arXiv:2304.00584v1 [cs.RO] 2 Apr 2023

面向任务的多模态人机协作的端到端人类模拟器

Afagh Mehri Shervedani[1], Siyu Li[1], Natawut Monaikul[2],BaharehAbbasi[3],BarbaraDiEugenio[2], 和 Milo˘s ˘Zefran[1]

摘要——本文提出了一种基于神经网络的用户模拟器，可为强化学习（RL）智能体提供多模态交互环境，用于训练涉及多种交流模式的协作任务。该模拟器基于现有的 ELDERLY-AT-HOME 语料库进行训练，并支持语言、指向手势和触觉示意动作等多种模态。本文还提出了一种新颖的多模态数据增强方法，旨在应对人类示范收集过程成本高、耗时长所导致的数据集有限这一挑战。总体而言，本研究凸显了利用强化学习和多模态用户模拟器开发与改进家用辅助机器人的潜力。

I. 引言

为家庭使用而设计的辅助机器人能够通过协助老年人和残疾人士完成烹饪、清洁等日常生活活动（ADL），帮助他们保持独立生活能力。这些机器人通常依靠三个主要模块来执行感知-规划-行动循环：感知模块处理感觉信息，执行模块 执行所需动作，而机器人的核心可能是交互管理器，它接收来自感知模块的输入，并确定机器人采取行动的最有效方案。图 1 展示了感知-规划-行动循环。

要让机器人像人一样协助完成日常生活活动（ADL），它不仅必须能够有效地协作完成任务，还必须能够自然地与用户进行交互。这种交互通常具有多模态特性，会无缝交替运用各种交流模式，例如语言、手势、面部表情和力交互。因此，为这种人机协作而设计的交互管理器必须足够稳健，能够处理这些多种模态。

在先前的工作中， [1], [2], 我们开发了一种名为分层二部动作-转移网络（HBATNs）的新框架，将其作为交互管理器的基础，供参与我们所谓的*Find* 任务的辅助机器人使用。在这项任务中，两名参与者合作寻找一个物体，其中一人可能行动能力受限。HBATNs 的设计直接借鉴了 ELDERLY-AT-HOME 语料库 [3], 中的人类示范。该语料库包含护理专业学生（标记为 HEL）与接受日常生活活动协助的老年人（标记为 ELD）之间经过标注的交互。虽然 HBATNs 根据人类示范，为参与“Find”任务的机器人提供了最优策略，但它们是手动构建的，难以轻易适配其他任务。

在我们最近的研究中 [4], ，我们提出了一种更具可扩展性的交互管理器，它采用强化学习（RL）自动提取机器人参与协作任务时的最优策略。训练该智能体的一项重大挑战，是构建一个能够为智能体提供奖励的交互环境。在面向任务的协作任务中，交互环境涉及人类。由于在强化学习训练阶段让人类与人工智能（AI）智能体交互并不现实，研究人员开发了用户模拟器来模拟这类交互中的人类行为。然而，现有用户模拟器仅支持一种模态，即语言。因此，我们开发了自己的用户模拟器，用于模仿多模态交互中的人类行为。该模拟器需要解读多模态输入并生成多模态响应。

本文提出了一种新颖的基于神经网络的用户模拟器，其灵感来自行为克隆 [5],[6] ，并使用 ELDERLY-AT-HOME 语料库中的“Find”任务演示进行训练。我们的用户模拟器的独特之处在于，它能够适应多种模态，例如语言、指向手势和触觉示意（H-O）动作 [3]。这是家用辅助机器人研发的一项重大进展，因为目前尚无模拟器能够为强化学习训练提供多模态交互环境。

另一项重要贡献是我们提出的新型多模态数据增强方法，该方法解决了由于收集人类示范成本高且耗时，导致只能使用有限数据集这一挑战。我们的数据增强方法能够有效处理数据中涉及的所有模态，为社区提供了宝贵的资源。总体而言，我们的研究凸显了使用强化学习和多模态用户模拟器开发和改进家用辅助机器人的潜力，并代表了该领域的一项重要进展。

本文其余部分安排如下。首先，我们

[1]A.M. Shervedani、S. Li 和 Milo˘s ˘Zefran 任职于伊利诺伊大学芝加哥分校电气与计算机工程系机器人实验室，美国伊利诺伊州芝加哥市，邮编 60607。[2]N. Monaikul 和 B. Di Eugenio 任职于伊利诺伊大学芝加哥分校计算机系自然语言处理实验室，美国伊利诺伊州芝加哥市，邮编 60607。[3]B. Abbasi 任职于加利福尼亚州立大学海峡群岛分校计算机科学系，美国加利福尼亚州卡马里奥市，邮编 93012。本研究获美国国家科学基金会资助，项目编号为 IIS-1705058 和 CMMI-1762924。

