response jittering, a technique to convert narration timestamps to temporal windows conditioned on the video.

**Temporal response jittering:** Our goal is to convert a narration timestamp $t_i$ from video $V_j$ into a response window $R_i = (t_s, t_e)$. First, we use “contextual variable-length clip pairing strategy” introduced in EgoVLP [23] to obtain a video-conditioned seed temporal window centered around $t_i$:

$$\bar{R}_i = [t_i - \beta_j / 2\alpha, t_i + \beta_j / 2\alpha]$$  \hspace{1cm} (1)

where $\beta_j$ captures the average temporal length between consecutive narrations in video $V_j$, and $\alpha$ is the average of all $\beta_j$ across all videos (please see [23] for details). While this offers a good starting point, it fails to address the inherent noise in $\bar{R}_i$ arising from the lack of explicit human annotation. The responses generated are also typically short (less than a second) and do not match the distribution over NLQ response windows that are 10 seconds long on average. To account for these factors, we transform $R_i = (t_s, t_e)$ further using a randomized window expansion and translation:

$$R_i = [(\bar{t}_c - \delta_t) - s\Delta, (\bar{t}_c - \delta_t) + s\Delta],$$  \hspace{1cm} (2)

where $\Delta = (\bar{t}_e - \bar{t}_s)/2$ is the half-width of $\bar{R}_i$, $\bar{t}_c = (t_s + t_e)/2$ is the center of $\bar{R}_i$, $s \sim U[1, S]$ is an expansion factor, and $\delta_t \sim U[-T, T]$ is a translation factor. Intuitively, the translation factor $\delta_t$ randomly shifts $\bar{R}$ to model uncertainty in its estimate, and the scaling factor $s$ randomly expands $\bar{R}$ to match the distribution of NLQ response windows. $S$ is a hyperparameter selected through validation, and $T$ is set as $(s - 1)\Delta$ after sampling $s$ to ensure that the seed temporal window $\bar{R}_i$ is contained within $R_i$.

Following this strategy, we can extract narrations and their inferred temporal windows for all video clips with available narrations (denoted by $V$) to obtain a dataset

$$D = \{(N^v_i, \cdots, N^v_n) \mid \forall v \in V\},$$  \hspace{1cm} (3)

where $N^v_i = (T, R_i)$ is the transformed sample that consists of a narration and its corresponding response window.

We apply this method to the video clips from the train split of the Ego4D Episodic Memory benchmark to create a dataset $D$ that contains 850k samples of transformed narrations from 4,851 video clips.

2. Generating episodic memory queries. Given the previous dataset of narrations with associated temporal windows $D$, we now convert these to a dataset of NLQ queries. Specifically, given a video $V_j$, we sample a narration $N_j$ from $V_j$ and obtain the task input $X = (V_j, T_j)$, where $T_j$ is the narration text, and the label $Y = R_i$ which represents the start and end times for a narration as defined in Eq. (2). In other words, the narration $T_j$ becomes the query that effectively asks the model to locate in $V_j$ where the activity described by $T_j$ can be found, i.e., the response window $(t_{\text{start}}, t_{\text{end}})$. We found that simply using narration text as the query to work well. This can be attributed
to pretrained BERT query encoders used in NLQ models [23, 24, 38], which can effectively adapt to the difference between declarative sentences and questions. However, it would be interesting to study techniques to transform narrations to questions in future work [37]. This dataset of \((X, Y)\) pairs is our Narrations-as-Queries (NaQ) dataset. Next, we incorporate this dataset into the NLQ training pipeline as a form of data augmentation.

3.3. Narrations-as-Queries training for NLQ

Our NaQ is model-agnostic: it stands to benefit any NLQ model out of the box without any model-specific modifications. We demonstrate the universal advantage of NaQ by benchmarking several baselines with NaQ in experiments.

Specifically, for a given NLQ model \(M\), we train it with NaQ in two stages. Let us denote the NaQ dataset as \(D_{NaQ}\) and the NLQ train dataset as \(D_{NLQ}\). First, we jointly train \(M\) with both \(D_{NaQ}\) and \(D_{NLQ}\), effectively treating NaQ as a query augmentation strategy. Since NaQ expands the training dataset significantly (by 2 orders of magnitude in size), we rely on large batch training with 2048 batch size and an appropriately large initial learning rate of 0.001 on 4-8 A40 GPUs. We train in this large-batch setting for 200 epochs, with early stopping when the validation performance saturates. We then finetune the model on \(D_{NLQ}\) with the default small-batch training used for \(M\), and perform a grid search to determine the learning rate based on \(M\) performance on the validation split.

