LIST: Language-Guided and Interpretable Sub-Trajectory Retrieval for Imitation Learning

Fatemeh Bahrani
Independant Researcher
nf.bahrani@gmail.com

Amin Banayeeanzade
Department of Computer Science
University of Southern California

Erdem Bıyık
Department of Computer Science
University of Southern California

Yuchen Cui
Department of Computer Science
University of California, Los Angeles

Abstract: Imitation learning is a widely used paradigm for robot learning, but it requires a large amount of data and performs poorly when only a few demonstrations are available. To address this issue, data retrieval methods augment the training dataset of the target task with additional similar samples from a larger prior dataset. Most existing retrieval methods embed both target and prior demonstrations into a shared latent space and retrieve based on their similarity in this space. Although they provide some improvements, these methods rely on embedding spaces whose features are difficult to interpret, making it hard to explain and control the retrieval process. In this paper, we propose LIST: a framework for interpretable sub-trajectory retrieval using natural language. LIST first identifies sub-trajectories within every demonstration using a general-purpose vision-language model and generates textual descriptions for them. It then uses an LLM as a judge to score the similarity between prior and target sub-trajectories based on their textual descriptions. LIST then retrieves the highest-scoring segments from the prior dataset to augment the training dataset. Experimental results show that LIST outperforms existing retrieval methods on LIBERO. Moreover, LIST, by design, provides an interpretable interface since the comparisons happen directly in the text space, enabling human experts to intervene and steer data retrieval according to their experimental requirements and observed failures. Overall, LIST demonstrates that language-based retrieval provides an effective approach for data-efficient robot learning.

Keywords: Data Retrieval, Imitation Learning, Interpretable Retrieval

1 Introduction

Imitation learning (IL) is widely used to train robots for complex manipulation tasks, but the cost of collecting expert demonstrations for a target task is often a major bottleneck that limits the success of the trained policies. Large prior datasets spanning multiple diverse tasks offer additional training data, but training on all of them is costly and may even harm the performance on the target task by introducing irrelevant demonstrations. Data retrieval approaches address this issue by augmenting the small set of demonstrations for the target task with relevant demonstrations from the prior dataset. This aggregate set of demonstrations is then used to train a policy for the target task.

The central challenge is to define and measure similarity of demonstrations in prior and target sets. Existing methods offer different answers [1, 2, 3], yet they typically follow a common approach: demonstrations are embedded into a feature space, using VAEs [4] or other self-supervised representation learning models, and similarity is defined by distances in the resulting latent space. However, learned features in representation spaces are often difficult to interpret [5]. This limits our ability to determine why a particular demonstration is considered relevant or to control which aspects of behavior should drive the retrieval. We therefore ask: *how do we design an effective retrieval process that is interpretable and controllable?*

10th Conference on Robot Learning (CoRL 2026), Austin TX, USA.

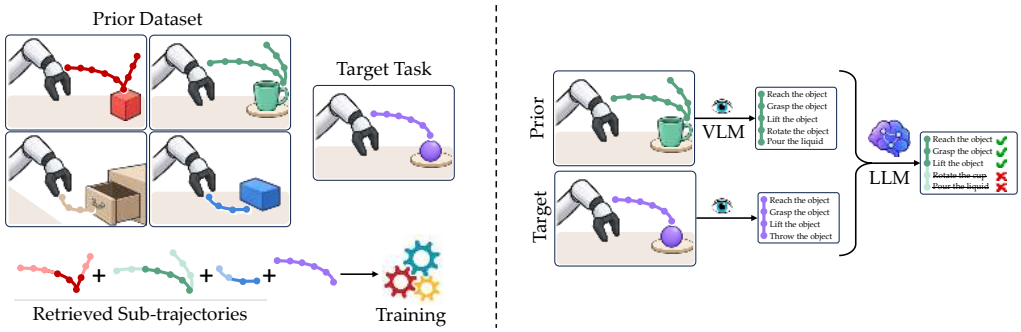


Figure 1: Overview of LIST. A VLM describes prior and target trajectories in natural language, and an LLM matches these descriptions to retrieve relevant sub-trajectories for policy training.