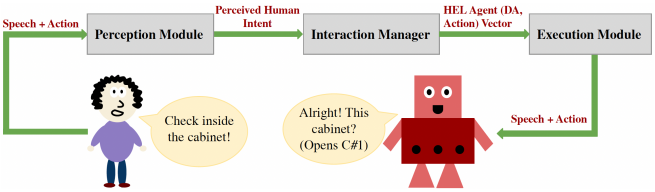


图 1：辅助机器人中的感知-规划-行动 循环。另请参见 [4]，其中有更详细的图示。

在第二节回顾相关研究。在第三节介绍必要的预备知识。在第四节展示我们的用户模拟器框架，并在第五节描述评估。最后，我们在第六节总结研究结果并给出结论。

II. 相关工作

模拟器是通过机器学习开发机器人自主能力的重要工具。人机交互（HRI）和辅助机器人开发面临的挑战在于，模拟器需要对复杂的人类行为进行建模。近期在人机交互和人因领域的研究，已成功地描述、理解并预测特定情境下的人类决策 [7]。与人机交互相关的方法大致可分为三类。一种常见的人类行为建模技术，是通过运动方程对人体运动的力学特性进行建模 [8]，[9]。另一种技术则假设人类智能体的奖励函数与机器人的奖励函数几乎相同。人机交互领域经常采用这一假设 [10]。例如，在人类智能体和机器人智能体从桌上挑选工具的场景中，奖励被定义为 xy 平面上的一个函数。此外，许多关于人机交互的研究假设人类智能体是与机器人交互的专家 [11]，[12]，，因此，人类总会选择便于机器人理解的动作。举例来说， [12]，人类智能体的策略是一个由机器人动作决定的固定且确定性函数。然而，所有这些技术都对人类智能体施加了严格的假设。在我们的工作中，我们放宽了其中一些假设，并根据真实世界数据对人类智能体进行建模。

在人机交互领域，家庭环境中的老年人照护受到广泛关注。我们先前的工作 [1]，[2]使用 HBATNs 同时对机器人助手和老年人进行建模，以维持任务驱动的多模态交互状态并规划后续动作。我们假设人类行为由对话动作、指向手势和触觉示意线索构成。该领域的类似研究，如 [13]，，提出了一种多模态动作识别框架，分别处理语音和视频信息，然后将二者融合。在 [14]，中，研究者提出使用定时 Petri 网来表示具有社会智能的机器人行为，并强调这种表示模型时间和异步性的能力的重要性

事件。然而，这些研究均使用手工构建的人类模拟器进行评估，因此难以推广到其他任务。本文在 HBATN 框架的基础上，提供了一个更通用的用户模拟器版本，使其能够在交互环境中实现多模态交互。

随着机器学习迅速发展，人机交互（HRI）迎来了新的机遇。智能对话系统的最新进展在对人类智能体的对话方面进行建模时展现出良好前景 [15]–[17]。例如，Li [17]开发了一种混合用户模拟器，能够自然地与对话系统交互，以进行端到端训练。然而，当考虑人类除了对话之外还能执行其他行为的场景时，用户模拟器研究就变得更加复杂。 [18] 指出，有三种方式可以建模人类行为变化对机器人状态的影响：将人类视为环境的一部分、让机器人学习人类如何反应，或假设人类具有自主性并拥有自己的“心智理论”。尽管前两种方式提供了更简单的黑箱解决方案，但我们采用的第三种方式能够更细致地理解人类行为与意图。然而，与传统机器学习相比，可用数据有限，因此训练此类模型颇具挑战。为解决这一问题，我们提出了一种在人机交互小型数据集上展现出良好效果的人类模拟器。

III. 预备知识

我们先前的研究 [1]，[2]，[4] 基于 ELDERLY-AT-HOME 语料库 [3]，开发，这是一个公开可用的人类之间多模态交互语料库。该语料库涉及辅助日常生活活动，例如穿鞋和准备晚餐。我们基于一个由 与查找 任务相关的交互组成的子语料库，开发了 HBATN 框架，随后又开发了强化学习框架。也就是说，老年参与者（ELD）会请求某个物体，而协助者（ HEL）会通过提出后续问题来尝试找到它。

在 [1],中，查找 任务被分解为一组子任务，以确定两个主要未知信息：目标物体（O）及其位置（L）。四个主要子任务分别是确定所需物体类型（确定(O_T)）、确定一个待检查的潜在位置（确定(L)）、打开该位置（打开(L)</style），以及确定实际物体（确定(O)）。这些子任务被建模为动作转换网络（ActNet）。

ActNet 是一种二分图，用于表示双方参与者的状态及其可能采取的多模态动作。这些动作定义为由语言特征（话语的对话行为（DA） [3]，即说话者的意图以及物体或位置相关词语）和物理特征（指向手势或触觉示意（H-O）动作）构成的向量。HBATN 包含这些 ActNet，使机器人不仅能够推断伙伴的状态，还能据此规划下一步动作。

随后，在 [2]，中，我们将模型泛化，使机器人既可以作为 ELD，也可以作为 HEL，方法是将子任务分解为我们称之为原始子任务的单元。在这一

新表述中，*Det(O_T)* 和*Det(L)* 用于建立物体类型及其位置（*Estab*），之后可能进行验证（*Verify*），或提出指定更多信息的问题（*Spec*）；而*Det(O)* 则确认当前地点是否存在所需物体（*Finish*），或与协作伙伴验证某个实体物体。我们证明，配备了训练好的分类器（用于判断参与者当前处于哪个子任务）的 HBATN，能够对 HEL 和 ELD 两种行为进行建模并执行。

最后，在我们最近的研究中 [4]，，我们提出了一种强化学习方法，用于提取人机交互中的最优策略。该策略取代了我们 HEL 机器人的手工设计的 HBATN 策略。如第 I 节所述，强化学习训练过程中的一项主要挑战是为智能体提供一个交互环境。因此，我们开发了一个基础用户模拟器，在训练期间充当 ELD 并与强化学习智能体交互。

我们在 [4]，中开发的用户模拟器（我们称之为基础用户模拟器模型（***BUS*** 模型））足以向 HEL 智能体提供正确的意图。然而，改进此类用户模拟器仍然是一项具有挑战性的研究问题。本文其余部分将重点介绍我们提高用户模拟器准确性的方法。我们将改进后的模型称为通用用户模拟器模型（***GUS*** 模型）。