4. Experiments

We evaluate our approach on the NLQ task from the episodic memory benchmark from Ego4D [16]. This benchmark has gained significant interest and has been the subject of two Ego4D challenges held at CVPR 2022 and ECCV 2022. The NLQ task contains 11.3k / 3.9k / 4k queries annotated over 136/45/46 hours of train / val / test videos. Each video clip is 8.2 minutes on average, and the ground-truth query response is 10.5 seconds on average in the train dataset. That means the response window occupies only 2% of the input video on average. We primarily perform experiments on Ego4D since it is consistent with our episodic memory motivation and uniquely supports our setting with a combination of (egocentric videos, NLQ annotations, large-scale narrations). We additionally experiment on the TACoS dataset of exocentric kitchen videos to test the generalization of our approach [30]. It contains 10k / 4.5k queries annotated over 75/25 train / val videos, and offers long videos with short response windows.

Evaluation metrics. We measure performance on NLQ using metrics from the video-language grounding literature and adapted for NLQ in [16]. We report the recall@k, IoU=m metric, where \(k = \{1, 5\}\) and \(m = \{0.3, 0.5\}\). This measures the percentage of times where at least one of the top-k predicted candidates have at least an intersection-over-union (IoU) of \(m\).

Baselines. We evaluate the impact of NaQ by combining it with 3 existing methods in the literature.

1) VSLNet treats natural-language grounding as a text-based question answering problem [38]. It represents the input video as a text passage and uses a span-based QA framework [32] to localize responses to text queries. This was adapted to perform the NLQ task in [16] by using Slow-Fast features pretrained on Kinetics 400 [12].

2) EgoVLP proposes to pretrain video and text backbones on the EgoNCE pretraining task [23]. By leveraging large-scale video + text narrations from Ego4D, they successfully transfer features to a variety of tasks including NLQ. It was the runner-up entry for the Ego4D NLQ challenge at CVPR 2022. This method replaces the Slow-Fast features from the VSLNet baseline with the EgoVLP pretrained backbones. This baseline is complementary to our own approach where we use narrations to augment the localization training for the NLQ task.

3) ReLER adapts VSLNet to use a multi-scale cross-modal transformer architecture [24]. It also proposes to augment the training data using video-level augmentation strategies like randomly sampling a subset of the video to try and mitigate overfitting. This was the winning entry of the Ego4D NLQ challenge at CVPR 2022. We augment ReLER with EgoVLP pretrained backbones to obtain a stronger ‘ReLER*’ baseline. Unlike ReLER, which augments the data at the video level, we propose to augment the data at the query level. We will demonstrate that NaQ is complementary and boosts the performance of ReLER.

Note that NaQ leverages the same narrations as EgoVLP and ReLER*, and requires no greater supervision or data.

Implementation details. For each baseline, we adapt the authors’ code to train with NaQ data augmentation. For consistency, we report the results of each method as reproduced using the provided code, in addition to reporting the official paper numbers. We train each method with NaQ augmentation for 200 epochs and stop training early when the validation performance saturates. We found that it was helpful to finetune for up to 30 epochs on only the NLQ dataset. Please see Sec. S1 in supp. for details.

4.1. Experimental results on Ego4D NLQ

We report results on the NLQ validation set in Tab. 1. The poor performance of the VSLNet baseline on NLQ highlights the difficulty of the task. It requires localizing responses typically shorter than 10 seconds in 8+ minute long egocentric videos. The limited size of the training dataset further exacerbates this problem, since there are only 11.3k training queries. However, when augmented with NaQ,
the performance across all metrics nearly doubles, indicating the effectiveness of NaQ in addressing these challenges. This is a dramatic gain, though it comes at the cost of larger narrations data that is not available to VSLNet.

When VSLNet is augmented with NaQ, it is already competitive with EgoVLP, which pretrained video and text backbones with Ego4D videos + narrations and uses the same VSLNet query-localization architecture (rows 3 vs. 5). When NaQ is combined with EgoVLP, it further improves the performance by 3.2 - 7.1 points across metrics (rows 5 vs. 6). This confirms that NaQ augmentation for query-localization training complements the EgoVLP pretraining of video-text backbones. Importantly, our gain here comes at no additional cost in data or annotations.

ReLER [24] uses SlowFast + CLIP video features. For a fair comparison, we replace the SlowFast features with EgoVLP features to obtain ReLER*. This improves by a large margin as expected, and gives us a stronger baseline to compare with (rows 8 vs. 9). Recall that ReLER* uses video-level data augmentation using variable-length sliding windows and video splicing [24]. When ReLER* is augmented with NaQ, the performance increases by a significant margin. This confirms the complementary nature of the query-level augmentation we propose in NaQ with video-level augmentation in ReLER.