In this work, we propose LIST: **L**anguage-Guided and **I**nterpretable **S**ub-**T**rajectory Retrieval for Imitation Learning. Our key novelty is to use natural-language descriptions of robot trajectories as the representation space. In particular, LIST first uses a vision-language model (VLM) to divide each prior and target demonstration into sub-trajectories, with each segment corresponding to a meaningful subtask. The VLM also generates a natural-language description for each segment, capturing attributes such as object types and robot motion. A large language model (LLM) then compares each prior segment with the target segments and assigns a similarity score according to a predefined rubric. Finally, we retrieve the prior segments with the highest scores and use them combined with the target task demonstrations to train a policy model. Figure 1 provides an overview of the LIST pipeline.

We evaluate LIST on the LIBERO [6] benchmark under the few-shot setting used by prior retrieval methods, retrieving from the LIBERO-90 as the prior dataset to solve LIBERO-10 target tasks. LIST improves task success rate over naive Behavior Cloning (BC) and existing baselines. Additionally, LIST allows human experts to monitor and intervene in the retrieval process by identifying failures of the trained policy and accordingly enhancing the retrieval rubric through simple prompt modifications. We provide demonstrative examples to support this claim.

In summary, our contributions are: (i) An interpretable retrieval framework for policy learning. We use VLM-generated language descriptions of sub-trajectories for retrieval, replacing embedding similarity with semantic matching over task descriptions. (ii) An empirical study on LIBERO. Under matched policy architectures, we show that LIST outperforms prior retrieval methods.

2 Related Work

Prior work has explored data retrieval as a mechanism for leveraging relevant demonstrations from large-scale robot datasets to improve performance on new tasks with limited supervision. Behavior Retrieval (BR) [1] learns a joint state-action embedding with a VAE and scores every prior transition by its L2 distance to the target demonstrations. Flow Retrieval (FR) [2] replaces visual-appearance similarity with motion similarity by learning a VAE over optical flow. SAILOR [7] employs an auxiliary temporal predictability objective via a VAE to group temporally similar sub-trajectories closely in a latent skill space.

Moreover, STRAP [3] extends sub-trajectory retrieval by replacing task-specific learned embeddings with features from a frozen vision embedding foundation model, followed by Subsequence Dynamic Time Warping [8] for selecting the retrieved dataset. COLLAGE [9] retrieves candidate sub-trajectories under several complementary features and combines them for each target task according to how well the data retrieved by that modality trains a reference policy.

IWR [10] uses importance weights, estimated via Gaussian kernel density estimation, to select data that maximizes the likelihood under the target task distribution. DataMIL [11] selects co-training data by estimating per-trajectory influence on policy performance via validation BC loss. Additionally, another line of research uses human hand demonstrations of the target tasks to retrieve similar

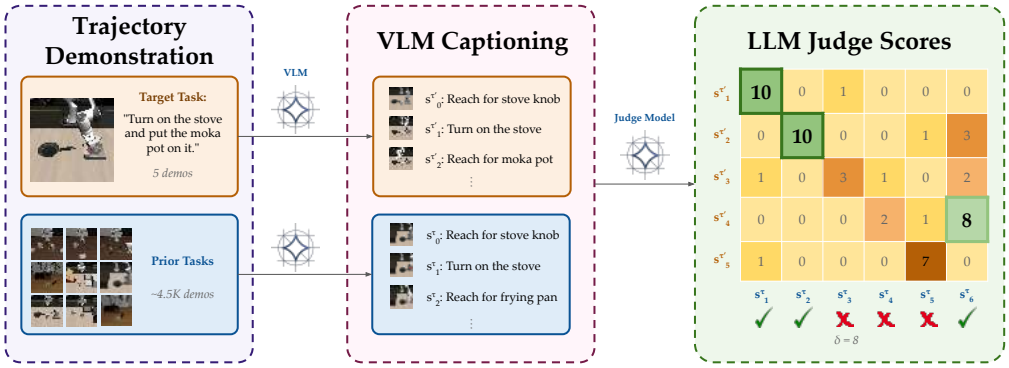


Figure 2: Overview of LIST pipeline. Left Block: target and prior datasets. Middle block: Segmenting and captioning by a VLM. Right Block: an LLM assigns a similarity score to each pair of target and prior segments, and we retrieve those that have a score above a determined threshold δ.