IV. 用户模拟器框架

A. 特征提取

在这项工作中，我们扩展了先前的工作，并引入了一种端到端模型，用于预测世界状态，其中包括 ELD 状态以及 ELD 的下一步动作。

我们的模型隐含地被设定为充当 ELD。然而，我们对数据的分析发现，HEL 可能会连续采取多个动作。为了处理 HEL 连续采取多个动作的情况，模型应判断 ELD 是否会在下一步采取动作。总之，模型的目标有两个：(1) 预测 ELD 的下一个动作以及该动作是否会发生。ELD 的动作由 ELD 的对话行为和动作决定。(2) 确定世界状态。世界状态由 ELD 对 HEL 关于物体类型、位置和物体的知识所持有的信念决定。

为了有效地训练端到端模型，我们必须首先确定哪些信息有助于模型判断 ELD 的状态及其动作。以下各项总结了选择特征时应考虑的重要要点，以及我们如何对收集的人际交互数据进行特征化和标注。

- 要判断 HEL 是否会连续动作，我们需要知道前一行动者（当前行动者是 HEL）。这一信息可以用一个长度为 2 的向量表示。如果试次尚未开始，且当前行动者 HEL 正在发起试次，则两个元素均为 0。如果 HEL 不会连续动作，前一行动者就是 ELD，向量的第一个元素为 1。否则，第二个元素为 1。

- 了解是否提及了某种物体类型或位置非常重要。这些可以用二元特征表示。

- 模型还需要包含有关 ELD 先前状态的信息。我们提出以下表示方式：(1) ELD 对 HEL 是否了解 O_T 的信念，该信念有三种可能值（ELD 认为 HEL 不知道目标 $O_T \rightarrow 0$、ELD 认为 HEL 知道目标 $O_T \rightarrow 1$，或 ELD 认为 HEL 正在考虑另一个 $O_T \rightarrow 2$）；(2) ELD 对 HEL 是否了解 L 的信念；以及 (3) ELD 对 HEL 是否了解 O 的信念。我们提取了这三个参数所有有意义的可能组合，得到以下 13 种不同状态：(1) 状态 (0，0，0)，(2) 状态 (0，1，0)，(3) 状态 (0，2，0)，(4) 状态 (1，0，0)，(5) 状态 (1，0，1)，(6) 状态 (1，0，2)，(7) 状态 (1，1，0)，(8) 状态 (1，1，1)，(9) 状态 (1，1，2)，(10) 状态 (1，2，0)，(11) 状态 (2，0，0)，(12) 状态 (2，1，0)，(13) 状态 (2，2，0)。我们使用一个长度为 13 的独热编码向量来表示这些状态。关于这些状态的详细说明见 V-B 节。

- 另一个需要考虑的信息是 HEL 是否做出了指向手势。如果做出了，还需判断其指向的是物体还是位置。这是一种类别特征，可转换为一个长度为五的向量。如果没有做出指向手势，向量的前两个元素均为 0。如果 HEL 指向位置，第一个元素为 1；如果 HEL 指向物体，则第二个元素为 1。向量的后三个元素用于判断 HEL 指向的位置或物体是否与 ELD 的位置或物体匹配。如果该向量的第三个元素为 1，表示 HEL 指向的位置或物体与 ELD 的相同；如果第四个元素为 1，则表示 HEL 指向了错误的位置或物体。如果第五个元素为 1，则表示 HEL 指向了正确的物体类型，但不是正确的具体物体。例如，假设 ELD 要求一个小碗，而 HEL 回应时指向了一个大碗；向量 (0，1，0，0，1) 表示 HEL 的指向手势。

- 同样，对于 H-O 动作，我们必须确认该动作是否已经发生，以及它是针对物体还是位置。此外，我们还需要确定 H-O 的类型（打开或关闭位置、触摸位置或物体、取出物体、拿持物体）。这是一个类别特征，其向量大小为十。前五个元素的解释方式与指向手势向量完全相同。H-O 动作向量的后半部分通过对动作类型进行独热编码，确定五种不同 H-O 动作中的哪一种已经执行。

- 我们还加入了当前 HEL 的动作和当前 HEL 的话语（对话行为标签），二者均为类别

特征，分别由大小为九和十四的向量表示。HEL 动作类别如下分类：（1）无动作，（2）请求 O_T，（3）请求 L，（4）验证 O_T，（5）验证 L，（6）验证 O，（7）确认，（8）是，（9）否。对话行为类别如下分类：（1）无话语，（2）指令，（3）确认，（4）疑问词查询，（5）是非查询，（6）带信息请求的回复，（7）肯定回复，（8）否定回复，（9）检查，（10）解释，（11）对齐，（12）肯定陈述，（13）否定陈述，（14）陈述。这些是用于标注“ELDERLY-AT-HOME”语料库的对话行为 [3]。

• 除当前的 HEL 信息外，我们的模型还依赖于 ELD 先前动作的信息。在此，我们使用 ELD 的动作和 ELD 的对话行为，它们分别是由七维和十四维向量表示的类别特征。ELD 的动作类别如下所示：(1) 无动作，(2) 给出 O_T，(3) 给出 L，(4) 给出 O_T，L，(5) 确认，(6) 是，(7) 否。ELD 的对话行为类别与 HEL 的相同。

