arXiv:2302.00763v1 [cs.LG] 1 Feb 2023

与语言模型协作进行具身推理

Ishita Dasgupta* DeepMind
Christine Kaeser-Chen DeepMind
Kenneth Marino DeepMind
Arun Ahuja DeepMind
Sheila Babayan DeepMind
Felix Hill DeepMind
Rob Fergus DeepMind

摘要

在复杂且模糊的环境中进行推理，是强化学习智能体的一项关键目标。尽管一些复杂的强化学习智能体能够成功完成困难任务，但它们需要大量训练数据，而且往往难以泛化到未曾见过的新环境和新任务。另一方面，大规模语言模型（LSLM）展现出了强大的推理能力，并能够通过上下文学习适应新任务。然而，LSLM本身并不具备探查或干预环境的能力。在这项工作中，我们研究如何在一个由三个部分组成的单一系统中结合这些互补能力：规划器、执行器和报告器。规划器是一个预训练语言模型，能够向一个简单的具身智能体（执行器）发出命令，而报告器则与规划器沟通，为其下一条命令提供信息。我们提出了一组需要推理的任务，测试该系统零样本泛化的能力并探究失败情况，同时展示如何通过强化学习训练系统中的各个组件以提升性能。

1 引言。

在具身环境中完成复杂任务通常需要逻辑推理。长期以来，逻辑推理一直是机器学习面临的一项挑战 [Russin 等人，2020年，Mitchell， 2021] ——对于具身智能体来说更是如此，因为除了要推理如何完成复杂任务之外，智能体还必须在环境中感知 并控制 。然而，近期的大规模语言模型（LSLM）在推理方面展现出巨大潜力 [Radford 等人，2019年，Brown 等人， 2020]。这种复杂的推理能力能否用于具身任务?

一个主要问题是，LSLM 并非具身的，也未接地。它们无法直接在具身环境中采取行动，也无法了解环境中正在发生什么。对于这些问题，我们依赖智能体模型的其他组件。在这项工作中，我们研究了一种智能体范式，并将其称为**规划器-执行器-报告器**。**规划器** 就是 LSLM——它读取任务描述，进行所需的逻辑推理，并将问题分解为一系列简单指令。这些指令会传递给**执行器**，它是一个经编程后可在环境中完成一小组简单指令的强化学习智能体。最后，为了完成反馈回路，我们设置了**报告器**，它会观察环境并将信息反馈给规划器，以便规划器调整发出的指令。参见图 1A。

近期的其他研究探讨了在具身推理任务中为 LSLM 提供闭环反馈的形式 Huang 等人 [2022], Ahn 等人 [2022]。在这项工作中，我们将这些方法推广为由三个部分组成的规划器-执行器-报告器范式。我们通过引入一系列任务并对其进行评估，凸显了这些组件各自发挥的关键作用：

*通讯作者；idg@google.com

第36届神经信息处理系统大会（NeurIPS 2022），LaReL 研讨会。

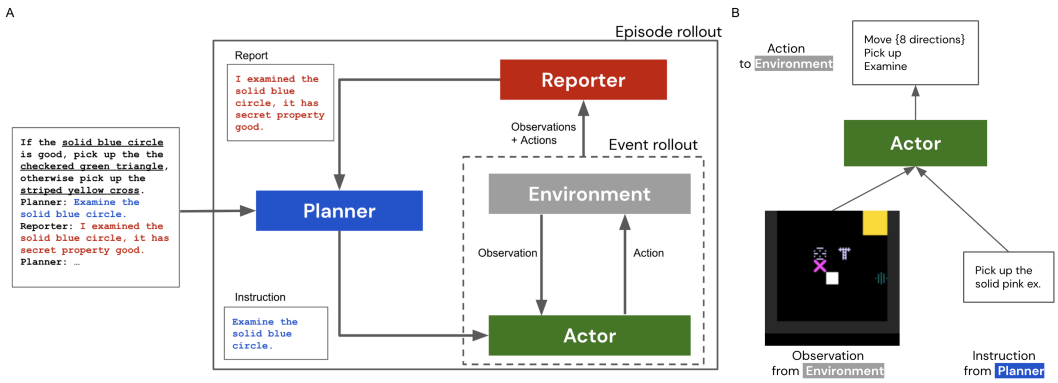

图 1：**设置**。 A. **规划器-执行器-报告器** 范式示意图及其交互示例。B. PycoLab 环境的观察和动作空间。

这些任务要求智能体探索世界以收集规划所需的信息、将复杂任务拆解为若干步骤，并将世界的视觉属性传达给规划器。最后，我们证明报告器模块可以通过强化学习（RL）进行训练，从而减少对人工指定反馈来源的需求。

2 方法

环境与执行器： 我们的环境是一个二维部分可观测网格世界。环境中包含由颜色、形状和纹理指定的独特物体，执行器看到的是自上而下的自我中心像素 RGB 视图，其可视范围为智能体周围 5 格。除了移动动作外，执行器站在物体上时还可以执行两种特殊动作：检查，用于揭示有关该物体的一段隐藏文本；以及拾取，用于将该物体加入物品栏。

执行器经过强化学习预训练，以遵循“拾取 X”或“检查 X”形式的指令。图 1B 展示了一个来自环境的观察示例；有关执行者架构和环境的详细信息，请参见附录 B。

规划器： 我们使用架构相同的预训练大型语言模型：Chinchilla [Hoff- mann 等人， 2022], , 参数规模为 7B 和 70B。为促进上下文学习中的落地 [Brown 等人， 2020], , 我们为每项任务提供 5 个随机选取的“少样本示例”（假设规划器、报告器和执行器均为最优；全文见附录 E），并直接将模型采样生成的语言用作执行器的输入。在每个时间步，采样生成的语言和报告器生成的信息都会附加到对话记录中，并作为提示输入，以便规划器在下一个时间步生成新的指令。

报告器： 我们进一步通过将报告器与分层强化学习[Sutton 等，1999，Kulkarni 等， 2016], 进行类比，来明确其作用：高层的“规划器”向低层的“执行器”发出时间抽象指令。这类设置与我们设置的一个关键区别在于，在我们的实验中，执行器和规划器的观测空间不同。在我们的设置中，执行器处理像素观测并生成移动动作，而规划器处理语言观测（提示）并生成语言动作（即发出的指令）。执行器以语言为条件，能够理解规划器的指令。但规划器无法解析执行器动作的结果（从而生成恰当的下一步动作）。报告器将执行器的动作+观测空间转换为规划器的观测空间。在最一般的情况下，报告器接收（一个序列的）执行器动作和像素观测，并生成一个文本字符串，其中包含真实 且相关 的信息，说明执行器做了什么以及环境如何响应。

实现报告器有多种方式，具体取决于报告的内容，以及其中有多少是硬编码的、预训练的或从头学习的。以往的工作采用了隐式报告器，由执行器内部实现，仅传达指令完成情况 [Ahn 等人， 2022], ；或采用预训练感知模型，根据执行器的观察回答自然语言问题[Zeng 等，2022年，Huang 等人， 2022]。在本研究中，我们首先使用硬编码的报告器，探索规划器-执行器交互在我们新颖的信息收集任务中的表现（第 3 节）。随后，我们率先在规划器-执行器-学习器循环中学习该报告器，以优化奖励（第 4 节）。

任务：我们构建了一组任务，用于检查具身环境中语言模型规划器能够助力应对的推理、泛化和探索挑战（详见附录 C）。我们聚焦于两类任务（条件任务和搜索任务），它们需要显式的信息收集，具体而言：a）规划器必须发出明确的信息收集指令；b）执行器必须执行该指令；c）报告器必须传达结果；之后，d）规划器才能发出下一条指令。

3 将语言模型用作交互式规划器

我们检查在需要规划器、执行器和报告器这三个组件协同才能成功完成的任务中，它们之间的交互。在先前工作的基础上 [Ahn 等，2022，Zeng 等人 2022] 表明，LSLM 能够将复杂的现实世界任务分解为一步步的指令；我们聚焦于这样的任务：规划器不仅需要明确发出信息收集指令，还要纳入报告的信息，以生成下一步指令。此外，我们的任务涉及具有抽象属性的物体，而这些属性无法接地于语言模型先前的语义经验，因此需要进行大量抽象逻辑推理。我们分析不同规划器的表现及其鲁棒性。所有组件均经过预训练。

任务设置如下：房间里的所有物体都有一种“秘密属性”（好／坏／未知）。当执行器“检查”某个物体时，真实值报告器会向规划器传递一个文本字符串：“我检查了{object}，它的秘密属性是{value}”。随后，规划器可以向执行器发出下一条指令。

3.1 秘密属性条件任务

我们从最简单的信息收集任务开始。回合的目标是根据另一个物体的秘密属性，拾取正确的物体。传递给规划器的任务描述如下：“如果 {decider object} 是好的，就拾取 {object 1}；否则拾取 {object 2}”。成功完成一个回合需要 5 个步骤：a) 规划器指示执行器检查 {decider object}；b) 执行器检查该物体；c) 报告器传达揭示的信息（在这种设定下总能正确完成）；d) 规划器根据报告判断应该拾取哪个物体（{object 1} 还是 {object 2}），并指示执行器拾取正确的物体；e) 执行器拾取正确的物体。

显式的信息收集动作通常是纯强化学习面临的难题。使用 LSLM 规划器，并以 5 条已解决示例的语言轨迹作为提示，再加上仅针对简单的拾取和检查任务训练的执行器，我们就能以较高准确率完成这一复杂的多步任务（图 2A）。即使经过 1 亿学习器帧，纯强化学习基线的表现仍然很差（见附录 D，图 2A）。

在我们的分析中，我们确定了两种主要的失败情况：LSLM 规划器无法根据环境反馈推断下一条指令，以及执行器未能遵循规划器提供的指令。在第一种情况下，我们发现，在给定所有信息的情况下，更小的语言模型（7B 参数）仅有 58% 的概率能够推断出应拾取哪个物体以获得奖励；更大的语言模型（70B 参数）则有 96% 的概率能够做到这一点。这表明，若不进行微调，即使是相对简单的推理任务，对更小的模型而言仍然难以胜任。在第二种失败情况下，我们发现执行器可能会遇到分布偏移，例如回合长度或指令格式发生变化，导致其无法遵循规划器的指令。

3.2 秘密属性搜索任务

我们通过要求增加信息收集步骤，扩展了先前的任务。智能体不再只需检查单个物体，而是需要检查多个物体，记录它们的秘密属性，并拾取正确的物体以获得奖励。该任务的描述为：“这些物体是 {}, {}, {}, 和 {}。拾取具有良好秘密属性的物体”。一次成功的回合包括：规划器要求执行器依次检查每个物体，直到找到具有“良好”属性的物体，此时规划器会要求执行器拾取该物体。

尽管此任务需要更多的信息收集步骤，且强化学习基线表现较差（见附录 D），但采用规划器–执行器–报告器的智能体框架仍能以零样本方式完成任务（即不与环境进行任何额外交互；图 2A）。有趣的是，我们发现智能体在此任务中的表现优于前一项任务；前一项任务只需检查一个物体（图 2A；另见附录 D）。我们推测，由于信息收集步骤的数量会发生变化，规划器不会采用僵化的“一次检查，一次拾取”策略，因此能够更好地应对错误。例如，执行器可能检查了错误的物体。我们发现规划器确实能够从此类错误中恢复（第 A.1 节）。与上述观察结果类似，我们注意到，更大的语言模型（70B）表现显著优于更小的模型（7B）（图 2A）。

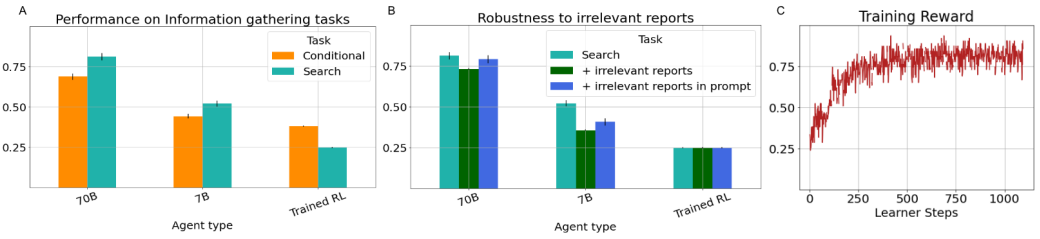

图 2：**结果**。A. 不同规划器和基线强化学习在秘密属性条件任务和秘密属性搜索任务上的表现。B. 在秘密属性搜索任务中，规划器面对不完美报告器时的鲁棒性。C. 报告器在视觉条件任务上经过训练后，性能的提升情况。所有误差线均表示多个回合的置信区间。

3.3 对无关报告的鲁棒性

我们在上一节的搜索 任务中看到，70B 规划器能够较好地应对执行器所犯的错误（例如，第 A.1 节）。在本节中，我们将检查它是否也能应对噪声报告器。我们不再假设环境中只有与任务相关的动作会被报告；环境中的无关动作（例如，“我向左移动了”/“我向右上方移动了”等）也会有 20% 的概率被报告。

我们发现，性能确实有所下降，但幅度并不显著（图 2B）。较小的70亿参数模型不如700亿参数模型鲁棒，性能下降也更为明显。我们发现，700亿参数规划器会采用重复策略（重复发出一条指令，直到收到相关报告，例如附录 A.2）或循环策略（依次循环检查所有物体，例如附录 A.3），或者结合两种策略，直到找到一个“优质物体”。

少样本提示没有提供如何回应无关报告的示例。当我们提供指导，并在提示示例中展示“重复”策略（例如附录 A.2）时，700亿参数规划器的表现恢复到没有无关报告时的水平（图 2B）；70亿参数规划器的表现有所提升，但未能完全恢复。这种鲁棒性表明，我们的方法（尤其是采用大型规划器时）有望扩展到不完美的报告器。不过，大型规划器的推理时间成本高，因此使用能够忽略无关事件的报告器效率更高。

4 训练一个真实的报告器

上一节中，我们重点研究了在智能体框架中，配备始终报告准确信息的报告器时规划器的行为。然而，在大多数环境中，这样的报告器并不存在。本节将探讨如何从零开始通过强化学习训练报告器。

我们考虑一种“视觉条件任务”：其中“秘密属性”不会通过特殊的“检查”动作直接以文本形式揭示，而必须从视觉观察中解码出来。具体而言，该任务规定：“如果 {decider object} 靠近墙，则拾取 {object 1}；否则拾取 {object 2}。”报告器的输入与执行器相同，都是视觉观察；其输出为一个二元分类器头，可在两种报告中进行选择（“物体{close to /far from}墙”）。请注意，训练刚开始时，报告器尚不具备任何现成的落地机制，无法准确报告场景信息。随着训练的持续进行，报告器可以利用回合的最终奖励，学习哪些信息对规划器最有帮助，最终逐渐学会只报告真实且相关的信息。

相比之下，近期研究采用具备视觉 grounding 的预训练模型（例如视觉语言模型 Zeng 等人 [2022], ）或手工设计的机制 Huang 等人 [2022]）来充当报告器模块。我们认为，构建有效的报告器模块应结合这两种方法：先使用预训练模块来引导感知与落地，再通过强化学习微调预训练模块，使其能够与规划器模块进行交流。我们的研究表明，采用强化学习进行报告器训练确实可行且有益。

