标准深度强化学习方法的样本效率低下，使得它们无法应用于许多实际问题。利用人体演示的方法需要较少的样本，但研究较少。正如计算机视觉和自然语言处理社区所证明的那样，大规模数据集可以作为新方法的实验和基准平台，促进研究。然而，与强化学习模拟器兼容的现有数据集没有足够的规模、结构和质量，无法进一步开发和评估侧重于使用人类示例的方法。因此，我们介绍了一个全面的、大规模的、模拟的人体演示数据集：miner。这个数据集由超过6000万个自动标注的状态-动作对组成，它们跨越了Minecraft这个动态的、3D的、开放世界环境中的各种相关任务。我们提出了一种新的数据收集方案，允许不断引入新的任务和收集适合各种方法的完整状态信息。我们展示了miner数据集的层次性、多样性和规模。此外，我们展示了Minecraft领域的困难，以及Miner在开发技术以解决其关键研究挑战方面的潜力。

随着深度强化学习（DRL）方法应用于越来越困难的问题，用于培训的样本数量也在增加。例如，Atari 2600游戏【Bellemare等人，2013年】被用于评估DQN【Mnih等人，2015年】、A3C【Mnih等人，2016年】和Rainbow DQN，每种游戏都需要44到2亿帧（200到900小时以上）才能达到人类水平的性能【Hessel等人，2018年】。在更复杂的领域：OpenAI Five利用了11000年以上的Dota 2游戏玩法【OpenAI，2018年】；AlphaGoZero在围棋中使用490万个自玩游戏【Silver等人，2017年】；AlphaStar使用了200年的星际争霸II游戏【DeepMind，2018年】。

这种固有的样本效率妨碍了标准DRL方法在不利用数据扩充技术的情况下应用于现实世界的问题[Tobin等人，2017年]，[Andrychowicz等人，2018年]、域对齐方法[Wang等人，2018年]，或者仔细设计真实世界的环境，以满足所需数量的试验【Levine等人，2018年】。最近，利用轨迹示例的技术，如模仿学习和贝叶斯强化学习方法，已经成功地应用于较旧的基准测试和实际问题中，其中来自环境的样本是昂贵的。

然而，对于一大类复杂的现实领域，这些技术仍然不够有效。

正如[Kurin et al.，2017]所指出的，数据集的引入和高效的大规模数据收集方案（如Switchboard[Godfrey et al.，1992]和ImageNet[Deng et al.，2009]）的引入，促进了机器学习的几个子领域。虽然强化学习社区已经创建了广泛的基准模拟器，但目前缺乏针对具有广泛结构约束和任务的领域的大规模标记数据集。

因此，我们介绍MineRL，一个大型的数据集，包含了超过6000万个国家行动对人类在Minecraft的一系列相关任务中的演示。为了在Minecraft中捕捉游戏和玩家交互的多样性，MineRL包含了六个具有各种研究挑战的任务，包括开放世界多智能体交互、长期规划、视觉、控制和导航，以及显式和隐式子任务层次。我们在现有的雷击模拟器中提供了这些任务的顺序决策环境的实现。此外，我们还介绍了一个新的平台和方法，以继续收集人类在布雷克雷夫特演示。当用户在我们的公共游戏服务器上玩游戏时，我们会记录数据包级别的信息，这样可以完美地重建每个玩家的视图和动作。该平台能够向MineRL数据集添加新任务并自动注释，以补充当前和未来应用于Minecraft的方法。有关数据集的演示视频和更多详细信息，请访问<http://miner.io>。

Minecraft是基于强化和模仿学习方法发展的一个引人注目的领域，因为它提出了独特的挑战：Minecraft是一个3D、第一人称、开放的世界游戏，围绕资源的收集和结构和项目的创建。它可以在单人模式或多人模式下播放，所有玩家都存在于同一个世界并影响着同一个世界。每个玩家都要进行多次游戏，总共持续数十小时。值得注意的是，按程序生成的世界是由允许修改的离散块组成的；在游戏过程中，玩家通过收集资源（如树上的木材）和建造结构（如庇护所和仓库）来改变周围环境。