B. 数据标注

ELDERLY-AT-HOME语料库中的 查找 任务数据 [3]此前已完成转录，并针对对话行为、指向手势和 H-O 动作进行了标注。如 IV-A 节所述，我们需要为网络提供额外特征，以训练我们的用户模拟器。我们根据上一节为每项特征引入的类别，对 ELD 对 HEL 的知识（包括 O_T、L 和 O）的信念，以及 ELD 和 HEL 的动作进行了额外标注。

ELD 对 HEL 是否知道 O_T、L 或 O 的判断，是基于这样一条启发式规则客观确定的：每当 HEL 表现出对这些实体的了解或不了解时，ELD 就会更新自己的信念。例如，ELD 假定 HEL 不知道 O_T 或 L，直到 HEL 确认 ELD 对它的描述，或采取动作予以确认。如果 ELD 已指定某个特定的 O_T 或 L，而 HEL 却选错了，ELD 就会假定 HEL 想的是另一个 O_T 或 L。例如，若 ELD 指着一个小碗说“把那个碗拿给我”，而 HEL 指着一个大碗问“那个碗吗？”，ELD 就会认为 HEL 想的是另一个碗。

两名标注员对 ELD 和 HEL 的动作进行了标注。由于动作标注没有遵循严格的准则，因此解释空间较大；对于两类动作，均使用科恩κ系数衡量标注员间的一致性。为检验标注的可靠性，从数据集中随机抽取了 40 个 ELD 动作和 40 个 HEL 动作，两名标注员先独立标注这些动作，再标注其余动作。结果显示，两名标注员对 ELD 动作（$\kappa = 1.0$）和 HEL 动作（$\kappa = 0.81$）的一致性都很高，表明动作标签可靠。

C. 数据增强

在 寻找任务期间收集的数据，适合用于为训练用户模拟器奠定坚实基础；然而，几乎没有或只有少数实例显示，ELD 认为 HEL 对 O_T 或 L 的理解有误，或者不知道 O_T。这种缺乏变化的情况并不令人意外，因为两个人之间的交互相对简单直接。然而，由于我们提出将这个用户模拟器作为用于强化学习训练的交互环境的主要组成部分，并且我们预计经过强化学习训练的 HEL 智能体会犯错，因此需要扩充数据，纳入缺失或罕见状态的示例。

为了增加 ELD 认为 HEL 不知道 O_T、状态为 (0,0,0) 和 (0,1,0) 的样本数量，我们抽取 ELD 认为 HEL 知道O_T 的样本，并将 HEL 的话语和动作替换为请求 O_T 的示例（见图 2a）。随后，ELD 会再次给出关于 O_T 的指令。

为了增加 ELD 认为 HEL 想错了 O_T 、状态为 (2, 0, 0) 和 (2, 1, 0) 的样本数量，我们抽取 HEL 在话语中提及目标O_T 的样本，并将其替换为错误的目标（见图 2b）。随后，ELD 会再次给出关于 O_T 的指令。

我们还通过抽取 HEL 的话语或动作中包含物体或位置的样本，并将其替换为错误内容，来增加 ELD 认为 HEL 想的不是同一个 L 或 O、状态为 (1, 2, 0) 和 (1, 1, 2) 的样本数量。在每种情况下，ELD 随后都会再次给出正确的 O_T 或 L （见图 2c）。这些合成示例将帮助用户模拟器恰当地应对 HEL 智能体所犯的错误。

在我们之前的研究中 [4]，我们仅对 ELD 的输出状态进行了数据增强。然而，仔细检查数据后，我们发现，输入状态中只包含此前提到的前九种状态组合，而 ELD 的输出状态中则能看到全部十三种组合，尽管后几种很少出现。也就是说，第 10 至第 13 种状态组合具有实际意义，并且很可能出现在用户模拟器与 HEL 机器人智能体的交互过程中。如果训练期间没有见到所有可能且有意义的输入，模型就会过于局限，无法应对可能遇到的所有情况。这也会导致最终的准确率评估不准确，因为模型是在变化不足的输入上进行训练、验证和测试的。因此，我们对数据点进行增强，使输入涵盖全部十三种状态组合，从而让模型更加准确、灵活。

为了合成前一动作中 ELD 认为 HEL 心中有错误 O 的输入状态，即状态 (1,1,2)，我们随机选取一些输入状态为 (1,1,1) 的实例，也就是说，前一动作中 ELD 认为 HEL 心中有正确 O 的实例，并相应地更改 ELD 的前一动作。为此，ELD 会再次向 HEL 告知该物体。因此，我们将 ELD 前一动作的对话行为和动作分别替换为“Instruct”和“Give Specific O_T”。需要指出的是，之后我们将动作“Give Specific O_T”和“Give O_T”合并为同一个类别，因为在与 HEL 交互时，这两个动作传达的是

相同的信息；当 ELD 在与 HEL 交互时宣布物体类型或具体物体时，重要的只有 ELD 的状态差异。

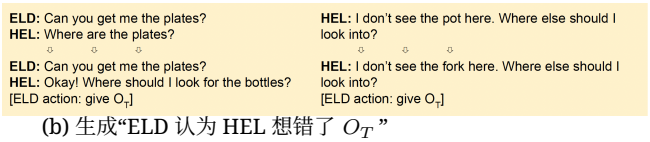
为了合成这样的输入状态：在上一步中，ELD 认为 HEL 已掌握有关 O_T 和 L的信息，但对 O_T 和/或 L有误解，状态为 (1,2,0)、(2,1,0)、(2,2,0)，我们随机抽取一些数据点，其中在上一步中 ELD 认为 HEL 对 O_T 和L 掌握正确，然后相应地更改 ELD 上一步的动作。对于状态 (1,2,0)，ELD 会再次告知 HEL 位置，因此我们将 ELD 上一步的对话行为和动作分别替换为“Instruct”和“Give L”。同样，对于状态 (2,1,0)，ELD 会再次告知 HEL 物体类型，因此我们将 ELD 上一步的对话行为和动作分别替换为“Instruct”和“Give O_T”。最后，对于状态 (2,2,0)，ELD 会再次告知 HEL 物体类型和位置，因此我们将 ELD 上一步的对话行为和动作分别替换为“Instruct”和“Give O_T，L”。