5 讨论与未来工作

我们主张采用由三部分组成的系统（规划器-执行器-报告器）：将预训练语言模型用作规划器，向具身执行器发出自然语言指令，并由报告器将信息传回规划器。我们提出了一系列任务，利用预训练语言模型的抽象推理能力，展现出令人瞩目且稳健的零样本性能，并分析了不同规模模型中的错误。我们首次进行了概念验证，证明报告器可以

接受训练，从而促进规划器与执行器之间更好的协作。未来令人期待的研究方向包括将预训练组件纳入报告器、扩展至更复杂或更贴近现实的任务，以及通过让大型模型参与其中来改进训练。

参考文献

M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog 等。《照我能做的做，而非照我说的做：让语言落地于机器人可供性》。*arXiv* 预印本 *arXiv:2204.01691*，2022年。T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. J. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever 和 D. Amodei。语言模型是少样本学习者。*ArXiv*，abs/2005.14165，2020年。L. Espeholt, H. Soyer, R. Munos, K. Si monyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, S. Legg 和 K. Kavukcuoglu。IMPALA：采用重要性加权执行器-学习器架构的可扩展分布式深度强化学习。*CoRR*，abs/1802.01561，2018年。网址：http://arxiv.org/abs/1802.01561。J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals 和 L. Sifre。训练计算量最优的大型语言模型。*ArXiv*，abs/2203.15556，2022年。W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar 等。内心独白：通过使用语言模型进行规划实现具身推理。*arXiv* 预印本 *arXiv:2207.05608*，2022年。R. Kirk, A. Zhang, E. Grefenstette 和 T. Rocktaschel。深度强化学习泛化研究综述。*ArXiv*，abs/2111.09794，2021年。T. D. Kulkarni, K. Narasimhan, A. Saeedi 和 J. Tenenbaum。分层深度强化学习：整合时间抽象与内在动机。神经信息处理系统进展，29，2016年。M. Mitchell。人工智能中的抽象与类比生成。《纽约科学院年刊》，1505(1):79–101，2021年。A. Radford, J. Wu, R. Child, D. Luan, D. Amodei 和 I. Sutskever。语言模型是无监督的多任务学习者。2019年。J. W. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, R. Ring, S. Young, E. Rutherford, T. Hennigan, J. Menick, A. Cassirer, R. Powell, G. van den Driessche, L. A. Hendricks, M. Rauh, P.-S. Huang, A. Glaese, J. Welbl, S. Dathathri, S. Huang, J. Uesato, J. F. J. Mellor, I. Higgins, A. Creswell, N. McAleese, A. Wu, E. Elsen, S. M. Jayakumar, E. Buchatskaya, D. Budden, E. Sutherland, K. Simonyan, M. Paganini, L. Sifre, L. Martens, X. L. Li, A. Kuncoro, A. Nematzadeh, E. Gribovskaya, D. Donato, A. Lazaridou, A. Mensch, J.-B. Lespiau, M. Tsimpoukelli, N. K. Grigorev, D. Fritz, T. Sottiaux, M. Pajarskas, T. Pohlen, Z. Gong, D. Toyama, C. de Masson d'Autume, Y. Li, T. Terzi, V. Mikulik, I. Babuschkin, A. Clark, D. de Las Casas, A. Guy, C. Jones, J. Bradbury, M. G. Johnson, B. A. Hechtman, L. Weidinger, I. Gabriel, W. S. Isaac, E. Lockhart, S. Osindero, L. Rimell, C. Dyer, O. Vinyals, K. W. Ayoub, J. Stanway, L. L. Bennett, D. Hassabis, K. Kavukcuoglu 和 G. Ir ving。扩展语言模型：Gopher 训练方法、分析与见解。*ArXiv*，abs/2112.11446，2021年。J. Russin, R. C. O'Reilly 和 Y. Bengio。深度学习需要前额叶皮层。连接人工智能与认知科学的研讨会，107:603–616，2020年。R. S. Sutton, D. Precup 和 S. Singh。介于马尔可夫决策过程与半马尔可夫决策过程之间：强化学习中时间抽象的框架。人工智能，112(1-2):181–211，1999年。

A. Zeng, A. Wong, S. Welker, K. Choromanski, F. Tombari, A. Purohit, M. Ryoo, V. Sindhwani, J. Lee, V. Vanhoucke 等。苏格拉底模型：将零样本多模态推理与语言相结合。*arXiv* 预印本 *arXiv:2204.00598*，2022年。

A 规划者-报告者-执行者对话示例

本节列举了规划器-执行器-报告器对话的示例，尤其是秘密属性搜索任务中的示例，以展示规划器面对执行器和报告器出错时的稳健性。

A.1 更稳健的搜索策略。

The 红框显示执行器出错的位置，绿框突出显示规划器的补救措施 .

QUESTION: The objects in the room are **vertical striped red inverse ex**, horizontal striped teal circle, **vertical striped dark blue tee** and **vertical striped pink inverse ex**. Get the object with a good secret property.
ANSWER:
LM: Examine **vertical striped red inverse ex**.<EOS>
Agent: I examined **vertical striped pink inverse ex**. Its secret property has value bad.<EOS>
LM: Examine horizontal striped teal circle.<EOS>
Agent: I examined horizontal striped teal circle. Its secret property has value bad.<EOS>
LM: Examine **vertical striped dark blue tee**.<EOS>
Agent: I examined **vertical striped dark blue tee**. Its secret property has value bad.<EOS>
LM: Examine **vertical striped pink inverse ex**.<EOS>
Agent: I examined **vertical striped pink inverse ex**. Its secret property has value bad.<EOS>
LM: Examine **vertical striped red inverse ex**.<EOS>
Agent: I examined **vertical striped red inverse ex**. Its secret property has value good.<EOS>
LM: Pickup **vertical striped red inverse ex**.<EOS>

A.2 应对无关报告的涌现策略：重复策略。

即使收到无关报告，规划器也会重复发送命令，直到任务完成。

QUESTION: The objects in the room are **noisy dark red tee**, vertical striped cyan plus, **noisy purple u** and **horizontal striped pink triangle**. Get the object with a good secret property.
ANSWER:
LM: Examine **noisy dark red tee**.<EOS>
Agent: I have moved up and right.<EOS>
LM: Examine **noisy dark red tee**.<EOS>
Agent: I have moved up and right.<EOS>
LM: Examine **noisy dark red tee**.<EOS>
Agent: I have moved down and right.<EOS>
LM: Examine **noisy dark red tee**.<EOS>
Agent: I have moved down and left.<EOS>
LM: Examine **noisy dark red tee**.<EOS>
Agent: I examined **noisy dark red tee**. Its secret property has value bad.<EOS>
LM: Examine vertical striped cyan plus.<EOS>
Agent: I have moved left.<EOS>
LM: Examine vertical striped cyan plus.<EOS>
Agent: I examined vertical striped cyan plus. Its secret property has value good.<EOS>
LM: Pickup vertical striped cyan plus.<EOS>

A.3 对无关报告的响应中涌现的策略：循环策略。

规划器会循环执行检查命令，直到收到优质物体的报告。//

```
QUESTION: The objects in the room are grid pink tee, horizontal striped yellow upside down tee, vertical striped
light yellow h and noisy dark blue upside down u. Get the object with a good secret property.
ANSWER:
LM: Examine grid pink tee.<EOS>
Agent: I have moved up.<EOS>
LM: Examine horizontal striped yellow upside down tee.<EOS>
Agent: I have moved up and right.<EOS>
LM: Examine vertical striped light yellow h.<EOS>
Agent: I have moved down and left.<EOS>
LM: Examine noisy dark blue upside down u.<EOS>
Agent: I examined noisy dark blue upside down u. Its secret property has value bad.<EOS>
LM: Examine grid pink tee.<EOS>
Agent: I have moved down and left.<EOS>
LM: Examine horizontal striped yellow upside down tee.<EOS>
Agent: I have moved down.<EOS>
LM: Examine vertical striped light yellow h.<EOS>
Agent: I have moved down.<EOS>
LM: Examine noisy dark blue upside down u.<EOS>
Agent: I examined vertical striped light yellow h. Its secret property has value bad.<EOS>
LM: Examine grid pink tee.<EOS>
Agent: I have moved up and right.<EOS>
LM: Examine horizontal striped yellow upside down tee.<EOS>
Agent: I examined grid pink tee. Its secret property has value good.<EOS>
LM: Pickup grid pink tee.<EOS>
```

B 环境与智能体架构详情

B.1 网格词语环境

我们使用 PyColab 库（https://github.com/deepmind/pycolab）实现了一个网格世界环境。该网格世界为 11×11，外围边界是无法通行的墙，内部没有障碍物。环境中有四个物体和一个智能体，每个各占一个网格。每局游戏开始时，它们的初始位置都会随机分配。

我们从预先定义的允许值列表中，随机为环境中的物体分配颜色、纹理和形状 属性。评估任务中的确切物体属性组合不会出现在训练任务中，但每个属性值都曾在训练期间出现过。

智能体在环境中的视野始终是从俯视视角观察场景的 11x11 裁剪区域，智能体位于中心。智能体移动时，裁剪区域会滚动，以使智能体始终位于视觉观察的中心。

B.2 智能体架构

我们的智能体由三个模块组成：规划器、执行器和报告器。规划器模块是一个经过预训练的 LSLM。在实验中，我们使用 Chinchilla 模型的两个变体 [Hoffmann 等人，2022], 分别是具有 700 亿参数的模型（称为 70B 模型）和具有 70 亿参数的模型（称为 7B 模型）。

执行器模块使用一个预训练策略，该策略在同一网格世界环境中的简单任务上进行训练。该策略使用一个包含 3 层的简单卷积视觉编码器来编码视觉观察，并使用基于 LSTM 的语言编码器来编码动作指令。智能体还配有一个基于 LSTM 的记忆模块，以便在策略输出时考虑先前的动作和观察。执行器的策略头会输出动作空间上的一个分布；在我们的案例中，动作空间包含离散的移动动作（例如在网格世界中向上或向下移动），以及特殊动作 拾取和 检查。然后从策略输出中采样，得到智能体实际执行的动作。

报告器模块与执行者架构有许多相似之处：它也包含视觉编码器和指令编码器、记忆模块以及策略头。在第 4 章所述的实验中，策略头只是一个在两种预定义文本报告上的二元分布。不过，我们计划在未来的工作中，允许报告器模块直接进行语言生成。

B.3 训练流程

执行器和报告器模块均采用标准的 VTrace 损失 [Espeholt 等人， 2018]进行训练。执行器训练在大约 100,000 帧后收敛，而报告器训练在 500 帧后收敛。

需要注意的是，尽管报告器训练所需的帧数不多，但训练时间可能很长。这是因为在每个回合中，可能需要多次查询 LSLM 规划器。这些模型的推理时间仍然相当长，因此收集足够的轨迹来训练智能体所需的实际耗时也会增加。

C 所有任务描述

我们从具身智能的四个基本方面来界定智能体的能力，以及相应地，任务套件的要求：

> 1. **逻辑推理：** 处理复杂指令并对其执行不同类型逻辑运算，从而确定正确行动方案的能力。2. **泛化：** 超越智能体以往经验进行泛化的能力。3. **探索：** 探索智能体周围世界以发现新信息，并以此辅助其推理、决定采取何种行动的能力。4. **感知：** 利用智能体获得的原始观察（通常是视觉信息）处理周围世界，并根据所见作出决策的能力。

逻辑推理：接收复杂指令并对其执行不同类型的逻辑运算，从而确定正确行动方案的能力。

这项能力主要体现在具身智能体解读目标说明（提示）的含义，以及整合其掌握的有关环境的其他信息的能力上。这可能包括 if-else 条件语句、通过排除选项来择优选择、挑选符合特定属性的物体等。LSLM 尤其擅长这类语言逻辑任务 [Rae 等人, 2021年, Hoffmann 等人, 2022]。

泛化：超越智能体以往经验进行泛化的能力。

在强化学习中，针对新输入进行泛化的能力已得到充分研究 [Kirk 等人，2021],但仍是一项重大挑战。在 LSLM 领域，我们见证了通过少样本学习，将学习能力迁移到新的文本任务和输入上所取得的显著成果 [Brown 等人, 2020] 。这些“少样本示例”是提示中提供的最优行为语言轨迹；对于任何泛化任务，智能体都不会接收到新的交互数据或演示。相比演示或交互，为新任务收集语言描述的成本低得多，也容易得多。我们考察智能体如何从提示中提供的“训练”示例泛化到新的测试提示。

我们研究物体的泛化，即任务在训练期间使用一组特定物体进行过，测试时则使用其他物体。我们研究语言提示的泛化，即同一任务可以通过多种不同方式表达，但任务本身的结构保持不变。最后，我们研究任务本身的泛化，例如：训练轨迹中只展示了收集 2 个物体，而测试时要收集 3 个物体。

探索：探索周围世界以获取新信息的能力，这些信息可以为你采取何种行动提供推理依据。

这个定义与强化学习文献中通常所说的探索略有不同。这里指的并非训练期间探索策略空间以寻找更优策略（而非利用已知策略）的能力，而是指智能体主动发现当前状态下无法观察到的信息的能力。在我们的任务中，智能体必须移动并使用“检查”操作来发现有关物体的文本线索，或者运用感知来查看当前不在其视野中的物体。

感知： 利用智能体所获得的原始观测（通常是视觉）来处理世界，并根据所看到的内容做出决策的能力。

这最终是智能体最基本的能力，即观察周围环境并理解其含义。它对于强化学习至关重要，以至于通常甚至不会被列为一项核心能力。然而，这正是 LSLM 的核心不足之处：LSLM 仅在文本上运行，并未接地。本工作的核心贡献之一，就是展示如何将这些互补的能力结合起来。

C.1 任务

最后，我们展示一系列任务，以说明上述讨论的挑战。

选项排除： 在这里，我们探讨逻辑推理以及（物体和提示）泛化。请看下面这个启发性示例：

我知道开瓶器要么在餐具抽屉里，要么在酒柜上。我刚检查了餐具抽屉，但里面没有。你能帮我找到开瓶器吗?

我们期望智能体采取的行为相对简单，也就是“前往酒柜”。然而，要推断出这条简单的指令并非易事，因此我们考察 LSLM 是否能提供帮助。我们在 PyColab 环境中设计了一项任务，以模拟这类任务。每个回合开始时，我们会选取 4 个不同的物体并随机摆放。然后，我们根据这些物体的名称生成一个模板化的指令字符串。我们采用 10 种不同的语言格式，从中选择 7 种用于 N-shot 提示，并将另外 3 种留作测试格式，以考察跨提示的泛化。在所有这些格式中，4 个物体中的 3 个会被“排除”作为目标物体，而最优行为是让智能体前往最终未被排除的那个物体。如果智能体与任何其他物体互动，回合就会结束，且不会获得奖励。

基础步骤任务： 我们考虑此类任务中最简单的例子：任务是拾取房间里的两个物体，例如“按顺序拾取物体 X 和 Y”。我们希望语言模型能够将这条指令拆解为各个组成部分，即“拾取 X”和“拾取 Y”，并按顺序将它们发送给低层智能体。这是第一个多步任务，因此需要一个报告器（用于告知规划器何时发出下一条指令）。我们假设报告器会如实且相关地报告：每当智能体拾取一个物体时，便生成文本字符串“I have picked up <object name>.”。我们考察其对不同物体数量的泛化能力。