作为一个开放的世界游戏，Minecraft没有单一的目标。相反，玩家开发他们自己的子目标，这些子目标形成了许多自然的等级。尽管这些等级可以被利用，但它们的规模和复杂性导致了Minecraft固有的困难。一个这样的层次结构是物品收集：对于地雷阵中的大量目标，玩家必须创建特定的工具、材料和物品，这些物品需要收集一组严格的必备物品。这些依赖项的聚合形成了一个大规模的任务层次结构（见图2）。

除了获取项目，隐式层次结构通过游戏的其他方面出现。例如，玩家（1）建造结构，为自己和储存的资源提供安全，以防自然发生的敌人；（2）探索世界寻找自然资源，经常与非玩家角色进行战斗。这两个游戏元素都有很长的时间范围，并且由于环境相关的需求（例如耕种生存所需的某种资源，使探索能够收集另一种资源，等等）而呈现出灵活的层次结构。

分类和自然语言数据集从Mechanical Turk等数据收集平台的存在中受益匪浅，但相比之下，游戏数据的收集通常需要为每个游戏实施新的平台和用户获取方案。为此，我们引入了第一个端到端平台，用于收集Minecraft中的玩家轨迹，从而能够构建MineRL-v0数据集。如图1所示，我们的平台包括（1）一个公共游戏服务器和一个网站，在这里我们可以记录自然游戏中Minecraft玩家的轨迹；（2）一个定制的Minecraft客户端插件，它记录客户端和服务器之间的所有包级通信，因此，我们可以通过修改游戏状态和图形来重新模拟和渲染人类演示；以及（3）一个数据处理管道，它使我们能够生成任务演示的自动注释数据集。

数据采集。Minecraft玩家在标准Minecraft服务器列表中找到miner服务器。玩家首先使用我们的网页提供IRB同意匿名记录他们的游戏体验。然后他们为他们的Minecraft客户端下载一个插件，该插件将用户的客户端服务器游戏包记录并流式传输到miner数据存储库。当在我们的服务器上玩游戏时，用户选择一个独立的任务来完成，并根据获得的奖励金额来获得游戏币。对于生存游戏模式（在没有已知的奖励功能的情况下），玩家只在游戏持续时间内获得奖励，这样就不会产生艺术奖励功能。我们在Malmo中实现这些独立任务。

数据管道。我们的数据管道能够持续扩展miner数据集发布所伴随的结构化信息；它允许我们重新计算、修改和扩充记录的轨迹，使之成为几种算法上可使用的格式。该管道作为核心Minecraft游戏代码的扩展，并使用我们的自定义API将每个记录的数据包从Miner数据存储库同步重新发送到Minecraft客户端，以进行自动注释和游戏状态修改。这个API允许我们根据游戏状态的任何方面添加注释，这些注释可以从现有的Minecraft模拟器访问。

可扩展性。我们的目标是使用我们的平台来提供一个详尽而广泛的多任务数据集（beyond miner-v0），与RL环境相结合，跨越自然语言、具体化推理、分层规划和多智能体协作。服务器的模块化设计允许我们为越来越多的独立任务获取数据。此外，游戏中的经济和服务器社区从用户群中创建了一致的参与度，使我们能够以不断增长的速度收集数据，而不会产生额外的成本。数据管道的模块化、模拟器兼容性和可配置性也允许创建新的数据集，以补充利用人类演示的新技术。例如，可以通过重复渲染具有不同约束条件的数据来进行大规模的泛化研究：改变照明、相机位置（具体和非体现）以及其他视频渲染条件；在观察、奖励和行动中注入人工噪声；和游戏层次重组（交换游戏项目的功能和语义）。

在本节中，我们将介绍和分析miner-v0数据集。我们首先给出数据集的详细信息，包括其大小、形式和包装。然后，我们通过详细描述包含的任务系列，然后分析数据质量、覆盖率和层次结构，来说明这个初始版本的广泛适用性。为了说明miner-v0数据集的有用性，在第5节中，我们展示了我们的任务在开箱即用方法方面的困难，以及通过使用miner-v0的基本模拟学习技术实现的性能提升。