为合成这样的输入状态：在上一步中，ELD 认为 HEL 想错了 O_T，并且对 L 和 O 一无所知，即状态 (2,0,0)，我们随机抽取一些样本，其中 ELD 在上一步中认为 HEL 想对了 O_T，然后将 ELD 上一步的对话行为和动作分别改为“指令”和“给予 O_T”。

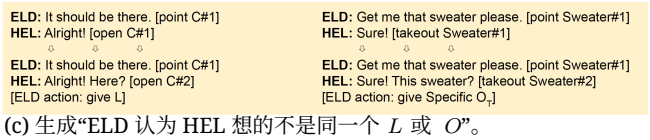
通过这种数据增强方案，我们将数据点从 693 个增加到 1932 个。



(a) 生成“ELD 认为 HEL 不知道 O_T”



(b) 生成“ELD 认为 HEL 想错了 O_T”



(c) 生成“ELD 认为 HEL 想的不是同一个 L 或 O”。

图 2：使用未见过或罕见状态增强数据的示例

D. 模型架构与训练

我们的模型是一个由三个全连接（密集）层和一个丢弃层组成的神经网络，丢弃率为=0.2，以防止过拟合并提升模型的泛化能力。训练过程中，我们使用了交叉熵损失函数和 Adam优化器。训练轮次最多为 100 轮，同时我们还在训练期间评估模型在验证集上的表现，以便提前停止训练。神经网络的输入是前文所述的特征，输出则是 ELD 的下一个状态、对话行为和动作，这些内容均由人工在数据中进行了标注。我们使用 PyTorch 库实现了该模型 [19]。

V. 评估

A. 基于数据的模型评估

模型使用 80% 的数据（共 1548 个数据点）进行训练。约 10% 的数据（共 183 个数据点）在训练过程中用于验证，以便在发生过拟合之前提前停止训练。其余数据（共 201 个数据点）用于评估充分训练后的模型性能。

除了整体准确率外，我们还评估了模型每项单独输出的分类准确率，即：（1）预测的 ELD 动作；（2）预测的 ELD 对话行为；（3）预测的 ELD 状态，以及 ELD 对 HEL 的信念（O_T，L，O）。

需要指出的是，[4] 中报告的分数是基于在原始测试集上测试 BUS 模型得出的，而非基于增强测试数据。不过，在增强数据上测试 BUS 模型也会得到相似的结果。

此处报告了在原始测试数据集和增强测试数据集上测试 BUS 模型和 GUS 模型的准确率结果。结果汇总于表 I 中。

	总体准确率	动作准确率	对话行为准确率	状态准确率
BUS 模型，Org.测试数据	46.27%	52.22%	53.73%	83.58%
BUS 模型，Aug.测试数据	45.83%	56.77%	51.04%	68.75%
GUS 模型，Org.测试数据	66.67%	81.82%	72.73	93.94%
GUS 模型，Aug.测试数据	70.85%	77.89%	75.38	89.44%

表 I：BUS 和 GUS 模型的分类准确率

表 I 中的结果显示，BUS 模型到 GUS 模型取得了显著提升。这一显著提升源于我们对特征提取、数据增强以及模型架构本身所做的改动，可概括如下：（1）按照 IV-A 节中的详细说明，更改 ELD 的状态表示；（2）将 ELD 先前的对话行为和动作信息作为模型的输入；（3）按照第 IV-C 节中的详细说明，扩充 ELD 的输入状态；（4）在网络架构中增加层，并移除网络中的非线性 ReLU 激活函数；（5）不进行样本重加权，因为通过适当的数据增强，类别不平衡问题已得到解决；此外，鉴于数据量有限，我们不希望过度重加权。考虑到数据量较少，重加权实际上会对模型性能产生不利影响。

为进一步分析表 I 中报告的评估结果，我们研究相应的混淆矩阵。在继续讨论混淆矩阵之前，需要记住，正如 IV-A 节所解释的，我们的训练好的模型还应判断 ELD 是否会执行动作。如果 HEL 执行

	否Utt	Inst	Ack	Q-w	Q-yn	R-w	R-y	R-n	Chk	Exp	Algn	St-y	St-n	St
否 Utt	**67.79%**	0%	0%	0%	0%	1.69%	16.95%	0%	0%	0%	0%	6.78%	0%	0%
Inst	12%	**46%**	0%	0%	0%	12%	0%	**30%**	0%	0%	0%	0%	9.09%	0%
Ack	0%	0%	**66.67%**	0%	0%	0%	0%	0%	0%	0%	0%	**33.33%**	0%	0%
Q-w	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Q-yn	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R-w	0%	2.32%	0%	0%	0%	**93.03%**	4.65%	0%	0%	0%	0%	0%	0%	0%
R-y	0%	0%	0%	0%	0%	0%	**100%**	0%	0%	0%	0%	0%	0%	0%
R-n	0%	20%	0%	0%	0%	0%	0%	**80%**	0%	0%	0%	0%	0%	0%
Chk	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Exp	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Algn	–	–	–	–	–	–	–	–	–	–	–	–	–	–
St-y	20%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	**80%**	0%	0%
St-n	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	**100%**	0%
St	–	–	–	–	–	–	–	–	–	–	–	–	–	–