条件隐藏属性： 请看下面这个例子：如果咖啡还是热的，就给我拿咖啡；否则就给我拿一杯汽水。若没有具身智能体检查咖啡并报告其温度，语言模型就无法知道咖啡是否还热；语言模型可以据此发出下一条指令。为了在网格世界中模拟这类信息——例如温度或纹理，只有具身智能体直接与物体交互才能获取——我们假设每个物体都有一组隐藏属性。当智能体对某个物体执行特殊的“检查”动作时，环境会生成一个列出所有这些属性的文本字符串。

我们从房间里随机放置的 4 个互不相同的物体开始，智能体的目标是拾取一个目标物体。挑战在于确定正确的目标物体是什么——这需要通过与环境交互来收集信息。在我们的任务中，智能体必须检查一个任意指定的判定物体，该物体可能具有 *good* 或 *bad* 属性。根据它具有哪种属性，智能体必须拾取不同的目标物体。除判定物体外，其他物体既非好也非坏，其秘密属性为未知；检查一个非判定物体 X 时，会返回字符串“我检查了 X。它的秘密属性值未知。”

搜索秘密属性：

请看下面这个例子：你能把我的水瓶拿给我吗？它可能在冰箱里、健身包里，也可能在背包里。语言模型不知道水瓶在哪里，低级智能体必须收集这方面的信息。此任务的挑战在于制定并执行一系列信息收集动作，识别智能体何时已获得必要信息、可以停止继续收集信息，并根据收集到的信息发出最终指令。

我们在一项搜索任务中模拟这种结构：房间里的四个物体中有一个是目标，但智能体不知道是哪一个。相关信息包含在物体的隐藏属性中：四个物体中有三个具有 坏 属性，剩下的一个具有 好 属性，而智能体必须拾取具有 好 属性的物体。检查物体 X 会返回文本字符串“I examined X. Itssecret

property has value good/bad.' 因此，智能体必须依次检查每个物体；当它遇到具有好 属性的物体时，就应该拾取它。

视觉颜色条件任务：

我们从一个简单的条件任务开始，目标物体会根据智能体的颜色而变化。语言模型无法获取智能体的颜色信息。因此，报告器必须学会从视觉观察中解读颜色信息，并将其报告出来。随后，语言模型便能发出正确的下一条指令，从而获得奖励。此任务不需要信息收集动作或命令，因为智能体生成时就能直接看到自己的颜色。

视觉位置条件任务： 前一个任务实际上是一个单步任务——从任务的第一步起，就能获得确定目标物体所需的信息（即智能体的颜色）。为了将前一节中考察的多步任务挑战与学习报告相结合，我们考虑一种更类似于前面所考察任务的条件任务，在这种任务中，智能体必须执行信息收集动作才能获取相关信息。我们考虑这样一种条件任务：智能体必须拾取哪个物体，取决于指定的判定物体是否靠近墙。为了判断这一点，智能体必须移动到该物体旁，检查其周围环境。

D 强化学习基线

我们还将执行器（在"拾取"和"检查"任务上预训练）直接用于秘密属性条件任务和搜索任务（3）的训练。请注意，由于存在轻微的分布偏移，我们也能看到基础"拾取"和"检查"任务的学习曲线——这里的执行器还会将事件报告作为单独的观察接收（使用与指令编码相同的架构进行编码）；这是执行信息收集任务（即利用生成的信息）所必需的。对于基础的拾取和检查任务，智能体只需学会忽略这一字段即可（因此学得很快）。

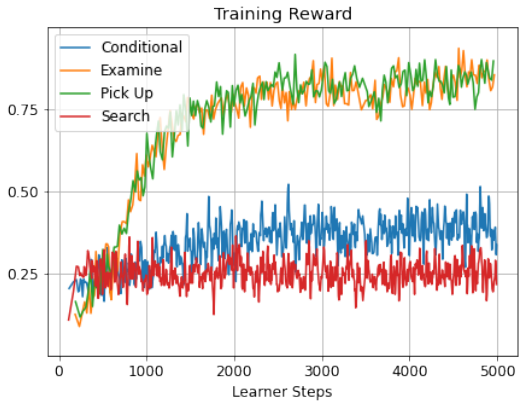

图 3：**纯强化学习基线**。详情请参阅正文。

经过 5000 次学习器更新（1.4 亿学习器帧）后，我们发现搜索任务的表现约为 25%（若不先检查物体以判断其是否合适，而是随机拾取一个物体，则这一表现相当于随机猜测）；条件任务的表现略高，约为 33%。这表明强化学习智能体能够学会拾取提示中的三个物体之一（并忽略第四个），但没有学会先检查判定物体，以确定应该拾取哪个目标物体。

这些基线并不是要表明这些任务不可能通过纯强化学习学会——更好的架构、折扣方法、探索策略等无疑会显著提升性能。然而，我们证明了，即使进行了多次环境交互，纯强化学习基线仍然难以完成这些任务；而我们的规划器-执行器-报告器智能体无需额外的环境交互，仅凭规划器提示中的5个最优表现示例，就能取得相当不错的表现。