4.1数据集详细信息

大小。MineRL-v0数据集包括500多小时的人类演示，这些演示来自数据采集平台，涉及六个不同的任务。发布的数据由四个不同版本的数据集组成，这些数据集具有不同的分辨率（64×64和192×256）和纹理（默认的Minecraft和Simplified）。对于低分辨率和中等分辨率的数据集，每个版本各自总计超过6000万个状态操作对，大小分别为130 GB和734 GB。

形式。每个轨迹都是一组连续的状态-动作对，对每一个Minecraft游戏滴答声（每秒20个游戏滴答声）进行采样。每个状态都由玩家视角的RGB视频帧和游戏状态的一整套功能组成：玩家库存、物品收集事件、目标距离、玩家属性（健康、等级、成就），以及玩家当前打开的GUI的详细信息。在每个滴答声中记录的动作包括：客户端上的所有键盘按下、视图俯仰和偏航的变化（由鼠标移动引起）、所有玩家GUI单击和交互事件、发送的聊天消息以及聚合动作（如项目制作）。

附加注释。人类的轨迹伴随着大量自动生成的注释。对于所有的独立任务，我们会记录指示演示质量的指标，例如时间戳奖励、无操作数、死亡数和总分。此外，轨迹元数据包括用于分层标记的时间戳标记；例如，当一个类似房子的结构被建造或某些目标（如砍倒一棵树）得到满足时。

包装。数据集的每个版本都打包为Zip存档，每个任务族有一个文件夹，每个演示有一个子文件夹。在每个轨迹文件夹中，状态和动作都存储为玩家POV的H.264压缩MP4视频，最大比特率为18Mb/s，JSON文件包含游戏状态的所有非视觉特征，以及与视频每一帧对应的玩家动作。此外，对于特定的任务配置（操作和状态空间的简化），我们提供了由矢量形式的状态-动作-奖励元组组成的Numpy.npz文件，从而提高了数据集的可访问性。打包数据和随附文档可从<http://miner.io>。

4.2任务

最初的miner-v0数据集包括六个独立任务，这些任务被选择来代表Minecraft的不同方面，这些任务反映了该领域广泛研究的挑战：层次性、长期规划和复杂的定向运动。在所有任务中，代理可以访问与人类玩家相同的一组操作和观察结果，如第4.1节所述，所有任务都有时间限制，这是观察的一部分。每个任务的详细信息如下。

导航。在导航任务中，智能体必须移动到程序生成的非凸地形上的随机目标位置，这些地形具有可变的材质类型和几何体。这是一个子任务的许多任务贯穿整个地雷。除了标准观测之外，代理还可以访问“指南针”观测，它指向距起始位置64个街区（米）的设定位置。目标与此位置有一个小的随机水平偏移，可能略低于表面水平，因此代理必须根据视觉特征搜索最终目标。提供的奖励函数有两种变体：稀疏（到达目标时为+1，此时事件终止）和密集（奖励与朝目标移动的距离成比例）。

砍树。Treechop任务复制获取木材以生产更多物品。木材是采矿工艺中的关键资源，因为它是所有工具的先决条件（如图2和图6中木棒的放置所示）。特工从森林生物群落（许多树附近）开始用铁斧砍树。每获得一个单位的木头，就会给他+1的奖励，当他获得64个单位时，这一集就结束了。

获取物品。我们包括四个相关的任务，这些任务要求代理在项目层次结构中进一步获取项目：获取铁镐、获取钻石、获取CookedMeat和获得bed。代理总是在一个没有任何物品的随机位置开始；这符合地雷阵中人类玩家的启动条件。不同的任务变体对应不同的常用物品：铁镐、钻石、熟肉（四种变体，每种动物来源一种）和床（三种变体，每种染料需要一种）。铁镐是获取关键材料所需的工具。钻石是高级地雷游戏的核心，大部分游戏都围绕着钻石的发现。熟肉用来补充体力，睡觉需要床。这些物品共同代表了玩家生存和进入游戏的更多区域所需要的东西。代理人获得所需物品将获得+1奖励，此时事件结束。

生存。除了特殊设计任务的数据外，我们还提供生存数据，这是大多数玩家使用的标准开放式游戏模式。从一个没有任何物品的随机地点开始，玩家制定自己的高级目标并获得物品来完成这些目标。此任务的数据可用于学习人类在开放游戏中遵循的复杂奖励函数和相应的政策。该数据还可用于培训试图完成其他结构化任务的代理，或进一步提取政策草图，如[Andreas等人，2017]。

4.3分析

人的表现数据集中的大多数人类演示完全属于专家级的游戏。图4显示了完成每个独立任务所需的时间分布。每个柱状图中的红色区域表示与专家级游戏相对应的时间范围，计算为至少具有五年采矿经验的玩家完成任务所需的平均时间。大量的专家样本和丰富的演示性能标签使许多标准的模拟学习技术得以应用，这些技术假设了基本策略的最优性。此外，初级和中级的轨迹允许进一步发展利用不完美演示的技术。

覆盖Miner-v0几乎完全覆盖了雷艇。在生存游戏模式下，获得不同物品的371个子任务中的大部分已经被玩家演示了数百到上万次。此外，其中一些子任务需要数小时才能完成，需要长时间的挖掘、建造、探索和打击敌人。由于大量的任务级注释，数据集可用于大规模的选项提取和技能获取，从而扩展了[Shu et al.，2017]和[Andreas et al.，2017]的工作。此外，获取<Item>任务的丰富标签层次结构可用于构建提取选项的可解释性和质量度量。

除了项目覆盖范围外，Miner数据收集平台的结构旨在促进游戏条件的广泛表示。当前的数据集包括从1002个独特的播放器会话中提取的不同演示集。在生存游戏模式下，记录的轨迹总共覆盖24393057平方米的游戏内容，其中一平方米对应于一个布雷克雷格区块。对于所有其他任务，每个演示都发生在一个随机初始化的游戏世界中，因此我们为每个任务收集大量不同的独特轨迹：在图5中，我们显示了玩家在完成每个任务的过程中的自上而下的位置，其中起始状态为（0，0）。不仅每个玩家在不同的游戏世界中行动，而且每个玩家在每个任务中都会探索一个大的区域。

层次性如图2所示的项目图所示，Minecraft具有很深的层次结构，MineRL数据收集平台设计用于显式和隐式地捕获这些层次结构。作为一个主要的例子，获取<Item>独立任务隔离了项目层次结构中困难但重叠的核心路径。由于miner-v0中提供了子任务标签，我们可以检查和量化这些任务的重叠程度。

层次性的直接度量是通过项目优先频率图（item precedence frequency graphs）得到的，其中节点对应于任务中获得的项，有向边对应于玩家在目标节点项之前获得源节点项的次数。

这些图表提供了人类元策略的统计视图，以及他们的子策略在任务之间转移的程度。图6显示了由Miner轨迹构造的优先频率图，这些轨迹在获得Diamond、AcquitaineCookedMeat和AcquitainIronPickAxe任务上构建。检查发现，获得钻石的策略包括获得木材、火炬和铁矿石的子策略。所有这些都是获取铁镐任务所必需的，但只有一部分是在获得烤肉任务中使用的。这些重叠子策略的效果可以在图5中看到：玩家在具有重叠层次结构的任务中移动相似（例如获得铁镐和获得钻石），在重叠较少的任务中移动不同。此外，这些图描绘了任务中人类元策略的分布图：尽管存在必要的图遍历模式（如木材→石镐），但取决于具体情况，玩家通过获取通常在项目优先图中稍后发现的项目（当先前的项目不可用时）通过较长的路径来调整策略。这反过来又使miner-v0能够用于开发分布式分层强化学习方法。

5个实验

5.1实验配置

为了展示Minecraft的困难，我们评估了三种强化学习方法和一种行为克隆方法在我们最简单的任务（Treechop和Navigate（稀疏））上的表现，以及一个简单的任务（额外的、成形的奖励、导航（密集的）。具体而言，我们评估（1）双深度双Q网络（DQN）[Mnih et al.，2015]，一种基于Qlearning的关闭策略方法；（2）预训练DQN（PreDQN），DQN具有额外的预训练步骤，并通过Miner-v0的专家演示初始化重放缓冲区；（3）Advantage Actor Critic（A2C）[Mnih et al.，2016]，一种on策略，策略梯度法；和（4）行为克隆（BC），一种使用标准分类技术从演示中学习策略的方法。为了确保这些方法的可重复性和精确评估，我们在OpenAI基线实现的基础上构建[Dhariwal等人，2017]。

观测值将转换为灰度，并调整为64x64的大小。由于Minecraft中有数千个动作组合，加上基线算法的局限性，我们将动作空间简化为10个离散动作。但是，行为克隆没有这样的限制，并且在没有操作空间简化的情况下执行类似的操作。为了使用预先训练的DQN和行为克隆的人类演示，我们用10个动作原语中的一个来近似每个动作。我们训练了1500集（约1200万帧）的强化学习方法。为了训练行为克隆，我们使用来自每个任务族的专家轨迹并进行训练，直到策略性能达到最大。

5.2评估和讨论

我们通过在训练期间获得的100集窗口的最高平均奖励来比较算法。我们还报告了随机策略的性能和第50百分位人类绩效。结果汇总在表1中。

在所有任务中，学习型代理的表现都比人类表现差得多。Treechop表现出最大的差异：人类的得分为64分，而增强剂的得分低于4分。这表明我们的任务非常困难，尤其是考虑到获取<Item>任务是在Treechop任务的基础上建立起来的，需要完成几个额外的子目标（≥3）。我们的假设来自于一个长期的环境问题。例如，特工很难学会在水中航行，因为在溺水死亡之前，特工要经历多次转换。

鉴于这些困难，我们的数据有助于提高性能和样本效率：在所有任务中，利用人工数据的方法表现更好。如图7所示，专家演示能够获得更高的每集回报，并通过较少的样本获得高性能。专家演示在随机探索不太可能产生任何回报的环境中特别有用，比如导航（稀疏）。

6

相关工作

许多领域以前已经通过模仿学习和人类演示数据集得到解决。其中包括使用Atari Grand Challenge数据集的Atari域【Kurin等人，2017年】，以及使用按需数据集的Super Tux Kart域【Ross等人，2011年】。与Minecraft不同，它们是简单的域：它们具有浅层次的依赖关系，并且不是开放的世界。由于动作空间和状态空间较小，这些领域已经通过使用相对较少的样本（在[Kurin et al.，2017]中五个游戏中的970万帧和[Ross et al.，2011]中的2万帧）来解决。相比之下，我们提供了6000万个自动注释的状态-动作对，并且没有达到人的性能。

针对具有挑战性、未解决的领域的现有数据集主要用于现实世界中缺少模拟器限制开发速度的任务。例如，KITTI数据集【Geiger等人，2013年】，是一个包含3小时真实交通的3D信息的数据集。同样，Dex Net【Mahler等人，2019年】是一个500万个抓取数据集，其中包含用于机器人操作的相应3D点云。与这些数据集不同，miner与模拟器Malmo直接兼容，因此允许在收集数据的同一领域进行训练，并与不基于模拟学习的方法进行比较。此外，相对于领域难度，miner的规模比KITTI和Dex Net数据集更大。

唯一一个复杂的，尚未解决的领域与现有的模拟器和大规模的数据集是星际争霸2。然而，星际争霸2不是开放的世界，所以不能用来评估3D环境中为具体任务设计的方法。目前最大的数据集是StarData【Lin等人，2017年】。与miner不同，它由标准游戏的未标记、提取的轨迹组成。相比之下，miner包含了越来越多的相关任务，它们代表了整个Minecraft任务层次结构的不同组件。此外，miner还包含丰富的自动生成的注释，包括子任务完成、玩家技能级别和扩展这些标签的API。这些属性一起允许使用和评估利用层次结构的方法。

7

结论和今后的工作

MineRL-v0目前在一个开放世界、模拟器配对的环境中提供了6000万个状态-动作对的程序化注释人类演示。它目前包含六个任务的数据，没有一个可以用标准的深度强化学习方法完全解决。我们的平台允许对现有任务和新任务进行持续的演示收集。因此，我们在社区可访问的网站上托管MineRL-v0，http://miner.io，并将收集有关向miner添加新注释和任务的反馈。随着我们扩展miner，我们期望它在一系列的方法中越来越有用，包括反向强化学习、分层学习和终身学习。我们希望miner将成为顺序决策研究的中心资源，支持人工智能的许多分支朝着开发能够解决更广泛的现实世界环境的方法的共同目标迈进。