表 II：DA分类混淆矩阵。含有 – 的单元格表示评估数据集中不存在此类DA标签，且模型未预测出任何此类对话行为。

	无行为	给出 O_T	给出 L	给出 O_T, L	Ack	Yes	No
无行为	**74.08%**	0%	0%	0%	0%	25.92%	0%
给予 O_T	8.82%	**69.12%**	0%	0%	0%	22.06%	0%
给出 L	10%	0%	**83.33%**	0%	0%	6.67%	0%
给出 O_T, L	–	–	–	–	–	–	–
Ack	0%	0%	0%	0%	**66.67%**	33.33%	0%
Yes	0%	3.12%	0%	0%	0%	**96.88%**	0%
No	0%	11.11%	0%	0%	0%	0%	**88.89%**

表 III：动作分类混淆矩阵。带有 – 的单元格表示评估数据集中没有此类动作标签，且模型未预测出任何此类动作。

连续动作，模型将分别输出 *0* 和 *None*，作为预测的 ELD 动作和对话行为。

表 II 中的对话行为混淆矩阵显示，主要的混淆是将“指令”对话行为错误分类为“否定回复”。造成这种混淆的一个原因是数据量有限，此外还存在类别不平衡。然而，查看标注数据后，我们发现，ELD 在回答核实问题或 O_T/L 查询（分别对应“是非/检查查询”和“疑问词查询”对话行为）时，大多数情况下会继续给出进一步的指令。这意味着，在许多情况下，ELD 的话语虽然被标记为“否定回复”，但也可以被解读为“指令”，最终向 HEL 传达相同的意图。例如，HEL 通过询问 ELD“你是说一口锅吗？”来核实 O_T，ELD 则回答“不是，给我拿个碗。”ELD 的对话行为可以标记为“否定回复”或“指令”，但重要的是，HEL 从 ELD 那里接收到的意图相同。

查看表 II 中的混淆矩阵，我们发现其余对话行为类别要么分类效果很好，要么与不会影响网络总体结果的类别混淆。例如，“确认”这一对话行为在 66.67% 的情况下被正确分类，在 33.33% 的情况下被分类为“肯定陈述”。不过，这种误分类不会影响 HEL 接收到的信息。

对动作混淆的分析（结果见表 III）显示，22.06% 的“给 O_T”样本被错误地分类为“是”。这同样是由于可用数据点数量有限以及类别不平衡造成的。不过，与我们上文的解释类似，这种误分类可能是因为某些情况下，ELD 在回答核实问题后继续给出指令。例如，HEL 通过询问 ELD“你说的是那个橱柜吗？”来核实 L，而 ELD 回复“是的，把餐具拿给我。”。ELD 的动作可以标记为“是”或“给 O_T”。无论哪种情况，HEL 收到的消息都相同。

我们还发现，33.33% 的“确认”类别被误分类为“是”动作。同样，由于这两种动作本质上非常相似，这种误分类不会影响 HEL 接收到的信息。

总之，许多对话行为和动作的错误分类，是因为很难区分“指令”、“肯定回复”、“否定回复”、“肯定陈述”、“否定陈述”等对话行为类别，以及“给予 O_T”、“给予 L”、“确认”、“是”和“否”等动作类别。这是因为在我们现有的数据中，ELD 回应“是非查询”问题时，大多数情况下都会先回答是或否，或表示确认，然后引导 HEL 前往某个位置，和/或提供有关物体的信息。

B. HBATN 模型评估

为进一步考察模型表现的真实性，我们将其表现与此前开发的 HBATN 模型进行比较。GUS 模型的行为是否与 HBATNs 相似至关重要，因为我们的 HBATN 是由人工标注员精心手工构建的，他们的见解源自数据。为了进行这项比较，我们需要了解训练好的模型面对不同变体时的响应

	输入	HBATN 输出	GUS 输出
建立(O_T)	(0, 0, $Inst$)/(0, 0, Qw)	(1, $*$, $Inst$)/(1, $*$, Rw)	(1, 0, Rw)
验证(O_T)	(1, 0, Chk)/(1, 0, Qyn)	($*$, 0, Ry)/($*$, 0, Rn)/(1, 0, $Inst$)/(1, 0, Rw)	(1, 0, Ry)/(1, 1, Rn)/(1, 0, $Inst$)/(1, $*$, Rw)
指定(O_T)	($*$, 0, Qw)	($*$, 0, $Inst$)/($*$, 0, Rw)	($*$, 0, Rw)

表 IV：Det(O_T) 子任务的 GUS 性能评估。($*$) 表示 0 或 1。

	输入	HBATN 输出	GUS 输出
建立(L)	($*$, 0, $Inst$)/($*$, 0, Qw)	($*$, $*$, $Inst$)/($*$, $*$, Rw)	($*$, $*$, $Inst$)/($*$, $*$, Rw)/($*$, $*$, Ry)/($*$, $*$, Rn)
验证(L)	(0, 0, –)/(0, $*$, Chk)/ (0, $*$, Qyn)	(0, $*$, Ry)/(0, $*$, Rn)/($*$, $*$, –)/ ($*$, $*$, Rw)/($*$, $*$, $Inst$)	(0, $*$, Sty)/(0, $*$, Ry)/(0, $*$, Rn)
指定(L)	($*$, 1, Qw)	($*$, $*$, $Inst$)/($*$, $*$, Rw)	(0, $*$, $Inst$)/(0, $*$, Rn)/($*$, $*$, –)