E 所有使用过的提示格式

E.1 条件任务提示

问题：如果深蓝色实心 h 是好的，则拾取浅绿色横条纹反向加号。否则，拾取棋盘格棕色T形。答案：语言模型：检查深蓝色实心 h。<EOS>智能体：我检查了深蓝色实心 h。它的秘密属性值为坏。<EOS>语言模型：拾取棋盘格棕色T形。<EOS>智能体：我拾取了棋盘格棕色T形。<EOS>完成

问题：如果浅绿色噪声加号是好的，就拾取橙色实心圆 .
否则，拾取蓝色竖条纹倒置 u。答案：LM：检查浅绿色噪声加号。<EOS>智能体：我检查了浅绿色噪声加号。它的秘密属性值为坏。<EOS>LM：拾取蓝色竖条纹倒置 u。<EOS>智能体：我拾取了蓝色竖条纹倒置 u。<EOS>完成

问题：如果薰衣草色竖条纹倒置 T 形是好的，就拾取橙色实心圆反向加号。否则，拾取绿色竖条纹 h。答案：LM：检查薰衣草色竖条纹倒置 T 形。<EOS>智能体：我检查了薰衣草色竖条纹倒置 T 形。它的秘密属性值为坏。<EOS>LM：拾取绿色竖条纹 h。<EOS>智能体：我拾取了绿色竖条纹 h。<EOS>完成

问题：如果粉色竖条纹加号是好的，就拾取水平条纹蓝色茶。否则，拾取垂直条纹橙色叉。答案：LM：检查粉色竖条纹加号。<EOS>智能体：我检查了粉色竖条纹加号。它的秘密属性值为坏。<EOS>LM：

拾取垂直条纹橙色叉。<EOS>智能体：我拾取了垂直
条纹橙色叉。<EOS>完成

问题：如果网格青色H形是好的，就拾取垂直条纹紫色T形；否则，拾取垂直条纹深红色倒U形。答案：语言模型：检查网格青色H形。<EOS>智能体：我检查了网格青色H形。它的秘密属性值为好。<EOS>语言模型：拾取垂直条纹紫色T形。<EOS>智能体：我拾取了垂直条纹紫色T形。<EOS>完成

E.2 搜索任务提示

问题：房间里的物体有棋盘格棕色T形、水平条纹浅绿色反加号、深蓝色实心 h 和垂直条纹蓝色T形。找到秘密属性为好的物体。答案：语言模型：检查棋盘格棕色T形。<EOS>智能体：我检查了棋盘格棕色T形。它的秘密属性值为坏。<EOS>语言模型：检查水平条纹浅绿色反加号。<EOS>智能体：我检查了水平条纹浅绿色反加号。它的秘密属性值为坏。<EOS>语言模型：检查深蓝色实心 h。<EOS>智能体：我检查了深蓝色实心 h。它的秘密属性值为好。<EOS>语言模型：拾取深蓝色实心 h。<EOS>智能体：我拾取了深蓝色实心 h。<EOS>完成

问题：房间里的物体有噪声绿色加号、水平条纹蓝色 U 形、嘈杂绿色圆形和纯色浅绿色倒U形。拿起具有良好秘密属性的物体。答案：LM：检查噪声绿色加号。<EOS>智能体：我检查了噪声绿色加号。它的秘密属性值为坏。<EOS>LM：检查水平条纹蓝色 U 形。<EOS>智能体：

我检查了水平条纹蓝色 U 形。它的秘密属性值为坏。<EOS>LM：检查嘈杂绿色圆形。<EOS>智能体：我检查了嘈杂绿色圆形。它的秘密属性值为好。<EOS>LM：拾取嘈杂绿色圆形。<EOS>智能体：我拾起了嘈杂绿色圆形。<EOS>完成

问题：房间里的物体有纯绿色三角形、纯黄色 H 形、方格薰衣草色加号和嘈杂蓝色 T 形。拿起秘密属性良好的物体。答案：语言模型：检查纯绿色三角形。<EOS>智能体：我检查了纯绿色三角形。它的秘密属性值为良好。<EOS>语言模型：拾取纯绿色三角形。<EOS>智能体：我拾取了纯绿色三角形。<EOS>完成

问题：房间里的物体有水平条纹棕色 T 形、网格黄色 X 形、纯深红色加号和方格浅黄色 X 形。拿起秘密属性良好的物体。答案：语言模型：检查水平条纹棕色 T 形。<EOS>智能体：我检查了水平条纹棕色 T 形。它的秘密属性值为良好。<EOS>语言模型：拾取水平条纹棕色 T 形。<EOS>智能体：我拾取了水平条纹棕色 T 形。<EOS>完成

问题：房间里的物体有垂直条纹黄色加号、方格深红色 U 形、方格蓝色加号和垂直条纹桃色倒 T 形。拿起秘密属性良好的物体。答案：语言模型：检查垂直条纹黄色加号。<EOS>智能体：我检查了垂直条纹黄色加号。它的秘密属性值为良好。<EOS>语言模型：拾取垂直条纹黄色加号。<EOS>智能体：

我拿起了垂直条纹黄色加号。<EOS>DONE}