TRAIL Team Description Paper for RoboCup@Home 2024

Soshi Tsunashima, Tomotaka Hoshina, Ayaha Motomochi, Kosei Kubo, Koki Muramoto, Hiroki Watanabe, Yuki Izumi, Ryo Tejima, Yuya Ikeda, Tatsuya Matsushima, Yutaka Matsuo, Yusuke Iwasawa,
The University of Tokyo
Website: https://trail.t.u-tokyo.ac.jp/project/robocup2024/
Qualfication video: https://youtu.be/blGN0qe5qm8

November 27, 2023

Abstract. This paper provides an overview of the main developments and activities of team TRAIL. Our team, TRAIL, consists of AI/ML laboratory members from The University of Tokyo. We leverage our extensive research experience in state-of-the-art machine learning to build general-purpose in-home service robots. We previously participated in many competitions using Human Support Robot (HSR): RoboCup@Home Japan Open 2022 and 2023 (DSPL), and RoboCup@Home 2023. Throughout the competitions, we showed that a data-driven approach and foundation model-based system are effective for performing in-home tasks. In addition, to stimulate research all over the RoboCup@Home community, we build a platform that manages data collected from each site belonging to the community around the world, taking advantage of the characteristics of the community.

1 Introduction and Relevance

1.1 TRAIL

Our team, TRAIL (Tokyo Robotics and AI Lab), was launched in 2020 as a project and was organized in 2021 as a group under Matsuo Laboratory[1] at The University of Tokyo. Our laboratory mainly engages in fundamental research on deep learning, especially world models, and also focuses on robotics as an application of deep learning. TRAIL aims to realize general-purpose robots that can perform various tasks in diverse environments by utilizing robot learning.

[1] https://weblab.t.u-tokyo.ac.jp/en/

Table 1. Results of Competitions

Competitions	Results
RoboCup@Home Japan Open 2020	*2nd place* at DSPL league
	1st place at Technical Challenge
WRS2020, equivalent to RoboCup (worldwide)	*2nd place* at Partner Robot Challenge, equivanlent to DSPL league
RoboCup@Home Japan Open 2022	*3rd place* at DSPL league
RoboCup@Home Japan Open 2023	*1st place* at DSPL league
RoboCup@Home 2023	*3rd place* at DSPL league

As part of these activities, we are making the most of our AI/ML lab knowledge to conduct experiments to build a real robot system that can carry out daily life support tasks in household environments. More specifically, we aim to build a well-generalized and fast-adaptable system leveraging various *foundation models*.

1.2 The Experience and Achievements in Local Tournaments

So far, we have participated in the competitions listed in Table 1 as opportunities to validate our in-home service robot system and won awards as indicated. In the competitions, we leveraged a data-driven approach to address tidy-up tasks in household environments rather than a pre-programmed approach to handle numerous edge cases. Through the competitions, we showed that a data-driven approach adapts better to diverse household environments and flexibly handles various edge cases than a pre-programmed approach; refer to [2] for details. In RoboCup@Home Japan Open 2023 and RoboCup@Home 2023, we also showed that the foundation model-based system is effective.

2 Approach

2.1 System Overview

Figure 1 shows the overview of our foundation model-centric system. The foundation models which consist of our system are Whisper [6], GPT-4 [4], Detic [11], CLIP [5], and CLIP-Field [7] (a model that consists of an integration of foundation models), and they have the ability to enhance the system to be generalized and adaptive with prompting.

The system has two aspects: getting environmental information and making a plan to execute tasks. As for getting environmental information, recognize the objects by using Detic and CLIP at first, and then, based on the recognition of the objects, make a semantic map with CLIP-Fields. As for making a plan to execute tasks, recognize the command that humans give by using Silero VAD and Whisper at first, and then plan with GPT-4 and make an LLM Plan. Finally, the system integrates the collected environmental information and the LLM Plan, makes an executable plan, and executes it. Information on the details of the entire system can be obtained from our previous paper [8].