表 V：GUS 在 Det(L) 子任务上的性能评估。($*$) 表示 0 或 1。

	输入	HBATN 输出	GUS 输出
指定(O_T)	($*$, 0, Qw)	($*$, 0, $Inst$)/($*$, 0, Rw)	($*$, $*$, $Inst$)/($*$, $*$, Rw)/($*$, 0, Ry)/($*$, $*$, Rn)
验证(O)	($*$, 0, –)/($*$, 0, Chk)/ ($*$, 0, Qyn)/($*$, $*$, St)	($*$, 0, Ry)/($*$, 0, Rn)/($*$, 0, Rw)/($*$, 0, $Inst$)	($*$, 0, Ry)/($*$, 0, Rn)/($*$, $*$, $Inst$)
Finish(L)	($*$, $*$, Sty)($*$, $*$, St)/ ($*$, $*$, Stn)	(0, 0, Ack)	(0, 0, Ry)/(0, 0, Sty)

表 VI：GUS 对 Det(O) 子任务的性能评估。($*$) 表示 0 或 1。

输入。

为了为 GUS 模型生成不同且有意义的输入，我们采用了一种自动化方法。在解释该方法之前，我们需要先明确以下几条基本规则：

- 对于 ELD 的状态表示，在 ELD 说出 O_T 之前，ELD 对 HEL 的知识中 O_T 的信念不能从 0 变为 1 或 2。
- 在 ELD 说出 L之前，ELD 对 HEL 的知识的信念不能从 0 变为 1 或 2。
- 在 ELD 对 HEL 关于 O_T 的知识的信念变为 1 之前，ELD 对 HEL 的知识的信念不能从 0 变为 1 或 2。
- 在 ELD 对 HEL 关于L的知识的信念变为 1 之前，ELD 对 HEL 的知识的信念可以从 0 变为 1 或 2。
- 在 ELD 宣布 O_T 和 L 之前，HEL 无法验证它们。
- 对于$O_T/L/O$ 的验证，HEL 只执行指点和 H-O 动作。

与 ELD 状态表示相关的基本规则会产生十三种不同且有意义的组合，我们已在 IV-A 节中详细说明。将这些状态与其他输入组合后，再将其余基本规则应用于所有组合，我们便能自动生成所有有意义的输入。

随后，我们将 GUS 模型置于先前从 HBATN [2] 中提取的不同状态，通过向模型输入不同内容，最后将 GUS 模型的输出与 HBATN 的输出进行比较。

将我们的 GUS 模型与 HBATN 进行比较的结果汇总于表 IV、V、VI。在表示输入或输出的 (a, b, c) 元组中，c 表示 HEL 对话行为（输入）或 ELD 对话行为（输出），a 和 b 还决定了是否分别说出了 O_T 和 L 。每个输入/输出还包括 ELD 的前一状态、“指点/H-O”动作、ELD 的前一对话行为和动作，以及 HEL 的对话行为和动作特征。为简洁起见，我们在表格中省略了这些参数。不过，对于这些比较，我们已谨慎地将自动生成的输入映射到不同的 HBATN 状态。

比较结果表明，GUS 的输出与 HBATNs 高度吻合。两者只有少数细微差异，不会影响 ELD 与 HEL 之间的交互及其意图。例如，在某些情况下（如表 IV 中的原始子任务“建立(O_T)”），我们的 GUS 模型会混淆“指令”和“带信息请求的回复”这两种对话行为：对于所有标记为“指示/带信息请求的回复”的输出，它都只输出“带信息请求的回复”。对 HEL 而言，话语被标记为其中任一种都没有影响，因为这两种对话行为传达的信息相同。

在其他一些情况下，例如表 V 中的原始子任务“指定(O_T)”，我们的 GUS 模型会针对标记为“指令”的输出，给出“指示/肯定回复/否定回复”这几种对话行为。这同样不是致命错误，因为 GUS 模型输出为 ELD 动作的动作会就 O_T 或 L 提供指示。在表 VI 的原始子任务“Finish(L)”中，GUS 模型会混淆“确认”这一对话行为与“肯定回复/肯定状态”。由于这两种对话行为本身具有相似之处，这种差异也可忽略不计。尽管如此，这是交互中的最后一个动作，因此完全不会影响交互。

还应指出，在某些情况下，当 HBATNs 中的 ELD 说出或未说出 O_T 或 L 时，我们的 GUS 模型会有所不同。在少数情况下，当 HBATNs 中的 ELD 未说出 O_T 或 L 时，GUS 模型会说出它们，反之亦然。这一小误差源于我们采用机器学习方法来构建作为用户模拟器的人工智能代理。尽管这些小误差在极少数情况下可能会使交互时间变长或导致交互失败，但我们的 GUS

模型仍优于手工设计框架。读者可参阅我们先前的研究，其中对用户模拟器与强化学习 HEL 智能体之间的交互进行了详细阐述 [4]，该研究表明，我们的用户模拟器可有效用于辅助机器人的强化学习训练。

