SKILL COMPOSITION

Vaishnava Hari

Dream A marketplace of skills, from which a robot installs skills. Skill is the ability to perform part of a task. This task could be new for the robot and set in an unseen environment. A task could be accomplished by utilizing a set of skills.

1 Breaking down the problem

1.1 Decomposition of task

First challenge is to decompose a task into smaller sub-tasks, which then can be executed by a subset of skills. Or in other words, given a set of skills, how to select a subset of skills to execute a task during inference. This is a well studied problem in the field of Hierarchical Reinforcement Learning (HRL) [1]. In HRL, a high level policy selects a low level policy from a set of low level policies, this is also called as *options* [2]. In other words, the high level policy is a policy over options and the low level policies are a policy over primitive actions. There could be multiple levels of such hirarchy, where middle level policies are also policy over options, but this is beyond the scope of this work. Each *option* (or a low level policy) is temporally extended and has its own termination condition. Policy gradients methods have been extended to this case [3].

1.2 Subtask discovery

Next challenge is to identify which skills to learn during training. This is usually done by,

- **Bottom-up approach:** Handpicking a set of primitive skills, intuitively, and training them individually. Later, they are frozen and put together for training the high-level policy. This method is widely used [4].

However, this approach can not be generalized. And also handcrafting the set of skills is not optimal [5]. This can be overcome by using the following approach:

- **Top-down approach:** Train both high-level and low-level policies together. This introduces the following hurdles: non stationary transistion function for the high-level policy and effective exploration by options.

1.3 Non-stationary transition function

This problem occurs because the state transistion for the action (ie selecting a low-level policy) taken by high-level policy is dependent on the low-level policy, which in itself is still learning.

This can be can be tackled by learning high-level policy independent of option's policy gradient. Manager-Worker architecture [6] adresses this approach.

1.4 Effective exploration of subtask

Beacuse the options (or low-level policies) is trained along with the high-level policy, it is difficult to ensure that all the options have explored their subtask space well. Or in other words, for a option to try a new set of actions, it needs to part of the high-level policy task, which may always not be the case. This is accomplished in data-efficient way by using a suitable model of the environment. But, learning such a model from experience is still an open problem.

1.5 Distillation

There have been recent works on utilizing Vision-Language-Action (VLA) models to train expert RL policy [7]. Also Model-based RL would address the former problem. Combining these two approaches, it could be a good idea to utilize a VLA model to act as both reward model and a teacher, effectively distilling the knowledge of the VLA model into a RL agent for that task.

2 Next steps

The scope of this project is further simplified with the assumptions made for the following phases:

2.1 Phase 1: Distillation of a single skill

Choose a suitable application task and distill a policy from a pre-trained VLA model.

- **Scene**: Wheeled robot
- **VLA model**: SAM2ACT [8].

3 Previous Works

3.1 Vision-Language-Action

Vision-Language-Action (VLA) models are a class of models based on the transformer architecture and are designed to process and understand both visual and textual information. They are made up of two parts:

1. VLM (Vision-Language Model): This part of the model is responsible for understanding and processing visual and textual information.
2. Action Head: Made of diffusion transformers and are responsible for generating actions using the embeddings from the VLM. They can also take in additional inputs such as proprioceptive and previleged environment information.

Some of the popular VLA models include: - SoFar [9], state of the art in robot manipulation tasks in the Google SimplerEnv dataset [10].

- OpenVLA [11], a 7B parameter model. It uses Lama 2 as the backbone and is trained on 970k robot demos in Open X-Embodiment dataset [12].

- SAM2ACT [8] is a new model focused on manipulation tasks. It is the leader in the RLbench dataset [13].

References

[1] M. Hutsebaut-Buysse, K. Mets, and S. Latré, "Hierarchical Reinforcement Learning: A Survey and Open Research Challenges," *Machine Learning and Knowledge Extraction*, vol. 4, no. 1, pp. 172–221, Mar. 2022.

[2] R. S. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, vol. 112, no. 1-2, pp. 181–211, Aug. 1999.

[3] P.-L. Bacon, J. Harb, and D. Precup, "The Option-Critic Architecture," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, Feb. 2017.

[4] S. Pateria, B. Subagdja, A.-h. Tan, and C. Quek, "Hierarchical Reinforcement Learning: A Comprehensive Survey," *ACM Comput. Surv.*, vol. 54, no. 5, pp. 109:1–109:35, Jun. 2021.

[5] D. Silver and R. S. Sutton, "Welcome to the Era of Experience."

[6] A. S. Vezhnevets, S. Osindero, T. Schaul, N. Heess, M. Jaderberg, D. Silver, and K. Kavukcuoglu, "FeUdal Networks for Hierarchical Reinforcement Learning," in *Proceedings of the 34th International Conference on Machine Learning*. PMLR, Jul. 2017, pp. 3540–3549.

[7] T.-Y. Xiang, A.-Q. Jin, X.-H. Zhou, M.-J. Gui, X.-L. Xie, S.-Q. Liu, S.-Y. Wang, S.-B. Duang, S.-C. Wang, Z. Lei, and Z.-G. Hou, "VLA Model-Expert Collaboration for Bi-directional Manipulation Learning," Mar. 2025.

[8] H. Fang, M. Grotz, W. Pumacay, Y. R. Wang, D. Fox, R. Krishna, and J. Duan, "SAM2Act: Integrating Visual Foundation Model with A Memory Architecture for Robotic Manipulation," Feb. 2025.

[9] Z. Qi, W. Zhang, Y. Ding, R. Dong, X. Yu, J. Li, L. Xu, B. Li, X. He, G. Fan, J. Zhang, J. He, J. Gu, X. Jin, K. Ma, Z. Zhang, H. Wang, and L. Yi, "SoFar: Language-Grounded Orientation Bridges Spatial Reasoning and Object Manipulation," Feb. 2025.

[10] X. Li, K. Hsu, J. Gu, K. Pertsch, O. Mees, H. R. Walke, C. Fu, I. Lunawat, I. Sieh, S. Kirmani, S. Levine, J. Wu, C. Finn, H. Su, Q. Vuong, and T. Xiao, "Evaluating Real-World Robot Manipulation Policies in Simulation," May 2024.

[11] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "OpenVLA: An Open-Source Vision-Language-Action Model," Sep. 2024.

[12] O. X.-E. Collaboration, A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Kolobov, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. V. Frujeri, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Yang, G. Wang, H. Su, H.-S. Fang, H. Shi, H. Bao, H. B. Amor, H. I. Christensen, H. Furuta, H. Bharadhwaj, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Vakil, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh, K.-H. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Castro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. Liu, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi, P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Mart'in-Mart'in, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. Sonawani, S. Tulsiani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Kumar, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, X. Liangwei, X. Li, Y. Pang, Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Dou, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y.-H. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, Z. Fu, and Z. Lin, "Open X-Embodiment: Robotic Learning Datasets and RT-X Models," Jun. 2024.

[13] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "RLBench: The Robot Learning Benchmark & Learning Environment," Sep. 2019.