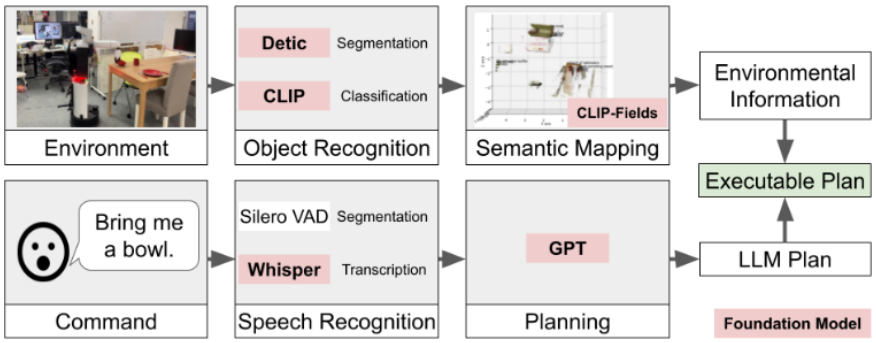

Fig. 1. Overview of our foundation-model-based system. The foundation models collaborate to process the environment and a natural language command into an executable plan.

2.2 Foundation Model

Foundation model is a large-scale model that is pre-trained with a large amount and a variety of data. By engineering the prompts, which are given in natural language to specify the task, it can handle various downstream tasks and edge cases even without additional learning.

2.2.1 Foundation Model for Speech Recognition

We leverage Whisper, a foundation model for speech recognition. The detailed procedure is as follows. First, we leverage the DNN model Silero VAD [9] to segment speech intervals where the probability of spoken English is above a certain threshold. Next, only those segments are input to Whisper for transcription. This approach aims to achieve robust recognition in the presence of noise. In addition, Whisper utilizes prompts, such as conversational speech, to understand context and perform transcription accordingly. Therefore, by providing Whisper with information on GPSR task settings and an explanation of the execution environment as a prompt, it can not only recognize simple phrases but also perform transcription that considers the relationship between the speech and the GPSR task. Specifically, we were able to remove noise (out-of-context words) during the transcription phase, improve the recognition accuracy of task-specific hard-to-hear words (e.g., bamboo shoot), and achieve more robust recognition with respect to accents.

2.2.2 Foundation Model for Task Planning

We leverage GPT-4 for task planning. To execute various commands, we prepare 23 skill functions (e.g., navigation function, pick function) corresponding

one-to-one to 23 different actions (e.g., navigation, pick). Each skill function executes the corresponding action by specifying one to three arguments. In task planning, we leverage GPT-4 to select, sequence, and specify arguments of skill functions (hereinafter, this process is referred to as "planning") so that commands can be accomplished by combining skill functions. The final output of planning is the LLM plan.

Planning is conducted in two stages, using the Chain-of-Thought [10,1] method with multi-stage prompts. In the first stage, the command given is used as input, and a rough plan with the thought process is output. We call this plan the intermediate plan. In the second stage, the output of the first stage, the intermediate plan, is used as input and the LLM plan, a combination of skill functions with specified arguments, is output. In this second step, the argument is specified and the skill function is called using function calling, which is a function that calls a specific function according to the input character string. By going through the intermediate planning, it is possible to plan in consideration of the actual order of operation, compared to direct planning with one step. This method is especially effective when the word order of the command and the order of the skill functions corresponding to the words are different (e.g., Tell me how many fruits are on the dining table.).

2.2.3 Foundation Model for Object Recognition

The object recognition module consists of two modules: the object detection module and the object classification module. We leverage a foundation model for each module: Detic for the object detection module and CLIP for the object classification module.