Overall, we find that NaQ augmentation greatly improves the performance of all methods across all metrics. The absolute gains across metrics are remarkably consistent regardless of the underlying method. When averaged across the methods, NaQ improves the absolute recall@1 performance by 5.06 at IoU=0.3 and 3.08 at IoU=0.5, and the absolute recall@5 performance by 6.88 at IoU=0.3 and 4.98 at IoU=0.5. This confirms the generality and effectiveness of NaQ at expanding the limited NLQ annotations by bootstrapping it with narrations, a relatively cheaper and more abundant data source. More importantly, the insight in NaQ is not simply that large-scale data benefits performance. Rather, we emphasize how to use this data: we leverage narrations as queries for query-localization network training. This is evidenced by our experiments demonstrating major gains on EgoVLP and ReLER*, methods which also benefit from large-scale pretraining on video-narrations data.

### Ego4D NLQ challenge

We submitted our best performing method (ReLER* + NaQ) to the Ego4D NLQ challenge leaderboard, where the NLQ evaluation is performed on an EvalAI server on a held-out set of test annotations [16]. Note that while the videos are available to participants, the annotations (including narrations) are not accessible. The results are shown in Tab. 2. VSLNet is the baseline provided by the organizers. ReLER and EgoVLP were the winning and runner-up entries from the CVPR 2022 edition of the challenge. InternVideo [5], Badgers@UW-Madison [27], and CONE [18] are the top three entries from the ECCV 2022 edition of the challenge. At the time of submission, NaQ was the leading entry among all methods on the leaderboard. Post-acceptance, we combined NaQ with the ECCV and CVPR challenge winners (i.e., ReLER architecture with InternVideo features) to obtain NaQ++. Our results set the state-of-the-art for NLQ, outperforming prior work by a large margin. NaQ++ is also the official baseline for the CVPR 2023 Ego4D NLQ challenge.

### TRJ ablation

We study the impact of using temporal response jittering (TRJ) (Sec. 3.2) in an ablation study. We observe that using TRJ improves the performance by up to 0.8 points in recall @ 1 metrics and 1.6 in recall @ 5 metrics consistently across all methods. Please see Sec. S3 for the complete results.

### 4.2. Experimental results on TaCOS NLQ

Existing third-person (aka exo) video datasets for language grounding lack large-scale narrations, which prevents a direct analogue of our experiments in exo videos. Therefore, we perform an ego-video variant using the TACoS dataset of exo kitchen videos [30], where we jointly train on the NaQ dataset from Ego4D’s (ego videos, narrations) and (exo videos, language queries) from TACoS. See Tab. 3.
**4.3. Performance analyses**

In the previous section, we verified the effectiveness of our approach through a careful comparison with recent state-of-the-art methods. We now ascertain the strengths and weaknesses of our approach through a series of quantitative studies and discuss qualitative results in Fig. 4. For performing analysis-specific experiments, we adopt the NaQ method since it requires lower computational cost and time to train.

(1) **How does performance scale with narrations?** One of the key benefits of using narrations for pretraining is that they are available on a large scale. We generated 850k narrations as queries for the NLQ task, which is two orders of magnitude larger than the NLQ dataset containing 11.3k train queries. We now study performance scaling as a function of the amount of narrations used for training. For this, we add 10%, 25%, and 50% of the narrations (1 query per 1.4 minutes of videos on average). How- ever, since narrations are more densely annotated throughout the video (20+ narrations per minute), they contain rich information about objects that are rarely queried about. We therefore study if pretrained NLQ localization models with narrations can help respond to queries about long-tail objects. We divide objects from the NLQ train annotations into 3 types (as shown in Fig. S1): 1. **high-shot objects** which are queried more than 50 times (65 in total), 2. **mid-shot objects** which are queried about 10 to 50 times (147 in total), and 3. **low-shot objects** which are queried about between 2 to 10 times (967 in total). The results are in Tab. 4.

<table>
<thead>
<tr>
<th>Object Type</th>
<th>VSLNet R@1</th>
<th>VSLNet R@5</th>
<th>EgoVLP R@1</th>
<th>EgoVLP R@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>High-shot</td>
<td>23.39</td>
<td>19.13</td>
<td>18.24</td>
<td>13.78</td>
</tr>
<tr>
<td>Mid-shot</td>
<td>32.69</td>
<td>26.43</td>
<td>27.25</td>
<td>20.03</td>
</tr>
<tr>
<td>Low-shot</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 4. Performance breakdown across object types.

For object type queries, we categorize objects into low-shot, mid-shot, and high-shot objects based on their frequency of occurrence. We report the recall@1 metric at IoU=0.3 and IoU=0.5. We highlight cases where NaQ improves recall over the baseline.