demonstrations in the prior set [12, 13, 14, 15, 16]. Unlike these methods, LIST operates directly in natural-language space, making retrieval criteria interpretable and easy to refine.

3 Method

Problem formulation. Assume a target task T with a dataset of N_T robotic demonstrations, $\mathcal{D}_T = \{\tau_1^T, \ldots, \tau_{N_T}^T\}$. Each trajectory τ is a collection of visual observations as well as actuator actions collected from an expert demonstration. Similarly, assume a prior dataset $\mathcal{D}_P = \{\tau_1^P, \ldots, \tau_{N_P}^P\}$ containing demonstrations collected from a diverse set of tasks. The goal of data retrieval is to identify relevant samples from $\mathcal{D}_P$ and use them to augment $\mathcal{D}_T$, such that training a policy on the combined demonstrations performs well on the target task T.

The Proposed Pipeline. LIST consists of three main components: (i) Identifying sub-trajectories and captioning them using a VLM, (ii) Measuring similarity in the text space using an LLM as a judge, and (iii) Retrieving relevant sub-trajectories and training the target policy. An overview of our pipeline is shown in Figure 2.

VLM-based segmentation and captioning. For each trajectory $\tau \in \mathcal{D}_T \cup \mathcal{D}_P$, we prompt a VLM to segment the demonstration into an ordered sequence of M_τ sub-trajectories, $(s_1^\tau, \ldots, s_{M_\tau}^\tau)$, each corresponding to a meaningful subtask. We allow the VLM to decide M_τ for every trajectory depending on the number of detectable subtasks. The VLM also generates a sequence of textual captions, $(c_1^\tau, \ldots, c_{M_\tau}^\tau)$, where c_m^τ describes the robot and the scene in sub-trajectory s_m^τ. We apply this procedure independently to every trajectory in both the target and prior datasets to create a textual representation used in the subsequent retrieval steps. Appendix A provides the VLM prompt.

Scoring similarity with an LLM. We use an LLM as a judge [17] to compare pairs of sub-trajectories under a predefined rubric r. For each trajectory $\tau \in \mathcal{D}_P$ and $\tau' \in \mathcal{D}_T$, we prompt an LLM to estimate a similarity score $\phi(c_n^\tau, c_m^{\tau'}; r)$ for every pair of sub-trajectory captions, where $n \in \{1, \ldots, M_\tau\}$ and $m \in \{1, \ldots, M_{\tau'}\}$. A higher score indicates that the prior sub-trajectory s_n^τ has greater similarity with the target $s_m^{\tau'}$ under the rubric r. The rubric is flexible in our framework and depending on the task requirements, it may consider the object interacted with by the end-effector, the motion pattern of the robot, the scene elements, or a combination of these factors. We evaluate ϕ for every pair of target and prior sub-trajectories generated by the VLM in the previous stage. Appendix B provides our rubric and detailed prompts.

Segment aggregation and retrieval. Given a retrieval threshold δ, we select each prior sub-trajectory whose similarity score is at least δ for at least one target sub-trajectory. In other words, a prior segment is selected if it matches at least one target segment with a similarity higher than δ. We then combine the retrieved segments with the target demonstrations and train a policy model using behavior cloning.

Table 1: Comparing the success rate (%) of different retrieval methods in the LIBERO environment. **Bold** indicates the best and underline shows the runner-up results. $\pm$ indicates the std computed across 5 seeds.

	Stove-Moka	Bowl-Cabinet	Moka-Moka	Soup-Cheese	Soup-Sauce	Cheese-Butter	Mug-Mug	Book-Caddy	**Average**
BC	69.4±4.2	75.0±4.4	25.6±5.0	31.6±7.5	25.4±3.6	17.4±4.7	29.4±2.8	57.4±5.0	41.4
FR	80.4±5.1	83.4±4.1	21.0±4.5	**61.0**±3.2	67.0±6.2	24.8±3.8	61.4±6.0	92.8±2.1	61.5
STRAP	71.4±4.2	**94.2**±1.5	20.8±1.5	45.6±8.0	50.0±5.6	**41.4**±7.4	52.4±2.4	59.4±4.9	54.4
IWR	66.3±4.1	92.8±0.9	**38.8**±4.6	55.0±11.3	58.3±13.3	36.0±9.3	68.6±6.1	89.0±3.1	63.1
LIST	**87.5**±4.4	88.4±2.6	35.8±3.3	51.4±9.3	**70.6**±6.4	17.2±9.9	**72.4**±5.3	**95.2**±1.7	**64.8**

4 Experiments

Setup. We use the LIBERO benchmark [6], a set of long-horizon, multi-stage robot manipulation tasks across multiple scenes. We use LIBERO-90 as the prior dataset and LIBERO-10 as the target set. For the target tasks, we assume only 5 demonstrations per target task for both data retrieval and policy learning. The prior dataset contains 90 tasks, each with 50 demonstrations.

We use Gemma-4-31B-it [18] as our VLM and Gemma-4-26B-A4B-it as our LLM. For each prior and target demonstration, the VLM receives both wrist and third-person camera views. We set the retrieval threshold to $\delta = 8$.

We compare `LIST` against naive BC that only uses the few target demonstrations, as well as retrieval methods FR, STRAP, and IWR, as introduced in Section 2. We use the language-conditioned BC-Transformer from robomimic [19] as the policy model for all methods. We repeat each experiment for 5 random seeds.

Results. Table 1 summarizes our results. While no method consistently outperforms others across all tasks, `LIST` outperforms the baselines in four of the eight evaluated tasks and on average, demonstrating the effectiveness of our proposed method.

Finally, Figure 3 provides an example of the intermediate outputs of `LIST`, including VLM-generated sub-trajectory descriptions and LLM similarity scores based on the motion pattern of the robot arm. These textual representations make the retrieval process interpretable: users can inspect how demonstrations are segmented and described, as well as which behavioral attributes guide similarity scoring. Moreover, users can modify the scoring rubric to emphasize task-relevant attributes, providing a direct way to steer retrieval through natural-language instructions.

5 Conclusion

We introduced `LIST`, which uses language-based sub-trajectory retrieval to improve few-shot imitation learning while making retrieval interpretable and controllable. We showed that `LIST` outperforms prior work in the LIBERO tasks. Future work should evaluate `LIST` on real robots and larger datasets and measure the benefits of human feedback. Finally, future work could extend `LIST` to an active learning approach that uses a planner model to automatically analyze failures in current policy trajectories and retrieve additional demonstrations that address them.

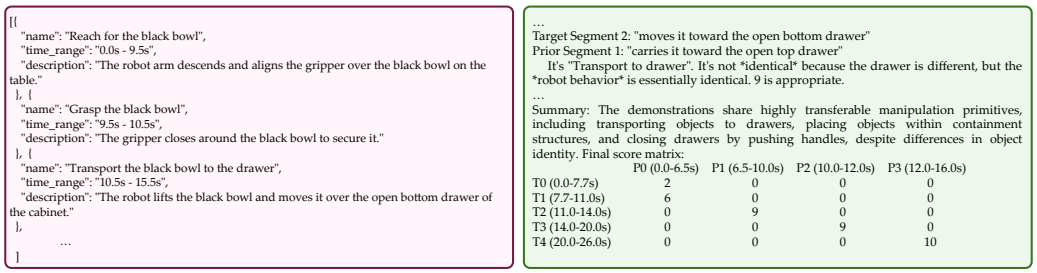


Figure 3: An example of (**Left**) VLM identifying sub-trajectories and their descriptions, and (**Right**) LLM scoring the similarity of sub-trajectories according to the motion pattern of the robot arm.

References

[1] M. Du, S. Nair, D. Sadigh, and C. Finn. Behavior retrieval: Few-shot imitation learning by querying unlabeled datasets. *arXiv preprint arXiv:2304.08742*, 2023.

[2] L.-H. Lin, Y. Cui, A. Xie, T. Hua, and D. Sadigh. Flowretrieval: Flow-guided data retrieval for few-shot imitation learning. *arXiv preprint arXiv:2408.16944*, 2024.

[3] M. Memmel, J. Berg, B. Chen, A. Gupta, and J. Francis. Strap: Robot sub-trajectory retrieval for augmented policy learning. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Learning Representations*, volume 2025, pages 63813–63832, 2025.

[4] D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2022.

[5] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 2668–2677, 10–15 Jul 2018.

[6] B. Liu, Y. Zhu, C. Gao, Y. Feng, qiang liu, Y. Zhu, and P. Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[7] S. Nasiriany, T. Gao, A. Mandlekar, and Y. Zhu. Learning and retrieval from prior data for skill-based imitation learning. In *6th Annual Conference on Robot Learning*, 2022.

[8] T. Giorgino. Computing and visualizing dynamic time warping alignments in r: the dtw package. *Journal of statistical Software*, 31:1–24, 2009.

[9] S. Kumar, S. Dass, G. Pavlakos, and R. Martín-Martín. Collage: Adaptive fusion-based retrieval for augmented policy learning. In *Proceedings of The 9th Conference on Robot Learning*, Proceedings of Machine Learning Research, 2025.

[10] A. Xie, R. Chand, D. Sadigh, and J. Hejna. Data retrieval with importance weights for few-shot imitation learning. In *9th Annual Conference on Robot Learning*, 2025.

[11] S. Dass, A. Khaddaj, L. Engstrom, A. Madry, A. Ilyas, and R. Martín-Martín. DataMIL: Selecting data for robot imitation learning with datamodels. In *Workshop on Making Sense of Data in Robotics: Composition, Curation, and Interpretability at Scale at CoRL 2025*, 2025.

[12] M. Hong, A. Liang, K. Kim, H. Rajaprakash, J. Thomason, E. Bıyık, and J. Zhang. Hand me the data: Fast robot adaptation via hand path retrieval. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

[13] Y. Zhu and F. Feng. Let me show you: Learning by retrieving from egocentric video for robotic manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

[14] N. Lin, T. Ohkawa, S. Chen, R. Wen, Z. Li, L. Huang, Z. Zhu, Y. Bao, Y. Li, M. Cai, X. Ma, W. Xu, and Y. Sato. Simdex: Mining similar egocentric videos for cross-embodiment dexterous manipulation, 2026.

[15] G. Papagiannis, N. Di Palo, P. Vitiello, and E. Johns. R+x: Retrieval and execution from everyday human videos. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[16] H. Qian, Y. Chen, Y. Xie, K. Ren, P. Chanrungmaneekul, G. Wang, B. Wen, C. Wei, and K. Hang. Robotok: An internet-scale data engine for human demonstration video retrieval and dexterous manipulation learning. *arXiv preprint arXiv:2609.03199*, 2026.

[17] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[18] G. Team. Gemma 4 technical report, 2026.

[19] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2021.

A VLM Prompt

We use the same prompt for captioning all prior and target demonstrations. It includes the task name and a one-shot example with a short video, its ground-truth segmentation, and captions. This keeps the segment granularity and output format consistent. The prompt allows segment boundaries to overlap and does not require strict start and end times. We also allow the model to generate thinking tokens. It outputs a JSON file with the start and end time of each segment and a description of the robot's behavior. Here is the most important notes of the caption generation prompt:

Segmentation and captioning prompt

```
You are analyzing a robotic manipulation demonstration.

You are given TWO videos of the SAME demonstration:
1. A third-person view of the whole workspace, showing the robot arm, the table
and the surrounding objects.
2. A wrist-mounted (in-hand) camera view, recorded from the robot's gripper and
looking out at whatever the gripper is pointed at.
...
The demonstrated task is:
"{TASK_DESCRIPTION}"

Your goal is to decompose this demonstration into meaningful manipulation
subtasks and estimate their approximate temporal intervals.

Use the task description above as the high-level goal that guides how you
segment the demonstration. The subtasks you produce should, taken in order,
accomplish this task. However, the task description is only a guide: every
subtask, boundary, object, and motion you report must still be grounded in what
is actually visible in the videos.
...
Object and scene interpretation:

A meaningful subtask is a coherent manipulation phase, such as but not limited
to:
reaching, contacting, grasping, lifting, transporting, inserting, placing,
releasing, pushing or pulling, opening or closing, repositioning, rotating,
sliding, pressing, aligning

Handling corrections, failures, and recovery:
Do not ignore visible corrective or recovery behavior.
If a correction occurs inside the same manipulation phase, merge it into the
parent subtask.
Do not mention the existence of corrections or failures in the description.

Boundary rules:
Use the earliest visible frame/time where the subtask is underway as the start.
Use the latest visible frame/time before the next distinct subtask is underway
as the end.

For each subtask, provide:
"name": short subtask name
"start_time" and "end_time" as approximate seconds
"description": concise visual description
...
```

B Rubric and LLM Prompt

For similarity score estimation, we give the model the generated descriptions of all segments from one prior and one target demonstration. It outputs a similarity score matrix for all their segment pairs at once. As in the previous stage, the model first generates thinking tokens and then outputs the matrix. Inspecting these tokens enables a human expert to identify scoring errors and refine the rubrics when needed. Here are the rubrics and the important parts of the prompt used for estimating the similarity score:

Segment transferability judge prompt

```
You are an expert in robotic manipulation, motion understanding, and robot task
transfer.

Your goal is to evaluate how transferable the robot motion and manipulation
behavior are between segments of two robotic demonstrations.
...
Focus primarily on motion similarity and manipulation transferability rather
than wording overlap or object identity.
...
When deciding similarity, look past the words and reason about:
* The PHYSICAL TYPE of the manipulated object and how it moves.
* The DIRECTION and geometry of the motion (left/right/up/down, toward/away, the
path the object travels) as well as the ORIENTATION and GRASPING state of the
robot end-effector during the sub-task.
...

Evaluate EVERY pair independently.

Do NOT force:
* one-to-one matching,
* temporal alignment,
* monotonic correspondence.

Multiple segments may receive high scores.

One segment may be relevant to multiple segments.
...

================================================
SCORING RUBRIC
================================================

For EACH target-prior segment pair assign ONE integer score from 0 to 10.

Interpret scores as TRANSFERABILITY of robot manipulation behavior.

0 = completely unrelated motion
No meaningful transferability in manipulation behavior.

1 = almost no similarity
Very weak overlap in generic movement only.

2 = minimal similarity
Small overlap in broad motion pattern (e.g., both involve approaching or generic
arm movement).

3 = weak similarity
Limited transferable behavior. Some overlap in interaction stage or motion
structure.

4 = low-moderate similarity
Partial overlap in manipulation dynamics, but transferability is limited.
```

```
5 = moderate similarity
Noticeable overlap in motion pattern, interaction type, or manipulation behavior.

6 = moderately strong similarity
Reusable aspects of the manipulation exist, though with important differences.

7 = strong similarity
Clearly reusable manipulation behavior. High overlap in manipulation primitive
and interaction mechanics.

8 = very strong similarity
Highly transferable robot behavior. Motion structure and functional interaction
are strongly aligned.

9 = near-direct transfer
Very high similarity in manipulation primitive, motion sequence, and interaction
behavior.

10 = essentially direct transfer
Almost identical reusable manipulation behavior.

...
Different objects may still receive high scores if the robot motion,
end-effector orientation, placement relation, and interaction mechanics transfer
well.
...
```