VI. 结论

本文介绍了一种新型用户模拟器，该模拟器利用神经网络，为强化学习智能体提供一个多样化的交互环境，使其能够在采用多种交流模式的协作任务中接受训练。该用户模拟器基于 ELDERLY-AT-HOME 交互语料库中的“Find”任务示范构建。它能够理解并回应多种模态，包括语言、指向手势和触觉示意动作。本研究的主要贡献是创建了一种可为强化学习训练提供多模态交互环境的用户模拟器，此前尚无此类工作。第二项贡献是一种新颖的多模态数据增强方法，有效克服了利用有限且稀疏的人类示范开发智能数据驱动型智能体所面临的挑战。我们对所开发的模拟器进行了评估，以测试其对训练数据的表征能力，并将其性能与人工构建的 HBATN 模型进行比较。结果表明，该模拟器性能良好，说明它可成功用于家用辅助机器人的强化学习训练。

参考文献

[1] B. Abbasi、N. Monaikul、Z. Rysbek、B. Di Eugenio 和 M. ˇ Zefran，《面向辅助机器人的多模态人机交互管理器》，载于*2019*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*）。IEEE，2019年，第6756–6762页。[2] N. Monaikul、B. Abbasi、Z. Rysbek、B. Di Eugenio 和 M. ˇZefran，《面向任务的多模态人机协作中的角色切换》，载于*2020*年第*29*届*IEEE*机器人与人类交互通信国际会议（*RO-MAN*）。IEEE，2020年，第1150–1156页。[3] L. Chen、M. Javaid、B. Di Eugenio 和 M. ˇ Zefran，《协作式多模态人际对话中触觉示意动作的作用与识别》，*Computer Speech & Language*，第34卷，第1期，第201–231页，2015年。[4] A. M. Shervedani、S. Li、N. Monaikul、B. Abbasi、B. Di Eugenio 和 M. Zefran，《面向与人类协作的机器人的多模态强化学习》，*arXiv*预印本 *arXiv:2303.07265*，2023年。[5] I. Bratko、T. Urban ˇci ˇc 和 C. Sammut，《行为克隆：现象、结果与问题》，*IFAC*会议论文集，第28卷，第21期，第143–149页，1995年。[6] F. Torabi、G. Warnell 和 P. Stone，《从观察中进行行为克隆》，载于第*27*届人工智能国际联合会议论文集，2018年，第4950–4957页。[7] M. A. Goodrich、A. C. Schultz 等，《人机交互：综述》，*Foundations and Trends® in Human–Computer Interaction*，第1卷，第3期，第203–275页，2008年。[8] L. Peternel、W. Kim、J. Babic 和 A. Ajoudani，《迈向符合人体工学的人机协同操作与交接控制》，载于*2017*年*IEEE-RAS*第*17*届人形机器人国际会议（*Humanoids*）。伯明翰：IEEE，2017年11月，第55–60页。[9] Z. Erickson、V. Gangaram、A. Kapusta、C. K. Liu 和 C. C. Kemp，《Assistive Gym：辅助机器人技术的物理仿真框架》，载于*2020*年*IEEE*机器人与自动化国际会议（*ICRA*）。IEEE，2020年，第10 169–10 176页。[10] S. Nikolaidis、J. Forlizzi、D. Hsu、J. Shah 和 S. Srinivasa，《人机协作中适应的数学模型》，*arXiv*预印本 *arXiv:1707.02586*，2017年。

[11] X. Wu、L. Xiao、Y. Sun、J. Zhang、T. Ma 和 L. He，《人在回路机器学习综述》，*Future Generation Computer Systems*，2022年。[12] A.-N. Sharkawy，《人机交互：应用》，载于*IFSA*第四次工业革命自动化、机器人与通信冬季会议（*ARCI'2021*），2021年，第98–103页。[13] C. M. Ranieri、G. V. Nardari、A. H. M. Pinto、D. C. Tozadore 和 R. A. F. Romero，《Lara：用于室内环境人机交互的机器人框架》，载于*2018*年拉丁美洲机器人研讨会、*2018*年巴西机器人学研讨会（*SBR*）和*2018*年教育机器人研讨会（*WRE*）。IEEE，2018年，第376–382页。[14] C. Chao 和 A. L. Thomaz，《多模态轮流互动中的时序：使用定时佩特里网进行控制与分析》，*Journal of Human-Robot Interaction*，第1卷，第1期，第4–25页，2012年。[15] L. El Asri、J. He 和 K. Suleman，《用于口语对话系统用户模拟的序列到序列模型》，*Interspeech 2016*，第1151–1155页，2016年。[16] F. Kreyssig、I. Casanueva、P. Budzianowski 和 M. Gasic，《用于口语对话系统基于语料库的策略优化的神经用户模拟》，载于第*19*届*SIGdial*话语与对话年会论文集，2018年，第60–69页。[17] X. Li、Y.-N. Chen、L. Li、J. Gao 和 A. Celikyilmaz，《端到端任务完成型神经对话系统》，载于第八届自然语言处理国际联合会议论文集（第*1*卷：长篇论文），2017年，第733–743页。[18] R. Choudhury、G. Swamy、D. Hadfield-Menell 和 A. D. Dragan，《论模型学习在人机交互中的效用》，载于*2019*年第*14*届*ACM/IEEE*人机交互国际会议（*HRI*）。IEEE，2019年，第317–325页。[19] A. Paszke、S. Gross、F. Massa、A. Lerer、J. Bradbury、G. Chanan、T. Killeen、Z. Lin、N. Gimelshein、L. Antiga、A. Desmaison、A. Kopf、E. Yang、Z. DeVito、M. Raison、A. Tejani、S. Chilamkurthy、B. Steiner、L. Fang、J. Bai 和 S. Chintala，《PyTorch：一种命令式风格的高性能深度学习库》，载于《神经信息处理系统进展》第*32*卷。Curran Associates, Inc.，2019年，第8024–8035页。 [在线]。获取地址：http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf