arXiv:2303.08409v1 [cs.CV] 15 Mar 2023

LANA：用于指令遵循与生成的具备语言能力的导航器

Xiaohan Wang Wenguan Wang JiayiShao Yi Yang*
ReLER, CCAI,浙江大学
https://github.com/wxh1996/LANA-VLN

摘要

近年来，视觉语言导航（*VLN*）——要求机器人智能体遵循导航指令——取得了长足进展。然而，现有研究大多侧重于将指令解读为动作，最终只能构建出“愚笨的”寻路智能体。本文提出**LANA**，一种具备语言能力的导航智能体，不仅能够执行人类编写的导航命令，还能为人类提供路线描述。为此，我们仅用一个单一模型，同时学习指令遵循与指令生成。具体而言，我们构建了两个编码器，分别用于路线编码和语言编码，并由两个解码器共享，分别用于动作预测和指令生成，从而利用跨任务知识并捕捉特定任务的特点。在预训练和微调过程中，指令遵循与指令生成均被设为优化目标。通过实证研究，我们验证了与近期先进的特定任务解决方案相比，LANA 在指令遵循和路线描述两项任务上均表现更优，且模型复杂度几乎减半。此外，凭借语言生成能力，LANA 能够向人类解释自身行为，并协助人类寻路。本研究有望推动未来工作，助力构建更值得信任且更具社会智能的导航机器人。

1. 引言

能够用自然语言与人类互动，同时感知并在环境中采取动作的智能体，是人工智能的基础目标之一。作为迈向这一目标的一小步，视觉语言导航（VLN）[4]——赋予智能体执行自然语言导航指令的能力——近来受到了广泛关注。在视觉语言导航领域，已有大量工作致力于语言落地——教会智能体将人类指令与感知相关的动作联系起来。然而，对于其反向任务，相关研究要少得多 [27, 71, 1, 78, 23]——语言生成——教会智能体如何

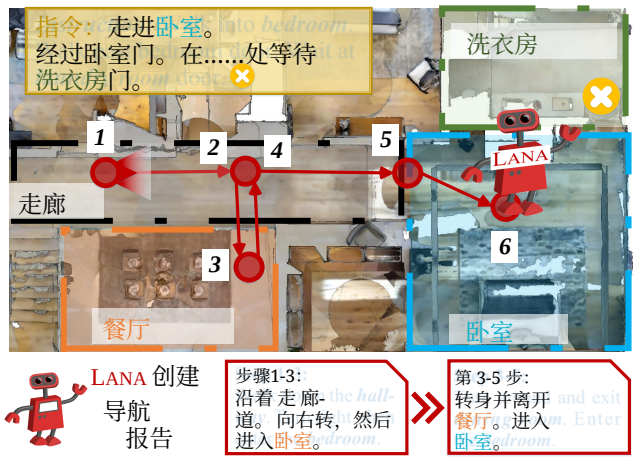

图 1：LANA 既能遵循指令，也能生成指令。它撰写的报告有助于人机协作，并且在一定程度上可以解释自身行为：在第 2 步，它误将餐厅当成卧室，因此采取了错误行动。在第 3 步收集更多信息后，它转向了正确方向。

用语言生动地描述导航路线。更重要的是，现有视觉语言导航研究通常分别训练专门执行单一任务的智能体。因此，训练出的智能体要么擅长寻路却不会说话，要么善于交谈、能够担任路线指导者，却不会行走。

本文强调了视觉语言导航中的一项根本挑战：我们能否训练出一个既能遵循导航指令，又能创建路线描述的单一智能体？

我们提出 LANA，这是一种具备语言能力的导航智能体，能够充分意识到这一挑战（图1）。通过同时学习指令落地和生成，LANA在统一框架中，将通过面向导航的自然语言传达的人类到机器人和机器人到人类通信形式化。这一点至关重要，因为：**i)** 它完善了人类与智能体之间必要的通信闭环，并提升了视觉语言导航智能体在现实世界中的实用性 [59]。例如，当智能体执行导航指令耗时较长时，持续要求人类关注既不可行也不理想，此时智能体应报告其进度 [73]。此外，智能体应能引导人类前往其已探索的区域 [82], , 这对灾区的搜救机器人尤为重要 [72, 19], 进行引导

*通讯作者：Yi Yang。

公共空间机器人 [78], 和视障人士导航设备 [36]。 **ii)** 双向通信是实现紧密人机协调不可或缺的一环（即“我会继续沿这条路走 · · ·”） [7],，并能增强人类对机器人的信任 [6, 24],，从而提高人们对导航机器人的接受度。 **iii)** 培养语言生成能力有助于打造更具可解释性的机器人，使其能够以人类可读的路线描述形式阐释自身的导航行为。

具体而言，LANA 是一个基于 Transformer 的多任务学习框架。该网络包含两个单模态编码器，分别用于语言和路线编码；还包含两个多模态解码器，分别基于这两个编码器执行路线到指令以及指令到路线的转换。整个网络采用端到端方式进行学习，在预训练和微调阶段同时执行指令落地和生成任务。综合这些特点，LANA 提供了一个统一而强大的框架，在模型设计和网络训练的核心环节探索任务特定知识与跨任务知识。因此，即使缺乏明确监督，LANA 也能更好地理解语言线索（例如词、短语和句子）、视觉感知、长时间跨度内的动作及其相互关系，最终同时使这两项任务受益。

我们在三个著名的视觉语言导航数据集上进行了大量实验（即R2R [4], R4R [38], REVERIE [63]），涵盖指令跟随和生成任务，并得到了一些有趣的发现：**第**一，LANA 仅用一个智能体便成功解决了这两项任务，无需在不同模型之间切换。**第**二，凭借优雅且集成化的架构，LANA 的表现与近期领先的任务专用替代方案不相上下，甚至更胜一筹。**第**三，与分别学习每项任务相比，在两个任务上联合训练 LANA 能以更低的复杂度和更小的模型规模取得更好的性能，证实了 LANA 在跨任务关联建模和参数效率方面的优势。**第四**，LANA 能通过口头描述其导航路线，向人类解释自身行为。从本质上说，LANA 可视为一种配备了自适应训练语言解释器的可解释视觉语言导航机器人。**第五**，主观分析表明，我们生成的语言质量高于基线，但仍不及人类生成话语。尽管仍有改进空间，我们的结果为未来视觉语言导航研究揭示了一个前景广阔的方向，并为可解释导航智能体和机器人应用带来了巨大潜力。

2. 相关工作

导航指令跟随。 构建基于语言的自主导航智能体，一直是自然语言处理和机器人学领域长期追求的目标。与以往局限于受控环境的研究不同 [55, 72, 10, 5, 57], Anderson 等. [4]将此类任务拓展至照片级真实环境——视觉语言导航（VLN），激发了计算机视觉领域日益浓厚的兴趣。早期工作建立在循环神经网络之上。它们探索了多种训练策略 [84, 83], 从合成样本中挖掘额外的监督信号 [27, 71, 28] 或采用辅助任务 [83,35, 53, 93, 78], 并探索智能路径规划 [39, 54,81]。为实现结构化的长程上下文建模，近期方案采用了环境地图 [92, 13, 21,80], Transformer架构 [33, 61, 48, 64, 11], 和多模态预训练 [56, 31, 30, 12]。

与现有的视觉语言导航（VLN）解决方案不同，后者全部 专注于导航指令跟随；我们的目标是构建一个强大的智能体，使其不仅能够执行导航指令，还能描述自己的导航路线。从网络设计、模型预训练到微调，我们始终围绕这一目标推进算法开发。通过联合学习指令执行与生成，我们的智能体能够更好地将指令与感知和行动联系起来，并在一定程度上解读自身行为、增进人类信任。我们的目标与视觉对话导航 [73]有所不同（但二者互为补充），后者仅关注智能体使用语言请求人类协助的情形。**导航指令生成**。 指令创建研究 [17] 可追溯至 20 世纪 60 年代 [52]。早期研究[88, 2, 51] 发现，人类的路线指引与认知地图 [42], 相关，并受多种因素影响，例如，文化背景 [74] 和性别 [37]。研究者还达成共识：加入逐步转向指引和显著地标能让人类更容易遵循指令 [50, 77, 67]。基于这些成果，研究者开发了一些计算系统，使用预先构建的模板 [50, 29], 或手工编写的规则 [18]。尽管这些系统在特定场景中能生成高质量输出，但它们需要语言学知识方面的专业能力，而且构建模板或规则也要耗费大量精力。后来出现了一些数据驱动的解决方案 [16, 59, 19, 26]，但它们仍局限于简化的类网格环境或感知能力有限的环境。

生成自然语言指令长期以来一直被视为社会智能机器人的核心功能，也受到机器人学 [29],语言学 [69], 认知 [42, 25], 心理学 [74],和地理科学 [20]等众多学科的广泛关注。令人意外的是，具身视觉领域对此鲜有研究。少数例外 [27, 71, 68,1, 78, 23], [27, 71, 68] 也只是为了增强训练数据以提升寻路能力，而且它们都训练了一个专门用于指令生成的单一智能体。我们的想法从根本上不同。我们旨在构建一个具备语言能力的导航智能体，使其同时精通指令跟随与指令创作。因此，本研究代表了在构建社会智能、具身导航机器人方面一次早期但扎实的尝试。

视觉语言导航中的辅助学习。 有几种视觉语言导航解决方案 [53, 93, 79] 利用辅助任务提供的额外监督信号，来辅助导航策略学习。对于辅助任务，具有代表性的包括下一步朝向回归 [93], 导航进度估计 [53],路径反向翻译 [93, 78], 轨迹-指令兼容性预

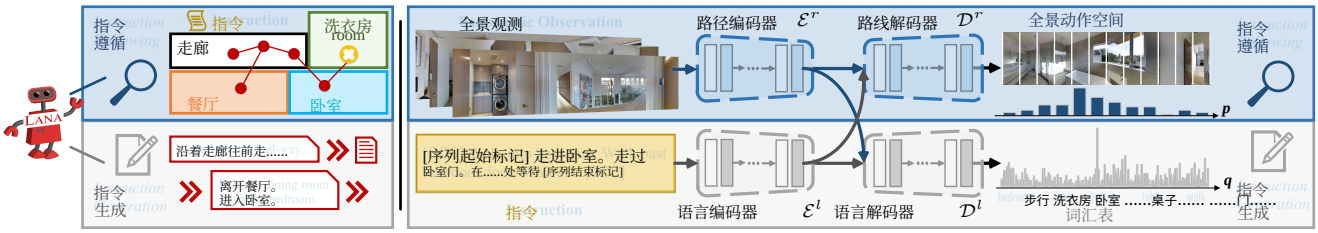

图 2：L 的架构概览ANA (§3.1)。

测 [93], 以及最终目标定位 [92]。

这些视觉语言导航方案侧重于指令跟随；辅助任务只是手段，而非目的。相比之下，我们旨在构建一个能够同时精通指令跟随和指令生成的单一智能体。尽管 [78] 在双任务学习方案下同等重视指令跟随和指令生成，但它仍然训练了两个彼此独立的单任务智能体。此外，原则上，前述辅助任务也可以由我们的智能体使用。

面向视觉语言导航的视觉语言预训练。基于大规模图文对的视觉语言预训练近年来取得了快速进展。研究已证明，此类预训练能够生成可迁移的跨模态表征，并促进下游任务的发展 [85, 47, 91,65, 44, 46]。这种训练范式在视觉语言导航领域日益普及。具体而言，早期的一些研究 [45, 33]直接将通用视觉语言预训练应用于视觉语言导航，而未考虑任务特性。后来， [30, 31, 12]利用大量网络图像描述 [30]或合成的轨迹-指令对 [31, 12]，通过不同的视觉语言导航专属代理任务开展预训练。 [11, 64]则引入了考虑历史信息的代理任务，以实现更加贴合视觉语言导航的预训练。

从代理任务的角度看，现有的视觉语言导航预训练采用掩码语言建模 范式 [40]。不同的是，我们的预训练基于语言生成，这有助于智能体捕捉语言结构，从而全面理解语言指令并提升指令执行能力。近期通用视觉语言预训练领域的进展 [87, 90, 22,86] 也证实了生成式语言建模的价值。此外，对于 LANA 来说，指令生成不仅是预训练后通常会被弃用的代理任务，也是微调阶段的主要训练目标，并且是部署期间实现基于语言的路线指引能力的根本基础。

3. 方法

任务设置。 我们的目标是构建一个具备语言能力的导航智能体，仅使用单个模型实例，同时掌握指令跟随和生成任务。

• 指令跟随：智能体需要为目标位置找到一条包含 $R=\{r_t\}_{t=1}^{T}$ 个 T 步骤的路线，并遵循一条由人类撰写、包含 $X=\{x_l\}_{l=1}^{L}$ L 个词的指令。在每一步t，智能体都会获取全景 RGB 感知 O^t，并将其离散化为 $K=36$ 个视图，即， $O_t=\{o_{t,k}\in\mathbb{R}^{3\times224\times224}\}_{k=1}^{K}$。智能体选择一个导航动作 a_t，即一个可导航视角，以$\{o_{t,k}\}_k$ 执行。 O_t 和 a_t 共同决定步骤 t 的路线状态 r_t，即， $r_t=(O_t,\ a_t)$。

• 指令生成： 智能体观察一条导航路线 $R=\{r_t\}_{t=1}^{T}$，即., 一系列动作 $\{a_t\}_{t=1}^{T}$ 以及全景感知 $\{O_t\}_{t=1}^{T}$，并且必须用语言描述该路线 $X=\{x_l\}_{l=1}^{L}$ R，生成有依据的描述。

方法概述。 为应对这一具有挑战性的设定，我们设计了 LANA，一个基于 Transformer 的多任务视觉语言导航智能体（图2），在架构设计（§3.1）和网络训练（§3.2）中充分利用跨任务共性。

3.1. 模型架构

LANA 可以将导航路线 $R=\{r_t\}_t$ 作为输入，并输出相应的描述 $X=\{x_l\}_l$，反之亦然。为实现导航路线 R 与指令 X之间的双向翻译并探索跨任务关联性，LANA 被精心设计为由以下部分组成：

• 路径编码器 $\mathcal{E}^r$ 和语言编码器 $\mathcal{E}^l$，基于自注意力对单模态表示进行编码；以及

• 语言解码器 $\mathcal{D}^l$ 和路线解码器 $\mathcal{D}^r$，基于交叉注意力实现路线与语言的双向翻译。

两个单模态编码器由两个多模态解码器共享，并与其联合训练。它们紧密协作：

• 指令遵循： 在导航步骤 t，LANA 分别将完整指令 X、历史路线状态序列 $\{r_1,\cdots,r_{t-1}\}$ 和当前感知O_t 输入相应的编码器，并利用路线解码器 $\mathcal{D}^r$预测导航动作 a_t。

• 指令生成： 在生成步骤l，LANA 分别将完整路线 R 和先前预测的词语$\{x_1,\cdots,x_{l-1}\}$ 输入相应的编码器，并利用语言解码器 $\mathcal{D}^l$预测下一个词 x_l。

我们指出，LANA 以自回归 的方式执行指令生成任务。通过这种方式，两个解码器都以两个编码器的输出为条件，从而实现广泛的视觉与语言知识交换。

路径编码器 $\mathcal{E}^r$。 $\mathcal{E}^r$ 在指令生成期间，以整条路径作为输入，即，$R=\{r_t\}_{t=1}^{T}=\{(O_t,\ a_t)\}_{t=1}^{T}$；在寻路期间，则以历史路径状态和当前观测作为输入，即，$\{r_1,\cdots,r_{t-1},O_t\}=\{O_1,a_1,\cdots,O_{t-1},a_{t-1},O_t\}$。因此，它具有两种类型的输入词元，分别对应全景观测 O_t 和动作 a_t。具体而言， O_t 的观测标记计算如下：

$$\begin{aligned}\boldsymbol{O}_t&=[\boldsymbol{o}_{t,1},\boldsymbol{o}_{t,2},\cdots,\boldsymbol{o}_{t,K}]\ \in\mathbb{R}^{K\times d},\\ \boldsymbol{o}_{t,k}&=\mathcal{F}^v(\boldsymbol{v}_{t,k})+\mathcal{F}^\theta(\boldsymbol{\theta}_{t,k})+\boldsymbol{\tau}^t+\boldsymbol{\tau}^O\ \in\mathbb{R}^d,\end{aligned}\tag{1}$$

其中，$\boldsymbol{v}_{t,\,k}$ 和 $\boldsymbol{\theta}_{t,\,k}$ 分别是视图 $o_{t,\,k}$ 的视觉嵌入和方向嵌入；$\mathcal{F}^{v/\theta}$ 是用于特征维度对齐的线性投影；$\boldsymbol{\tau}^t \in \mathbb{R}^d$ 对时间顺序 – t 进行嵌入；$\boldsymbol{\tau}^O \in \mathbb{R}^d$ 是可学习类型嵌入，用于表示 $\boldsymbol{o}_{t,\,k}$ 是观测标记。

类似地，a_t 的动作标记定义为：

$$\boldsymbol{a}_t = \mathcal{F}^v(\boldsymbol{v}_{t,a_t}) + \mathcal{F}^\theta(\boldsymbol{\theta}_{t,a_t}) + \boldsymbol{\tau}^t + \boldsymbol{\tau}^A \ \in \mathbb{R}^d, \tag{2}$$

其中 $\boldsymbol{v}_{t,\,a_t}$ 和 $\boldsymbol{\theta}_{t,\,a_t}$ 分别嵌入与动作 a_t 相关联的视觉视图和转向角。与公式 1 类似，$\boldsymbol{\tau}^A \in \mathbb{R}^d$ 编码动作标记类型。

对每个全景感知 O_t 的所有 K 子视图 $\{\boldsymbol{o}_{t,k}\}_k$ 进行标记化，使 LANA 能够访问/记忆导航路线上的所有观测结果。然而，考虑如此多的标记会给基于自注意力的编码带来难以承受的计算负担。为在计算成本和表征能力之间取得良好平衡，我们首先计算动作感知路线状态：

$$\begin{aligned} &\boldsymbol{r}_t = \boldsymbol{a}_t + \boldsymbol{c}_t \ \in \mathbb{R}^d, \\ &\boldsymbol{c}_t = cross_att(\boldsymbol{a}_t, \boldsymbol{O}_t) = cross_att(\boldsymbol{a}_t, [\boldsymbol{o}_{t,k}]_{k=1}^K) \ \in \mathbb{R}^d. \end{aligned} \tag{3}$$

通过交叉注意力，即，$cross\ att(\cdot,\ \cdot)$，收集与动作相关的视觉上下文 $\boldsymbol{c}_t$ 并将其压缩为一个 d 维向量。然后通过以下方式得到 $\mathcal{E}^r$ 的输出：

$$\begin{aligned} &\text{Ins. following: } [\bar{\boldsymbol{r}}_{1:t-1}, \bar{\boldsymbol{O}}_t] = self_att([\boldsymbol{r}_{1:t-1}, \boldsymbol{O}_t]) \in \mathbb{R}^{(t-1+K)\times d}, \\ &\text{Ins. generation: } \quad [\bar{\boldsymbol{r}}_{1:T}] = self_att([\boldsymbol{r}_{1:T}]) \in \mathbb{R}^{T\times d}. \end{aligned} \tag{4}$$

语言编码器 $\mathcal{E}^l$。$\mathcal{E}^l$ 在寻路期间以完整指令（即，$X = \{x_l\}_{l=1}^L$）作为输入；或在指令生成期间以先前生成的词（即，$\{x_1, \cdots, x_{l-1}\}$）作为输入。它采用标准的 Transformer 语言编码器，用于上下文化的语言特征提取：

$$\begin{aligned} &\text{Ins. following: } \quad [\bar{\boldsymbol{x}}_{1:L}] = \mathcal{E}^l([x_{1:L}]) \in \mathbb{R}^{L\times d}, \\ &\text{Ins. generation: } [\bar{\boldsymbol{x}}_{1:l-1}] = \mathcal{E}^l([x_{1:l-1}]) \in \mathbb{R}^{(l-1)\times d}, \end{aligned} \tag{5}$$

其中 $\mathcal{E}^l$ 包含若干个模块，每个模块都有一个多头自注意力层和一个前馈子层 [66]；为简洁起见，省略了位置嵌入。

值得注意的是，在训练指令生成任务时，因果未来掩码 [75]会应用于每个自注意力层，确保每个词元只能关注前面的词元，从而使我们的单一模型能够同时完成指令跟随和指令生成。

路线解码器 $\mathcal{D}^r$。$\mathcal{D}^r$ 用于将指令转换为路线。具体而言，在导航步骤 t，$\mathcal{D}^r$ 以完整的指令嵌入 $\bar{\boldsymbol{x}}_{1:L}$、历史路线状态 $\bar{\boldsymbol{r}}_{1:t-1}$ 以及当前观测特征 $\boldsymbol{O}_t = \bar{\boldsymbol{o}}_{t\text{作为输入},\ 1:K}$ 并输出在当前 $\boldsymbol{p}_t \in \triangle^K$ 子视图 K 上进行动作选择的概率分布 $o_{t,\ 1:K}$[1]。更具体地说，$\mathcal{D}^r$ 由多个基于交叉注意力的模块堆叠而成，用于建模跨模态关系。对于每个模块，我们有：

$$[\hat{\boldsymbol{r}}_{1:t-1}, \hat{\boldsymbol{o}}_{t,1:K}] = cross_att([\bar{\boldsymbol{r}}_{1:t-1}, \bar{\boldsymbol{o}}_{t,1:K}], \bar{\boldsymbol{x}}_{1:L}), \tag{6}$$

$$[\bar{\boldsymbol{r}}_{1:t-1}, \bar{\boldsymbol{o}}_{t,1:K}] \leftarrow self_att([\hat{\boldsymbol{r}}_{1:t-1}, \hat{\boldsymbol{o}}_{t,1:K}]). \tag{7}$$

式 6 通过交叉注意力获得语言增强的路线和观测表征，即，$\hat{}\boldsymbol{r}_{1:t-1}$ 和 $\hat{}\boldsymbol{o}_{t,\ 1:K}$。式 7 采用自注意力对历史路线状态 $\hat{}\boldsymbol{r}_{1:t-1}$ 之间的时间依赖关系进行建模，并捕捉 $\hat{}\boldsymbol{r}_{1:t-1}$ 与当前观测 $O_t = \hat{}\boldsymbol{o}_{t\text{之间的相关性},\ 1:K}$。

经过若干个 $\mathcal{D}^r$ 解码器模块后，K 子视图上的动作概率 $o_{t,\ 1:K}$ 表示为：

$$\boldsymbol{p}_t = softmax(\{\mathcal{F}^r(\bar{\boldsymbol{o}}_k)\}_{k=1}^K) \ \in \triangle^K, \tag{8}$$

其中 $\mathcal{F}^r$：$\mathbb{R}^d \to \mathbb{R}$ 是一个用于动作分数映射的两层前馈网络，与 [11, 92]中相同。

语言解码器 $\mathcal{D}^l$。$\mathcal{D}^l$ 用于将路线转换为指令。具体而言，在指令生成步骤 l，$\mathcal{D}^l$ 以完整路线状态 $\bar{\boldsymbol{r}}_{1:T}$ 和先前生成的指令词嵌入 $\bar{\boldsymbol{x}}_{1:l-1}$ 为输入，并输出在预定义词汇表（含 M 个词）上进行词语选择的概率分布 $\boldsymbol{q}_l \in \triangle^M$。与 $\mathcal{D}^r$ 类似，$\mathcal{D}^l$ 包含若干个基于交叉注意力的模块。每个模块表示为：

$$\hat{\boldsymbol{x}}_{1:l-1} = cross_att(\bar{\boldsymbol{x}}_{1:l-1}, \bar{\boldsymbol{r}}_{1:T}), \tag{9}$$

$$\bar{\boldsymbol{x}}_{1:l-1} \leftarrow causal_self_att(\hat{\boldsymbol{x}}_{1:l-1}). \tag{10}$$

公式 9 使文本能够关注路线上的下文。在公式 10 中，我们采用因果掩码自注意力，而非普通的双向自注意力，以迫使 $\mathcal{D}^l$“向前关注”，这是自回归推理所必需的。

经过若干个 $\mathcal{D}^l$ 解码器块后，M 词词汇表上的概率分布如下：

$$\boldsymbol{q}_l = softmax(\mathcal{F}^l(\bar{\boldsymbol{x}}_{l-1})) \ \in \triangle^M, \tag{11}$$

其中，$\mathcal{F}^r$:$\mathbb{R}^d \to \mathbb{R}^M$ 是一个两层前馈网络，用于预测词语分数分布。

3.2. 网络训练

LANA 的所有模块，即两个单模态编码器 $\mathcal{E}^r$ 和 $\mathcal{E}^l$，以及两个多模态解码器 $\mathcal{D}^r$ 和 $\mathcal{D}^l$，均通过优化指令跟随和生成的训练目标进行联合端到端学习。

指令生成。对于每个指令-路线训练对（X、R），其中 $X = x_{1:L}$ 和 $R = r_{1:T}$，LANA 通过基于完整路线 R 并依次预测参考词 $x_{0:l-1}$ 来学习指令生成。我们在 X 后附加两个特殊标记，即 $x_0 =$ [序列起始标记] 和 $x_{L+1} =$ [序列结束标记]，，分别表示指令句子的开始和结束。为了生成词 x_l，LANA 分别将 R 和 $x_{0:l-1}$ 输入 $\mathcal{E}^r$ 和 $\mathcal{E}^l$ 进行单模态编码（参见公式4和5）。在路线和语言嵌入（即 $\bar{\boldsymbol{r}}_{1:T}$ 和 $\bar{\boldsymbol{x}}_{1:l-1}$）的条件下，$\mathcal{D}^l$ 给出词语概率 $\boldsymbol{q}_l$（参见公式11）。指令生成的训练目标表示为语言建模损失，可写为：

$$\mathcal{L}^g = -\sum_{l=1}^{L+1} \log(p(x_l | x_{0:l-1}, R)) = -\sum_{l=1}^{L+1} \log(\boldsymbol{q}_l(x_l)), \tag{12}$$

[1]请注意，除了 K 动作子视图外，此处还考虑了停止标记，因此共有 K+1种决策选项。为简洁起见，我们省略停止标记。

其中， $\boldsymbol{q}_l$（x_l）$\in[0,1]$是词 x_l的概率。LANA通过最小化参考指令词的负对数似然进行训练。这里采用教师强制[89]，以支持并行文本输入。值得一提的是，现有的视觉-语言导航预训练方法 [30, 31, 30, 11, 64] 依赖掩码语言建模（MLM）策略。由于在每次训练迭代中，掩码语言建模仅预测输入词的一小部分（通常为15%），因此对于大规模预训练数据而言效率较低，正如近期许多关于通用视觉语言预训练的研究 [34, 9, 15]所指出的那样。指令跟随。对于每个训练对（X、R），其中$X=x_{1:L}$和 $R=r_{1:T}=$（O_t、a_t）$_{1:T}$，LANA通过基于完整指令 X、专家演示 $r_{1:t-1}$中的历史记录以及当前感知 O_t预测 a_t，同时学习指令跟随。具体而言，LANA分别将 X 和 $\{r_{1:t-1}, O_t\}$ 输入 $\mathcal{E}^l$ 和 $\mathcal{E}^r$（参见公式4和5）进行单模态编码。在输出的单模态编码（即 $\bar{\boldsymbol{x}}_{1:L}$ 和$[\bar{\boldsymbol{r}}_{1:t-1}, \boldsymbol{O}_t]$）的条件下， $\mathcal{D}^r$ 给出动作概率 $\boldsymbol{p}_t$（参见公式8）。指令跟随的训练目标是最小化目标视点动作 a_t的负对数似然：

$$\mathcal{L}^f=-\sum_{t=1}^{T}\log(p(a_t|r_{0:t-1},O_t,X))=-\sum_{t=1}^{T}\log(\boldsymbol{p}_t(a_t)). \quad (13)$$

LANA 在预训练和微调阶段，均通过两个训练目标进行端到端学习（参见公式12和13）。请注意，编码器 $\mathcal{E}^r$ 和 $\mathcal{E}^l$ 同时接收来自指令生成（参见公式 12）和遵循（参见公式13）的监督信号。此外，这种联合学习框架使 LANA 更具可解释性——LANA 可以被视为一个天生配备语言解释器的导航器 $\mathcal{D}^l$。

3.3. 实现细节

网络架构。路线 $\mathcal{E}^r$ 和语言 $\mathcal{E}^l$ 编码器分别有一层和九层，解码器$\mathcal{D}^r$和 $\mathcal{D}^l$ 均包含四个模块。特征维度设为 $d=768$。视图 o_k的方向特征 $\boldsymbol{\theta}_k$（参见公式 1）定义为： $\boldsymbol{\theta}_k=$(cos φ_k, sin φ_k, cos ϕ_k, sin ϕ_k)，其中 φ和 ϕ分别为航向角和仰角。训练。遵循近期视觉语言导航的做法 [56, 31, 30, 11,64],，采用预训练与微调范式：

• 预训练：使用这两个训练目标（参见公式12&13），LANA 在 PREVALENT 中离线采样的指令—路线对上进行预训练， [31], 其中包括 104K 个原始 R2R 样本和 6482K 个合成样本。LANA 训练了 100K 次迭代，使用 Adam优化器 [41]，学习率为 1e-4， $N=128$ 批量大小为该设定值。

• 微调：随后，我们在不同的视觉语言导航数据集上微调 LANA，仍采用我们的两项训练任务（参见公式 12&13）。遵循标准流程 [33, 78, 32, 83],，指令跟随的训练基于模仿学习与强化学习的结合。在此阶段，我们将学习率设为 1e-5，批量大小设为 8。

我们使用四块 NVIDIA Tesla A100 GPU 训练网络，并且每个小批次只采样一个训练任务。

推理。 训练完成后，LANA 无需任何架构变更，仅凭单个模型实例即可遵循导航指令并将其转述。具体而言，对于指令跟随，我们采用贪心搜索，即在每个预测步选择概率最高的动作，并在选择停止时终止。对于指令生成，则以自回归方式预测句子，即每次生成一个词，直到选择序列结束标记；生成过程以之前生成的内容为条件。

4. 实验

我们评估 LANA 在指令跟随（§4.1）和生成（§4.2）任务上的表现，随后开展一系列诊断实验（§4.3）和定性研究（§4.4）。

对于每项任务，我们给出 LANA 两个版本的得分：

• LANA$_{mt}$：在预训练和微调过程中联合学习两个目标任务。因此，这种多任务版本只有一个智能体实例，并在两项任务上进行测试。

• LANA$_{st}$：在两项任务上联合进行预训练，但针对每项任务分别进行微调。共有两个单任务智能体实例，每个实例仅在相应任务上进行测试。

我们在此总结后续实验中的关键观察：**i)** LANA 的表现与以往针对特定任务的智能体相当，甚至更好；**ii)** 相比 LANA$_{st}$，LANA$_{mt}$ 在参数利用方面更高效，表现也更出色；**iii)** LANA 可以通过用语言描述其导航路线，在测试时提供行为解读；以及**iv)**我们的模型设计和训练目标确实促成了出色的结果。

4.1. 指令跟随性能

数据集。 我们在三个视觉语言导航数据集上开展实验：

• R2R [4]：它包含四个划分，即训练集（61 个场景，14,039 条指令）、已见验证集（61 个场景，1,021 条指令）、未见验证集（11 个场景，2,349 条指令）和未见测试集（18 个场景，4,173 条指令）。训练集与未见划分之间没有重叠场景。

• R4R [38]：它通过连接 R2R 中两条首尾相接的轨迹及其对应指令，对 R2R 进行了扩展。R4R 包含三个集合，即., 训练集（61 个场景，233，613 条指令）、已见验证集（61 个场景，1，035 条指令）和未见验证集（11 个场景，45，162 条指令）。

• REVERIE [63]：它以目标位置和物体的高层描述取代了 R2R 中的详细指令。它由四个集合组成，即., 训练集（53 个场景，10， 466 条指令）、已见验证集（61 个场景，1，371 条指令）、未见验证集（10 个场景，3，753 条指令）和未见测试集（16 个场景，6，292 条指令）。

评估指标。 对于 R2R，我们遵循惯例 [4, 27]报告四项评估指标：i) 成功率 (SR)、ii)轨迹长度 (TL)、iii) 预言成功率(OR)以及 iv) 路径长度加权成功率 (SPL)，其中 SR

方法	R2R 未见验证集				R2R 未见测试集			
	成功率↑ 加权路径长度成功率↑目标进度率↑轨迹长度↓				成功率↑ 加权路径长度成功率↑ 目标进度率↑轨迹长度↓			
BT-follower [27] [NeurIPS2018]	36	-	45	-	35	28	44	14.8
EDrop-follower [71] [NAACL2019]	52	48	-	10.7	51	47	59	11.7
AuxRN [93] [CVPR2020]	55	50	62	-	55	51	62	-
PREVALENT [31] [CVPR2020]	58	53	-	10.2	54	51	-	10.5
视觉语言导航♢BERT [33] [CVPR2021]	63	57	-	12.0	63	57	-	12.4
AirBERT [30] [ICCV2021]	62	56	-	11.8	62	57	-	12.4
HAMT [11] [NeurIPS2021]	65	59	-	11.9	63	58	-	12.7
HOP [64] [CVPR2022]	64	57	-	12.3	64	59	-	12.7
LANA$_{st}$ （我们的）	66	60	73	11.9	64	59	71	12.4
LANA$_{mt}$ （我们的）	**68**	**62**	**76**	12.0	**65**	**60**	**71**	12.6

方法	R4R 未见验证集				
	CLS↑ nDTW ↑ SDTW ↑ 成功率 ↑ 轨迹长度 ↓				
BT-follower [27] [NeurIPS2018]	30	-	-	24	19.9
RCM [83] [CVPR2019]	35	30	13	26	28.5
PTA [43] [NeurIPS2021]	37	32	10	24	17.7
EDrop-follower [71] [NAACL2019]	34	-	9	29	27.0
OAAM [62] [ECCV2020]	40	-	11	31	13.8
ActiveVLN [81] [ECCV2020]	59	44	22	32	19.7
EGP [21] [NeurIPS2020]	44	37	18	30	18.3
HAMT [11] [NeurIPS2021]	57.7	50.3	31.8	44.6	-
LANA$_{st}$ （我们的方法）	58.6	51.9	31.4	43.0	22.7
LANA$_{mt}$ （我们的方法）	**59.7**	**52.3**	31.7	43.2	22.1

表1：定量比较结果（§4.1）用于 **R2R 上的指令跟随** 任务 [4]。“–”：无可用统计数据。

表2：定量比较结果（§4.1）**在 R4R 上进行指令遵循** [38]。

方法	REVERIE 未见验证集						REVERIE 未见测试集					
	SR↑	加权路径长度成功率↑	OR↑	TL↓	RGS↑	RGSPL↑	SR↑	加权路径长度成功率↑	OR↑	TL↓	RGS↑	RGSPL↑
RCM[83] [CVPR2019]	9.29	6.97	14.23	11.98	4.89	3.89	7.84	6.67	11.68	10.60	3.67	3.14
视觉语言导航♢BERT[33] [CVPR2021]	30.67	24.90	35.02	16.78	18.77	15.27	29.61	23.99	32.91	15.86	16.50	13.51
AirBERT[30] [ICCV2021]	27.89	21.88	34.51	18.71	18.23	14.18	30.28	23.61	34.20	17.91	16.83	13.28
HAMT[11] [NeurIPS2021]	32.95	30.20	36.84	14.08	18.92	17.28	30.40	26.67	33.41	13.62	14.88	13.08
HOP[64] [CVPR2022]	30.39	25.10	35.30	17.16	18.23	15.31	29.12	23.37	32.26	17.05	17.13	13.90
LANA（我们的方法）	**34.00**	29.26	**38.54**	16.28	**19.03**	16.18	**33.50**	**26.89**	**36.41**	16.75	**17.53**	**14.25**

表3：定量比较结果（§4.1）在**指令跟随**REVERIE 上 [63]。

方法	R2R 已见验证集						R2R 未见验证集					
	SPICE↑	BLEU-1↑	BLEU-4↑	CIDEr↑	METEOR↑	ROUGE↑	SPICE↑	BLEU-1↑	BLEU-4↑	CIDEr↑	METEOR↑	ROUGE↑
BT-speaker[27] [NeurIPS2018]	0.203	0.537	0.155	0.121	0.233	0.350	0.188	0.522	0.142	0.114	0.228	0.346
EDrop-speaker[71] [NAACL2019]	0.202	-	0.245	0.493	0.228	0.467	0.181	-	0.237	0.422	0.225	0.458
VLS[1] [CVPRW2019]	0.214	0.549	0.157	0.137	0.228	0.352	0.197	0.548	0.159	0.132	0.231	0.357
CCC-speaker[78] [CVPR2022]	0.231	0.728	0.287	0.543	0.236	0.493	0.214	0.708	0.272	0.461	0.231	0.477
LANA$_{st}$ （我们的方法）	0.251	0.743	0.305	0.522	0.243	0.502	0.223	0.722	0.287	0.433	0.235	0.490
LANA$_{mt}$ （我们的方法）	**0.256**	**0.759**	**0.314**	0.533	**0.245**	**0.503**	**0.226**	**0.736**	**0.298**	0.457	**0.238**	**0.498**

表4：定量　　对比结果（§4.2）针对**指令生成**　　在 R2R 上 [4]。

和 SPL 优先考虑。对于 R4R，我们进一步采用 v) 按长度加权的覆盖分数(CLS) [38], vi) 归一化动态时间规整 (nDTW)，以及 vii) 按 *nDTW* 加权的成功率 (SDTW)。对于 REVERIE，其导航子任务也采用前四项指标；此外，还使用 viii) 远程定位成功率 (RGS)和 ix)按路径长度加权的 *RGS* (RGSPL)评估总体表现。