(2) **What types of queries does NaQ benefit?** Next, we break down the NLQ performance across query types, i.e., the form of reasoning required by the query (e.g., **where did I put object X?** who did I talk to while doing activity Y?). The NLQ dataset was created by providing an initial set of 13 query templates [16]. For reliable evaluation, we select 10 out of the 13 templates which contain 100 or more samples in the validation split, and report results in Tab. S3 in supplementary. We observe that using NaQ leads to significant improvements (marked in green) on 8/10 templates for at least 2/3 methods. However, it only has a limited impact for **‘Where is object X?’** and **‘In what location did I see X?’** queries. These queries may require explicit spatial understanding to achieve better performance. Since all methods perform poorly on those queries and do not benefit from training on NaQ, it hints at the need to incorporate better spatial understanding for video models.

Table 3. Results on TACoS dataset of third-person cooking videos.

<table>
<thead>
<tr>
<th>Method</th>
<th>R@1</th>
<th>R@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSLNet</td>
<td>6.28</td>
<td>5.53</td>
</tr>
<tr>
<td>EgoVLP</td>
<td>13.15</td>
<td>10.20</td>
</tr>
<tr>
<td>NaQ</td>
<td>16.59</td>
<td>13.46</td>
</tr>
<tr>
<td>NaQ</td>
<td>17.07</td>
<td>13.46</td>
</tr>
</tbody>
</table>

**Table 3. Results on TACoS dataset of third-person cooking videos.** NaQ benefits both VSLNet and EgoVLP. EgoVLP underperforms VSLNet since its video features were pretrained on ego videos, while VSLNet uses SlowFast features pretrained on Kinetics 400. Though the performance gains from NaQ are lower than on Ego4D due to the ego/exo domain mismatch, these results reinforce our method’s generality.

(3) **Does NaQ help respond about long-tail objects?** The NLQ dataset has a long-tail of objects that are the subject of queries due to the sparse nature of NLQ annotations (1 query per 1.4 minutes of videos on average). However, since narrations are more densely annotated throughout the video (20+ narrations per minute), they contain rich information about objects that are rarely queried about. We therefore study if pretrained NLQ localization models with narrations can help respond to queries about long-tail objects. We divide objects from the NLQ train annotations into 3 types (as shown in Fig. S1): 1. high-shot objects which are queried more than 50 times (65 in total), 2. mid-shot objects which are queried about 10 to 50 times (147 in total), and 3. low-shot objects which are queried about between 2 to 10 times (967 in total). The results are in Tab. 4. Overall, we observe that NaQ improves performance by a large margin in most cases, and has the biggest gains on mid-shot and low-shot objects. This indicates that using narrations as queries helps mitigate some of the biases in the NLQ data, and improves responses to queries about less-frequently occurring objects.

(4) **Does NaQ facilitate zero-shot / few-shot NLQ?** Considering that NaQ enables better performance on long-tail objects, we next study whether it can facilitate zero-shot or few-shot learning for NLQ, i.e., given our large-scale NaQ data and little to no NLQ task annotations, can we learn good NLQ models? We are first to study this to the best of our knowledge. We train EgoVLP + NaQ method with all of NaQ and k% of NLQ train data, where k = {0, 10, 25, 35}. k = 0 represents the zero-shot case, and the rest represent few-shot learning. The results are in Fig. 6. The triangles represent EgoVLP + NaQ with k% NLQ data, and the horizontal line represents the EgoVLP baseline with no NaQ data. It is interesting to observe that even with no NLQ data,
perform EgoVLP by a large margin. This study suggests that we can leverage large-scale free-form narrations using NaQ to compensate for the lack of NLQ annotations. While these are not free to obtain, they are easier to annotate than NLQ and can also be used for various purposes other than the NLQ task itself [16], meaning that many research directions are likely to continue investing in them.

5. Conclusions

We propose Narrations-as-Queries, a simple data augmentation technique that dramatically improves state-of-the-art results on the Natural Language Queries task in the Ego4D Episodic Memory benchmark. Our key insight is to convert timestamped narrations in egocentric videos into natural language query annotations and use them as additional data for training NLQ localization models. To convert timestamped narrations into a form compatible with NLQ, we propose a temporal response jittering technique to convert a single timestamp into temporal windows. We perform experiments to demonstrate that our approach can be used as a simple plug-in to existing methods, massively improves multiple top methods for this task, and yields the very best performance to-date on the Ego4D NLQ benchmark. We hope that our approach serves as a useful tool for future research on this problem. Code, data, and models are available.

6. Acknowledgements

UT Austin is supported in part by the IFML NSF AI Institute and NSF CCRI. KG is a paid as a researcher at Meta.
References


[25] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end...