The object detection module performs three filtering operations on the detection results to remove objects that are only partially seen, too small, or included and outputs a segmented image. We also take advantage of the feature of Detic that arbitrary vocabulary (prompts) can be specified as classes, whereas conventional learned models have fixed classes. Conventional DNN-based modeling methods require training to detect unknown objects, and may also incorrectly detect objects that are not the target of detection. However, by using the prompt tuning technique to adjust the prompts to the objects placed in the room, it is possible to detect unknown objects without training and also to exclude objects that are not the target of detection. Specifically, a list of prompt-tuned prompts that enable only detecting all objects of interest is adapted to Detic. The object classification module uses CLIP to classify segmented images from the object detection module. Training is performed for unknown objects that are difficult to classify by prompt tuning alone. It collects 100-500 images for each object in 30-60 seconds, performs data augmentation, and tunes only the fully connected layer following the trained CLIP model.

2.2.4 Foundation Model for Integration of Environmental Information

To execute a command, it is necessary to associate an object with its location information. Our system uses CLIP-Fields to learn the correspondence between objects and their positions from environmental data (RGB-D images). Here, CLIP-Fields combines the foundation model CLIP and neural field representation which maintains a semantic representation in space.

First, we move the robot around the room using a controller to collect RGB-D images to train CLIP-Fields. Then, data is collected during task execution, CLIP-Fields is retrained, and environmental information is updated to respond to environmental changes.

3 Contribution

Re-usability of the System for Other Research Groups

The source codes we developed, including simulators and educational content, are available on GitHub (`https://github.com/matsuolab`). Recently, we collaborated with Google DeepMind and various research institutions to establish a dataset of real-world robot learning and contributed to developing a cross-robot manipulation policy [3]. The open-source dataset and model checkpoint would stimulate the community to leverage flexible learned policy for realizing general-purpose service robots.

We also aim to build a platform that integrates and manages data collected from each site belonging to the RoboCup@Home community through real robots and simulators, utilizing our experiences as the AI/ML lab. The unique characteristics of the RoboCup@Home community are desirable for realizing this platform, for the same robot (HSR) and the similar environment (RoboCup@Home) are located at different locations all over the world, which is one of the reasons why we are eager to take part in DSPL league. The accumulated data are accessible and usable from each site. Figure 2 shows its conceptual diagram. It will be an open-source platform, and we believe this scheme will greatly accelerate research across the whole community.

Contribution to Expand the Community

We have been sharing our knowledge and findings widely inside and outside The University of Tokyo, regardless of age or affiliation. The aim is to encourage people who do not major in robotics, such as those who specialize in machine learning and data science (we are from the AI/ML lab), to enter robotics by providing them with opportunities, which we believe will lead to the expansion of the community in the long run. Indeed, our activities motivated some members to participate in the RoboCup@Home competition.

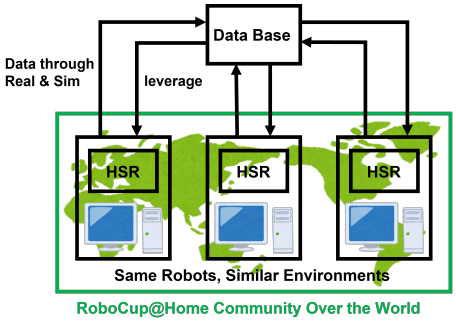

Fig. 2. Conceptual diagram of the platform. Taking advantage of the characteristics of the RoboCup@Home Community, we aim to build a system to collect and utilize data for robots.

Table 2. Part of our educational content open to the public

Educational Contents	Started	Remarks
Data Science Basic Courses	2014	more than 5000 participants
Deep Learning Basic Courses	2015	more than 6000 participants
World Models Seminar	2021	more than 300 participants
Robot System Tutorial [3]	2022	Published on the website

Release of the Knowledge Widely

For educational purposes, we open our activities, research achievements, and findings to the public (including junior high and high school students, other universities, and adults). Some of the contents [2] are shown in Table 2.

Research Projects at The University of Tokyo

We have been offering a part of our activities as project-based programs in the Faculty of Engineering at The University of Tokyo since 2021, as opportunities to operate real robots offline besides the online tutorial described in Table 2.

References

1. T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.
2. T. Matsushima, Y. Noguchi, J. Arima, T. Aoki, Y. Okita, Y. Ikeda, K. Ishimoto, S. Taniguchi, Y. Yamashita, S. Seto, S. S. Gu, Y. Iwasawa, and Y. Matsuo. World robot challenge 2020 – partner robot: a data-driven approach for room tidying with mobile manipulator. *Advanced Robotics*, 36(17-18):850–869, 2022.
3. Open X-Embodiment Collaboration, A. Padalkar, A. Pooley, A. Jain, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Singh, A. Brohan, A. Raffin,

[2] https://deeplearning.jp/
[3] https://matsuolab.github.io/roomba_hack_course/course/

A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Ichter, C. Lu, C. Xu, C. Finn, C. Xu, C. Chi, C. Huang, C. Chan, C. Pan, C. Fu, C. Devin, D. Driess, D. Pathak, D. Shah, D. Büchler, D. Kalashnikov, D. Sadigh, E. Johns, F. Ceola, F. Xia, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Schiavi, H. Su, H.-S. Fang, H. Shi, H. B. Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Kim, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Wu, J. Luo, J. Gu, J. Tan, J. Oh, J. Malik, J. Tompson, J. Yang, J. J. Lim, J. Silvério, J. Han, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Zhang, K. Majd, K. Rana, K. Srinivasan, L. Y. Chen, L. Pinto, L. Tan, L. Ott, L. Lee, M. Tomizuka, M. Du, M. Ahn, M. Zhang, M. Ding, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, P. R. Sanketi, P. Wohlhart, P. Xu, P. Sermanet, P. Sundaresan, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Moore, S. Bahl, S. Dass, S. Song, S. Xu, S. Haldar, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Dasari, S. Belkhale, T. Osa, T. Harada, T. Matsushima, T. Xiao, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, V. Jain, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Li, Y. Lu, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Xu, Y. Wang, Y. Bisk, Y. Cho, Y. Lee, Y. Cui, Y. hua Wu, Y. Tang, Y. Zhu, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Xu, and Z. J. Cui. Open X-Embodiment: Robotic learning datasets and RT-X models. *arXiv preprint arXiv:2310.08864*, 2023.

4. OpenAI. GPT-4 Technical Report. Technical report, 2023.
5. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
6. A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023.
7. N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam. CLIP-Fields: Weakly Supervised Semantic Fields for Robotic Memory. In *Robotics: Science and Systems*, 2023.
8. M. Shirasaka, T. Matsushima, S. Tsunashima, Y. Ikeda, A. Horo, S. Ikoma, C. Tsuji, H. Wada, T. Omija, D. Komukai, et al. Self-Recovery Prompting: Promptable General Purpose Service Robot System with Foundation Models and Self-Recovery. *arXiv preprint arXiv:2309.14425*, 2023.
9. S. Team. Silero VAD: pre-trained enterprise-grade Voice Activity Detector (VAD), Number Detector and Language Classifier. https://github.com/snakers4/silero-vad, 2021.
10. J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.
11. X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra. Detecting Twenty-thousand Classes using Image-level Supervision. In *ECCV*, 2022.

Software and External Devices

We use the *Toyota HSR*. No modifications have been applied.

Fig. 3. Our HSR

Robot's Software Description

For our robot we are using the following software:

- Object Detection: Mask R-CNN, UOIS
- Object Classification: CLIP (ViT-B/32)
- Speech Recognition: Whisper
- NLP and NLU: GPT-3
- Grasping: Multi-Object Multi-Grasp, GraspNet, GGCNN
- Simulators: Gazebo, PyBullet, Isaac Sim

External Devices

Our robot relies on the following external hardware:

- msi GS66 STEALTH with 10th Gen. Intel® Core™ i9 processor and NVIDIA® GeForce RTX™ 3080 Laptop GPU 16GB GDDR6 mounted on back of HSR
- Desktop PC with AMD Ryzen Threadripper 3970X 32-Core Processor CPU and 2 NVIDIA RTX 3090 GPU connected via wireless network