定量结果。 我们将若干知名且近期提出的先进方案 [27, 71, 93, 83, 81, 31, 33, 21, 30, 11, 78, 64] 纳入指令跟随任务的比较。需要注意的是，按照惯例，我们报告**单次运行**设置下单个模型的得分 [33, 78, 64, 11]。如桌子1所示，LANA$_{st}$仅在多任务预训练后针对寻路进行微调，在 R2R 上的结果与其他方案不相上下，甚至更胜一筹。值得注意的是，LANA$_{mt}$通过学习在遵循指令的同时解读导航路径，进一步提升了导航表现。例如，LANA$_{mt}$ 在 ANA$_{st}$ 的基础上，在未见验证集 和测试集上的 SPL 分别提升了 2% 和 1%。这验证了我们具备语言能力的导航方案和多任务学习策略的有效性。在 R4R（参见桌子2）和 REVERIE（参见桌子3）上还能观察到更显著的提升；前者着重于长指令下的长时程导航，而后者仅提供抽象指令和轨迹——前者着重于长指令下的长时程导航，而后者仅提供抽象指令。这些结果证实了我们方法的通用性和灵活性。值得注意的是，所有竞争方案都只能处理寻路，而我们的智能体能够生成有依据的路线描述，用于解读其导航行为/计划。

4.2. 指令生成的性能

数据集。我们在两个视觉语言导航数据集上，将机器生成的路线描述与人工编写的指令进行比较：

• R2R [4]：由于 R2R 测试集被保留用于对遵循指令的智能体进行基准测试，因此我们报告指令生成在验证集上的性能。每条 R2R 导航路径都对应三条真实指令。

• R4R [38]：报告了 R4R 验证集上的性能，其中每条路径对应九条真实指令。

REVERIE [63]不在考虑范围内，因为其指令是对远处物体的高层次、简洁描述，无法满足我们生成接地指令的目的。

评估指标。 遵循 [1, 78], 我们选用五种文本指标：i) BLEU [60],ii) CIDEr [76],iii) METEOR [8],iv) ROUGE [49],和 v) SPICE [3]。对于每条导航路径，这些指标均取其对应所有真实指令的平均值。SPICE 被视为主要指标。

定量结果。我们将 LANA 与四种指令生成算法进行比较 [27, 71, 1, 78]。表4 和表

方法	R4R 已见验证集						R4R 未见验证集					
	SPICE↑	BLEU-1↑	BLEU-4↑	CIDEr↑	METEOR↑	ROUGE↑	SPICE↑	BLEU-1↑	BLEU-4↑	CIDEr↑	METEOR↑	ROUGE↑
BT-speaker [27] [NeurIPS2018]	0.164	0.691	0.223	0.099	0.213	0.453	0.207	0.387	0.088	0.139	0.172	0.359
EDrop-speaker [71] [NAACL2019]	0.209	0.750	0.281	0.216	0.245	0.473	0.218	0.433	0.106	0.200	0.187	0.363
CCC-speaker [78] [CVPR2022]	0.219	0.758	0.312	0.245	0.252	0.480	0.233	0.403	0.115	0.206	0.193	0.365
LANA$_{st}$ （我们的方法）	0.237	0.768	0.327	0.264	**0.265**	0.483	0.259	0.437	0.123	0.216	0.199	0.375
LANA$_{mt}$ （我们的方法）	**0.245**	**0.772**	**0.333**	**0.287**	0.261	**0.484**	**0.262**	**0.443**	**0.128**	**0.231**	**0.200**	**0.376**

表 5：定量比较结果（§4.2）**R4R 上的指令生成** [38]。

#	预训练		微调		指令跟随				指令生成					
	指令遵循	指令生成	指令遵循	指令生成	成功率 ↑	加权路径长度成功率 ↑	目标进度率 ↑	轨迹长度 ↓	SPICE ↑	BLEU-1 ↑	BLEU-4 ↑	CIDEr ↑	METEOR ↑	ROUGE ↑
1			✔		52.1	48.3	59.3	11.2	-	-	-	-	-	-
2				✔	-	-	-	-	0.178	0.692	0.241	0.321	0.216	0.463
3			✔	✔	53.1	48.9	60.9	11.6	0.182	0.704	0.245	0.304	0.219	0.469
4	✔		✔		61.3	55.4	69.7	12.0	-	-	-	-	-	-
5		✔		✔	-	-	-	-	0.215	0.718	0.255	0.378	0.230	0.472
6	✔	✔	✔		65.7	59.9	73.4	11.9	-	-	-	-	-	-
7	✔	✔		✔	-	-	-	-	0.223	0.722	0.287	0.433	0.235	0.490
8	✔	✔	✔	✔	**67.9**	**61.6**	**75.7**	12.0	**0.226**	**0.736**	**0.298**	**0.457**	**0.238**	**0.498**

表6： **消融实验** (§4.3) 在 R2R 未见验证集 [4]。

5总结了我们的比较结果。我们发现，我们的专用任务智能体，即 LANA$_{st}$，在所有指标和数据集上均已胜过所有竞争方法。值得注意的是，CCC [78], 这一当前领先的解决方案借助一个独立的寻路器来学习指令生成模型。更令人印象深刻的是，我们的多任务智能体，即 LANA$_{mt}$，表现与 LANA$_{st}$不相上下，甚至更胜一筹，展现了我们方法在算法和功能上的优势。

用户研究。 为全面衡量我们生成的指令的质量，我们开展了一系列基于两两比较的人工评估实验。具体而言，我们请 50 名大学生分别比较 LANA$_{mt}$ 生成的指令与 CCC、BT-Speaker 及人工编写的指令，涉及共 100 条路径。这些路径采样自 R2R 未见验证集。最终，LANA 获得了更多偏好票数，即，**63.4%** 对比 CCC 的 36.6%，以及**75.1%** 对比 BT-Speaker 的 24.9%。不过，人工编写的指令更受青睐，即，69.3% 对比 LANA 的 30.7%，这表明仍有很大的提升空间。

4.3. 诊断实验

为深入研究我们的语言能力导航框架的有效性，我们在 R2R 的 未见验证集 上 [4], 针对指令跟随和生成任务开展了一系列诊断实验。实验结果总结于表 6。具体而言，我们的消融实验共涉及八种基线:

1. 仅在指令跟随上微调，不进行预训练；2. 仅在指令生成上微调，不进行 预训练；3. 在指令跟随和生成上微调，不进行 预训练；4. 仅在指令跟随上预训练并微调；5. 仅在指令生成上预训练并微调；6. 在指令跟随和生成上预训练，并仅在指令跟随上微调；7. 在指令跟随和生成上预训练，并仅在指令生成上微调；8. 在指令跟随和生成上预训练并微调。

这些基线大致可分为三类：**i**）基线 1、2 和 3 均未进行预训练，并针对每项任务分别或联合进行微调；**ii**）基线 4 和 5 进行预训练，并针对每项任务分别进行微调；以及**iii**）基线 6、7 和 8进行了联合任务预训练，并针对每项任务分别或联合进行微调。基线 6 和 7 是两个单任务智能体，即也就是说，LANA$_{st}$，而基线 8 是我们最终交付的智能体 LANA$_{mt}$；它们的表现已在 §4.1 和 §4.2中进行了全面报告。

此外，请注意，基线 1、4 和 6 只能掌握寻路，而基线 2、5 和 7 只能执行路线描述任务。基线 3 和 8 则两者皆可。

可以得出几个重要结论:

- 联合任务微调能够提升两项任务的性能（基线 3 与1 与 2）；
- 联合任务预训练和微调能够提升两项任务的性 能（基线 8 与 6 与 7）；
- 预训练能够提升最终性能（基线 4 与 1，基线 5 与 2，基线 6 与 1，基线 7 与 2，以及基线 8 与3）；
- 相比单任务预训练，联合任务预训练更受青睐（基线 8 与 4 与5）；
- 与其他所有训练策略相比，联合任务预训练和微调更受青睐（基线 8与 1-7）。

请注意，联合任务预训练和微调不仅提升了性能，还提高了参数效率，即，基线8（143 M）对比 6 + 7（220 M = 123 M +97 M）。总之，我们的消融实验充分验证了这一理念的优势、算法设计的有效性，以及我们在高效参数利用方面的优势。



图 3：(a-b) LANA$_{mt}$ 与 LANA$_{st}$ 在 (a) 指令跟随和 (b) 指令生成任务上的视觉对比结果。导航路线的起点和终点分别用 和 表示。(c) LANA 能够用自然语言解读其导航行为。生成的报告会让监控者[illegible]了解[illegible]航过程，甚至有助于揭示失败模式。例如，在第 2-3 步，LANA$_{mt}$ 进入储藏室，是因为 LANA$_{mt}$ 以为那是卧室。更详细的讨论请见 §4.4 。

4.4. 定性实验

图3展示了 R2R [4] 未见验证集中的三个典型导航片段。图3(a) 对比了 LANA$_{mt}$和 LANA$_{st}$ 在指令遵循任务上的表现。可以看到，LANA$_{mt}$ 在这一具有挑战性的案例中表现稳健，而 LANA$_{st}$ 未能到达目标位置。由于 LANA$_{mt}$和 LANA$_{st}$ 采用了相似的网络架构和预训练方案，我们将这种差异归因于微调过程中对跨任务知识的探索。图3 (b) 展示了指令创作方面的对比。可以看到，LANA$_{mt}$生成的指令更加贴合场景，其中包含精准的动作描述（例如，左转、向下走），以及显著的地标（例如，双开门、餐桌、冰箱）。这些描述与人工生成文本的特征相似，甚至提及了一些人工参考文本中遗漏、但具有提示作用的地标（例如，沙发）。图3 (c) 显示，LANA 可以通过展示其导航过程的文字报告，实时解读行为。这不仅免去了人类持续监控的负担，也在一定程度上揭示了其内部状态。例如，步骤1-3的报告指出，LANA 将储藏室误认为卧室——这就是 LANA 选择在第 2 步进入储藏室。简而言之，作为一种具备语言能力的导航器，LANA 在（事后）可解释性以及人机双向交流方面展现出优势，而这正是建立人类信任的基本前提。

5. 结论与讨论

本研究呼吁范式转变：从当前的视觉语言导航智能体——它们是强大的语言辅助寻路系统，却不具备语言生成能力——转向语言能力更强的导航机器人。这类机器人不仅能够执行导航指令，还能用语言描述导航路线。我们提出了 LANA，它通过单一模型学会同时精通指令跟随与生成。在这两项任务中，LANA 的表现与以往针对特定任务的解决方案不相上下，甚至更胜一筹，同时复杂度大幅降低。尤为重要的是，LANA 能够撰写高质量的路线描述，既能提供丰富信息，帮助理解其行为，也能为人类协作提供指引。我们相信，LANA 为打造具备语言能力的机器人奠定了坚实基础，并使我们更接近构建具有社会智能且值得信赖的机器人的终极目标。未来的工作应利用大规模预训练基础模型中的知识来增强 LANA。

LANA：用于指令跟随与生成的具备语言能力的导航器补充材料

本文档进一步介绍我们的方法，并提供更多实验结果，内容安排如下：

- §6LANA 的实现细节。
- §7 REVERIE 上的补充定量结果 [63]。
- §8关于路径编码器的更多消融研究。
- §9补充定性结果。
- §10关于社会影响与局限性的讨论。

6. LANA 的实现细节

在预训练期间，我们沿用以往工作，额外采用指令轨迹匹配（ITM）任务 [11] ，用于预测一对指令和轨迹是否匹配。三个任务：指令跟随（IF）、指令生成（IG）和指令轨迹匹配（ITM）的采样比例为 IG:IF:ITM=4:1:2。我们在算法1 中给出了预训练流程的伪代码（为简洁起见，省略 ITM）。微调时，指令遵循任务采用强化学习（RL）和模仿学习（IL）进行优化。IL 使用公式13 中的相同损失，而 RL 则基于异步优势演员-评论家（A3C）算法 [58]实现。微调期间，IG 和 IF 的采样比例设为 IG:IF=2:5；不再采用 ITM 任务。按照惯例 [11, 33, 64]，我们将物体特征与全景特征拼接起来，并为 REVERIE 上的指令遵循任务添加物体定位损失 [63]。LANA 的详细架构见表7。

	语言编码器 $\mathcal{E}^l$		路径编码器 $\mathcal{E}^r$		语言解码器 $\mathcal{D}^l$		路线解码器 $\mathcal{D}^r$	
层	自注意力 前馈网络	×9	交叉注意力 自注意 力前馈网络	×1	交叉注意力 自注意 力前馈网络	×4	交叉注意力 自注意 力 前馈网络	×4

表 7：L 的详细模型架构ANA (§6)。

7. REVERIE 上的其他定量结果

PREVALENT 数据集中的合成样本由一个使用 R2R 训练的说话人模型生成 [4]。近期的一项工作 DUET [13] 通过使用在 REVERIE 数据集上训练的说话人模型合成指令，收集了一个新的增强数据集 [63]。我们在桌子 8中报告了使用该数据集训练的 LANA 的更多量化结果。值得注意的是，这种训练策略大幅提升了 REVERIE [63]上的表现。使用相同训练集时，LANA 的导航表现优于 DUET [13]，这表明了我们方法在算法上的优势。

算法 1LANA 预训练的伪代码。

参数： 标注数据集 $\mathcal{H}=\{(R,X)\}$，最大迭—代次数 N、路径编码器 $\mathcal{E}^r$、语言编码器 $\mathcal{E}^l$，语言解码器、 $\mathcal{D}^l$和路线解码器 $\mathcal{D}^r$。

初始化 $\mathcal{E}^r, \mathcal{E}^l, \mathcal{D}^r, \mathcal{D}^l$
循环 迭代 $i \in [1,\dots,N]$ **执行**
  抽取批次 $B \subset \mathcal{H}$
  抽取一个预训练任务 $\mathcal{T}$，来自 $\{IG、IF\}$
  $\mathcal{L} \leftarrow 0$
  如果 $\mathcal{T}$是IG，**则**
    对于 $(R,\ X) \in B$ **执行**
      $[\bar{\boldsymbol{r}}_{1:T}] = \mathcal{E}^r(R)$
      $[\bar{\boldsymbol{x}}_{1:l-1}] = \mathcal{E}^l([x_{1:l-1}])$
      $\boldsymbol{q}_l = \mathcal{D}^l([\bar{\boldsymbol{x}}_{1:l-1}],\ [\bar{\boldsymbol{r}}_{1:T}])$
      估计 $\mathcal{L}^g$ ▷ 定义见公式12。
      $\mathcal{L} \leftarrow \mathcal{L} + \mathcal{L}^g$
    计算 $\partial\mathcal{L}$
    更新 $\mathcal{E}^r, \mathcal{E}^l, \mathcal{D}^l$
  否则如果 $\mathcal{T}$是 如果 **那么**
    对于 $(R,X) \in B$ **执行**
      $[\bar{\boldsymbol{r}}_{1:t-1}, \bar{\boldsymbol{O}}_t] = \mathcal{E}^r([\boldsymbol{r}_{1:t-1}, \boldsymbol{O}_t])$
      $[\bar{\boldsymbol{x}}_{1:L}] = \mathcal{E}^l(X)$
      $\boldsymbol{p}_t = \mathcal{D}^r([\bar{\boldsymbol{r}}_{1:t-1}, \bar{\boldsymbol{O}}_t], [\bar{\boldsymbol{x}}_{1:L}])$
      估算 $\mathcal{L}^f$ ▷ 定义见式13。
      $\mathcal{L} \leftarrow \mathcal{L} + \mathcal{L}^f$
    计算 $\partial\mathcal{L}$
    更新 $\mathcal{E}^r, \mathcal{E}^l, \mathcal{D}^r$
返回 $\mathcal{E}^r$, $\mathcal{E}^l, \mathcal{D}^r, \mathcal{D}^l$

8. 路径编码器的更多消融研究

在本节中，我们进一步研究路径编码器设计的有效性。我们的路径编码器 $\mathcal{E}^r$ 同时考虑先前的动作标记 $\{\boldsymbol{a}_t\}_t$ 和过去的全景观测 $\{\boldsymbol{O}_t\}_t$ （见公式3）。因此，我们报告了两个变体，其路径编码器分别：1）仅对先前的动作标记 $\{\boldsymbol{a}_t\}_t$执行时间自注意力；2）仅采用交叉注意力操作来感知历史全景观测 $\{\boldsymbol{O}_t\}_t$。

R2R未见验证集 [4] 的结果汇总于表 9。如图所示，同时整合历史全景观测和先前动作信息可获得最佳性能。

方法	REVERIE 未见验证集						REVERIE 未见测试集					
	SR ↑	加权路径长度成功率 ↑	OR ↑	TL ↓	RGS ↑	RGSPL ↑	SR ↑	加权路径长度成功率 ↑	OR ↑	TL ↓	RGS ↑	RGSPL ↑
RCM[85] [CVPR2019]	9.29	6.97	14.23	11.98	4.89	3.89	7.84	6.67	11.68	10.60	3.67	3.14
视觉语言导航○BERT[33] [CVPR2021]	30.67	24.90	35.02	16.78	18.77	15.27	29.61	23.99	32.91	15.86	16.50	13.51
AirBERT[30] [ICCV2021]	27.89	21.88	34.51	18.71	18.23	14.18	30.28	23.61	34.20	17.91	16.83	13.28
HAMT[11] [NeurIPS2021]	32.95	30.20	36.84	14.08	18.92	17.28	30.40	26.67	33.41	13.62	14.88	13.08
HOP[64] [CVPR2022]	30.39	25.10	35.30	17.16	18.23	15.31	29.12	23.37	32.26	17.05	17.13	13.90
DUET† [13] [CVPR2022]	46.98	33.73	51.07	22.11	32.15	23.03	52.51	36.06	56.91	21.30	31.88	22.06
LANA（我们的方法）	34.00	29.26	38.54	16.28	19.03	16.18	33.50	26.89	36.41	16.75	17.53	14.25
LANA† （我们的方法）	**48.31**	**33.86**	**52.97**	23.18	**32.86**	22.77	51.72	**36.45**	**57.20**	18.83	**32.95**	**22.85**

表 8：REVERIE 上指令跟随的补充定量结果 [63]。 † 表示该模型是在 DUET 数据集上训练的 [13]。详情请参阅 §7 。

#	路径编码器		指令跟随				指令生成					
	自注意力	交叉注意力	SR ↑	加权路径长度成功率 ↑	OR ↑	TL ↓	SPICE ↑	BLEU-1 ↑	BLEU-4 ↑	CIDEr ↑	METEOR ↑	ROUGE ↑
1	✔		65.6	60.1	72.7	11.7	0.202	0.703	0.262	0.375	0.224	0.476
2		✔	64.8	59.8	72.8	11.9	0.222	0.715	0.281	0.430	0.233	0.486
3	✔	✔	**67.9**	**61.6**	**75.7**	12.0	**0.226**	**0.736**	**0.298**	**0.457**	**0.238**	**0.498**

表 9：R2R 消融实验未见验证集 [4]。详情请参见 §8 。

9. 补充定性结果

本节将展示更多关于指令跟随和指令生成的定性结果。图4展示了 LANA_{mt} 与 LANA_{st} 在指令生成方面的对比。我们可以看到，LANA_{mt} 生成的指令更加准确、生动。具体而言，LANA_{mt} 不仅能够描述精确的动作（例如，左转、穿过），还能突出关键地标（例如，办公室、浴室、厕所）。

图 5比较了 LANA_{mt} 和 LANA_{st} 在指令跟随方面的表现。面对具有挑战性的指令“离开壁橱 ⋯ 在你左侧”，LANA_{mt} 成功执行动作并到达目标位置，而 LANA_{st} 则在错误的位置终止导航。这直观地证明了联合训练策略的有效性。

图6展示了 LANA_{mt}提供的实时行为描述。生成的报告使监控器能够及时了解导航过程，并揭示其内在决策模式。例如，针对步骤1-3和步骤3-8生成的路线描述，能够生动地向人类说明 LANA_{mt} 如何执行复杂指令“沿着走廊走到桌子和椅子的左侧”——先是“走进房间。停在桌子前面”，然后“离开房间并穿过走廊”。这个案例展现了 LANA 在可解释性和人机通信方面的优势。

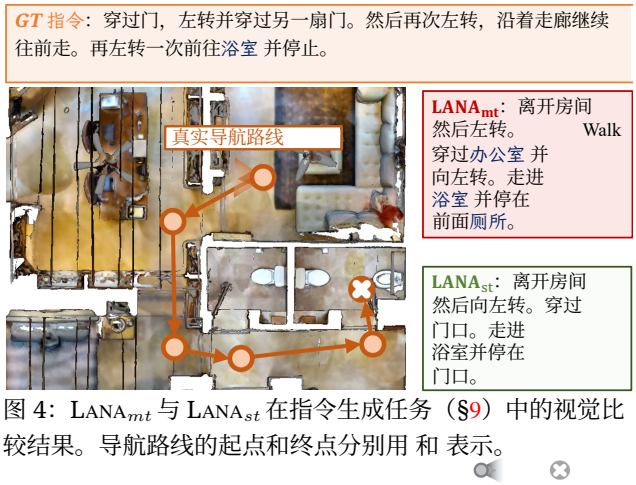

图 4：LANA_{mt} 与 LANA_{st} 在指令生成任务（§9）中的视觉比较结果。导航路线的起点和终点分别用 和 表示。

10. 讨论

社会影响。 与以往“愚笨”的导航器相比，具备语言能力的导航器能够应用于更广泛的场景，并能更深入地融入人们的日常生活。它还可以作为导引机器人，帮助低视力或失明人士。

局限性。 该智能体是在虚拟仿真环境中开发的。如果将该算法部署在真实动态环境中的真实机器人上，导航过程中的碰撞可能会对人员和财产造成损害。应开展更多工作，以推进真实场景中的部署，例如，在动作空间中引入硬性约束以避免碰撞，并开展更多实验来研究潜在损害的风险。此外，生成的路线描述虽然信息丰富、便于人类阅读，但属于一种事后解释，无法完美、准确地解释智能体的内部决策模式。

未来工作。 未来，除了研究我们的方法在其他导航任务（例如，物体目标导航、音频目标导航）中的有效性外，我们还将设计更紧凑的架构，以联合学习这两项任务。

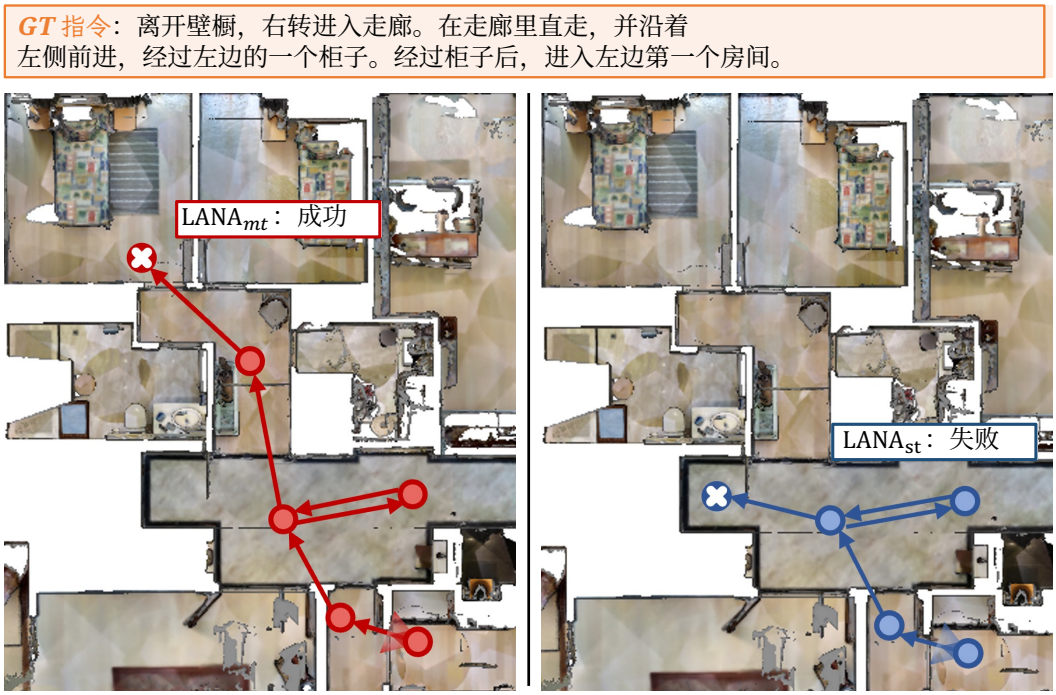

图5：LANA$_{mt}$ 与 LANA$_{st}$ 在指令遵循任务（§9）中的视觉对比结果。导航路线的起点和终点分别用和表示。

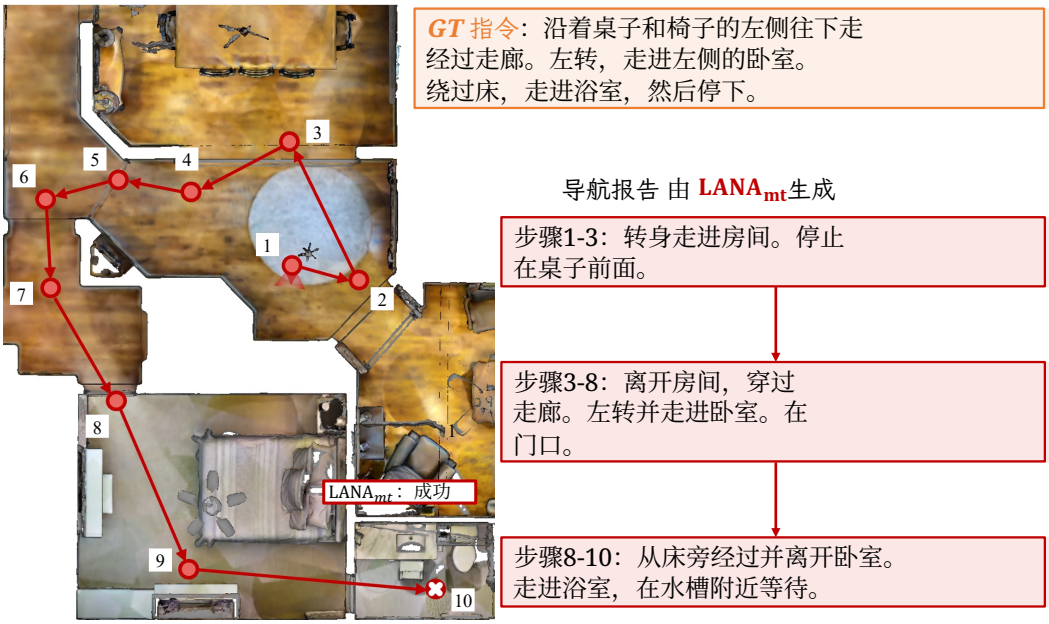

图6：逐步导航行为解释（§9）。导航路线的起点和终点分别用和表示。LANA 能够使用自然语言解释其导航行为。例如，在步骤 2–3中，LANA$_{mt}$ 进入房间后又离开房间，因为 LANA$_{mt}$ 打算寻找指令中提到的桌子。

参考文献

[1] Sanyam Agarwal、Devi Parikh、Dhruv Batra、Peter Anderson 和 Stefan Lee。用于生成有依据且可解释的导航指令的视觉地标选择。发表于 *CVPR*研讨会，2019年。1，2，6

[2] Gary L Allen. 从知识到词语再到寻路：路线指令生成与理解中的问题。载于国际空间信息理论会议，1997年。2[3] Peter Anderson、Basura Fernando、Mark Johnson 和

Stephen Gould。SPICE：语义命题图像描述评估。载于 ECCV，2016年。6[4] Peter Anderson、Qi Wu、Damien Teney、Jake Bruce、Mark Johnson、Niko S¨underhauf、Ian Reid、Stephen Gould 和 Anton van den Hengel。视觉与语言导航：在真实环境中解读基于视觉的导航指令。载于 CVPR，2018 1, 2, 5, 6, 7, 8, 9, 10。Jacob [5] Andreas 和 Dan Klein。基于对齐的组合语义与指令跟随。载于 EMNLP，2015年。2[6] Sean Andrist、Erin Spannan 和 Bilge Mutlu。修辞型机器人：利用体现专业知识的语言线索，让机器人更有效地表达。载于国际人机交互会议，2013年。2[7] Jacob Arkin、Daehyung Park、Subhro Roy、Matthew R Wal-ter、Nicholas Roy、Thomas M Howard 和 Rohan Paul。用于稳健执行机器人指令的潜在语义知识多模态估计与传达。《国际机器人学研究期刊》，39(10-11):1279– 1304, 2020 2。Satanjeev [8] Banerjee 和 Alon Lavie。METEOR：一种改进了与人类判断相关性的机器翻译自动评估指标。载于 ACL研讨会，2005年。6[9] Tom Brown、Benjamin Mann、Nick Ryder、Melanie Sub-biah、Jared D Kaplan、Prafulla Dhariwal、Arvind Neelakan-tan、Pranav Shyam、Girish Sastry、Amanda Askell 等。语言模型是少样本学习者。载于 NeurIPS，2020 5。 [10] David L Chen 和 Raymond J Mooney。从观测中学习解读自然语言导航指令。载于人工智能促进协会年会，2011年。2[11] S hizhe Chen、Pierre-Louis Guhur、Cordelia Schmid 和 Ivan Laptev。用于视觉与语言导航的历史感知多模态变换器。载于 NeurIPS，2021年。2、3、4、5、6、9、10[12] Shizhe Chen、Pierre-Louis Guhur、Makarand Tapaswi、Cordelia Schmid 和 Ivan Laptev。从无标注的三维环境中学习视觉与语言导航。载于 ECCV，2022年。2、3[13] Shizhe Chen、Pierre-Louis Guhur、Makarand Tapaswi、Cordelia Schmid 和 Ivan Laptev。全局思考，局部行动：用于视觉与语言导航的双尺度图Transformer。载于 CVPR，2022年。2、9、10[14] Yen-Chun Chen、Linjie Li、Licheng Yu、Ahmed El Kholy、Faisal Ahmed、Zhe Gan、Yu Cheng 和 Jingjing Liu。UNITER：通用图文表征学习。载于 ECCV，2020年。3[15]Aakanksha Chowdhery、Sharan Narang、Jacob Devlin、Maarten Bosma、Gaurav Mishra、Adam Roberts、Paul Barham、Hyung Won Chung、Charles Sutton、Sebastian Gehrmann 等。PaLM：利用路径扩展语言建模。arXiv 预印本 arXiv:2204.02311，2022年。5[16] Heriberto Cuay´ahuitl、Nina Dethlefs、Lutz Frommberger、Kai-Florian Richter 和 John Bateman。利用分层强化学习生成自适应路线指令。载于国际空间认知会议，2010 2。 [17] Amanda Cercas Curry、Dimitra Gkatzia 和 Verena Rieser。在虚拟环境中生成和评估基于地标的导航指令。载于欧洲自然语言生成研讨会，2015年。2

[18] Robert Dale、Sabine Geldof 和 J Prost。将自然语言生成用于自动路线。《信息技术研究与实践期刊》，36 3 23, 2004 2 ( ): 。 [19] Andrea F Daniele、Mohit Bansal 和 Matthew R Walter。将导航指令生成视为结合神经机器翻译的逆向强化学习。载于人机交互国际会议，2017年。1, 2[20] Laure De Cock、Kristien Ooms、Nico Van de Weghe、Nina Vanhaeren、Pieter Pauwels 和 Philippe De Maeyer。确定室内路线引导中决策点复杂性感知的构成因素。《国际地理信息科学杂志》，35(6):1232–1250，2021年。2[21] Zhiwei Deng、Karthik Narasimhan 和 Olga Russakovsky。演进式图形规划器：用于视觉与语言导航的情境化全局规划。载于 NeurIPS，2020 2, 6。Karan [22] Desai 和 Justin Johnson。Virtex：从文本注释中学习视觉表征。载于 CVPR，2021 3。 [23] Zi-Yi Dou 和 Nanyun Peng。Foam：关注跟随者的视觉与语言导航说话者模型。载于 NAACL，2022年。1, 2[24] Mary T Dzindolet、Scott A Peterson、Regina A Pomranky、Linda G Pierce 和 Hall P Beck。信任在自动化依赖中的作用。《国际人机研究杂志》，58 6 697 718, 2003 2 ( ): – 。Gary W [25] Evans、David G Marrero 和 Patricia A Butler。环境学习与认知地图构建。《环境与行为》，13 1 83 104, 1981 2 ( ): – 。Daniel Fried、 [26] Jacob Andreas 和 Dan Klein。用于生成和遵循指令的统一语用模型。arXiv 预印本 arXiv:1711.04987，2017年。2[27] Daniel Fried、Ronghang Hu、Volkan Cirik、Anna Rohrbach、Jacob Andreas、Louis-Philippe Morency、Taylor Berg-Kirkpatrick、Kate Saenko、Dan Klein 和 Trevor Darrell。用于视觉与语言导航的说话者-跟随者模型。载于 NeurIPS，2018年。1, 2, 5, 6, 7[28] Tsu-Jui Fu、Xin Wang、Matthew Peterson、Scott Grafton、Miguel Eckstein 和 William Yang Wang。通过对抗性路径采样实现反事实视觉与语言导航。载于 ECCV，2020 2。Robert Goeddel 和 Edwin Olson。 [29] Dart：一种用于生成易于遵循的路线指引的基于粒子的方法。载于智能机器人与系统国际会议，2012年。2[30] Pierre-Louis Guhur、Makarand Tapaswi、Shizhe Chen、Ivan Laptev 和 Cordelia Schmid。Airbert：面向视觉与语言导航的领域内预训练。载于 ICCV，2021年。2, 3, 5, 6, 10[31] Weituo Hao、Chunyuan Li、Xiujun Li、Lawrence Carin 和 Jianfeng Gao。通过预训练学习用于视觉与语言导航的通用智能体。载于 CVPR，2020年。2, 3, 5, 6 [32] Yicong Hong、Cristian Rodriguez-Opazo、Yuankai Qi、Qi Wu 和 Stephen Gould。用于智能体导航的语言与视觉实体关系图。载于 NeurIPS，2020年。5[33] Yicong Hong、Qi Wu、Yuankai Qi、Cristian Rodriguez-Opazo 和 Stephen Gould。用于导航的循环视觉与语言BERT。载于 CVPR，2021年。2, 3, 5, 6, 9, 10[34] Xiaowei Hu、Zhe Gan、Jianfeng Wang、Zhengyuan Yang、

Zicheng Liu、Yumao Lu 和 Lijuan Wang。扩展视觉语言预训练以用于图像描述。载于 CVPR，2022年。5[35] Haoshuo Huang、Vihan Jain、Harsh Mehta、Alexander Ku、Gabriel Magalhaes、Jason Baldridge 和 Eugene Ie。视觉与语言导航中的可迁移表征学习。载于 ICCV，2019年。2[36] Zanming Huang、Zhongkai Shangguan、Jimuyang Zhang、Gilad Bar、Matthew Boyd 和 Eshed Ohn-Bar。Assister：通过条件指令生成实现辅助导航。载于 ECCV，2022 2。Alycia M Hund 和 Jennifer L Minarik[37]。从这里到那里：空间焦虑、寻路策略、方向类型与寻路效率。《空间认知与计算》，6(3):179–201，2006年。2[38] Vihan Jain、Gabriel Magalhaes、Alexander Ku、Ashish Vaswani、Eugene Ie 和 Jason Baldridge。沿路径前行：视觉与语言导航中的指令忠实度。载于 ACL，2019年。2, 5, 6, 7[39] Liyiming Ke、Xiujun Li、Yonatan Bisk、Ari Holtzman、Zhe Gan、Jingjing Liu、Jianfeng Gao、Yejin Choi 和 Siddhartha Srinivasa。战术回退：通过回溯实现视觉与语言导航中的自我纠正。载于 CVPR，2019年。2[40] Jacob Devlin、Ming-Wei Chang、Kenton 和 Lee Kristina Toutanova。BERT：用于语言理解的深层双向 Transformer 预训练。载于 NAACL，2019 3。Diederik P Kingma[41] 和 Jimmy Ba。Adam：一种随机优化方法。载于国际学习表征会议，2015 5。Benjamin Kuipers[42]。空间知识建模。《认知科学》，2(2):129–153，1978年。2[43] Federico Landi、Lorenzo Baraldi、Marcella Cornia、Massimiliano Corsini 和 Rita Cucchiara。感知、变换与行动：用于视觉与语言导航的多模态注意力网络。arXiv 预印本 arXiv: *1911 12377*, 2020 6。Junnan Li、Dongxu Li、Caiming Xiong 和 Steven Hoi[44]。BLIP：用于统一视觉语言理解与生成的引导式语言图像预训练。载于 ICML，2022年。3[45] Xiujun Li、Chunyuan Li、Qiaolin Xia、Yonatan Bisk、Asli Celikyilmaz、Jianfeng Gao、Noah A Smith 和 Yejin Choi。结合语言预训练与随机采样的鲁棒导航。载于 EMNLP，2019年。3[46] Chen Liang、Wenguan Wang、Tianfei Zhou、Jiaxu Miao、Yawei Luo 和 Yi Yang。用于语言引导视频的局部-全局上下文感知 Transformer。PAMI，2023 3。Chen Liang、Wenguan Wang、Tianfei Zhou 和 Yi Yang[47]。视觉溯因推理。载于 CVPR，2022年。3[48] Chuang Lin、Yi Jiang、Jianfei Cai、Lizhen Qu、Gholamreza Haffari 和 Zehuan Yuan。用于视觉与语言导航的可变长度记忆多模态 Transformer。载于 ECCV，2022 2。Chin-Yew Lin。ROUGE：用于摘要自动评估的工具包[49]。载于 Text Summarization Branches Out，第74–81页，2004年。6[50] Gary Look、Buddhika Kottahachchi、Robert Laddaga 和 Howard Shrobe。用于生成描述性步行路线指引的位置表征。载于智能用户界面国际会议论文集，2005 2。Kristin L Lovelace[51]、Mary Hegarty 和 Daniel R Montello。熟悉和不熟悉环境中优质路线指引的要素。载于《空间信息理论国际会议》，1999 2。Kevin Lynch[52]。《城市意象》。麻省理工学院出版社，1960年。2[53] Chih-Yao Ma、Jiasen Lu、Zuxuan Wu、Ghassan AlRegib、Zsolt Kira、Richard Socher 和 Caiming Xiong。通过辅助进度估计实现自我监控的导航智能体。载于国际学习表征会议，2019。2[54] Chih-Yao Ma、Zuxuan Wu、Ghassan AlRegib、Caiming Xiong 和 Zsolt Kira。懊悔的智能体：通过进度估计实现启发式辅助导航。载于 CVPR，2019 2。Matt MacMahon、Brian Stankiewicz 和 Benjamin Kuipers[55]。知行合一：在路线指令中连接语言、知识与行动。载于人工智能促进协会年会，2006。2[56] Arjun Majumdar、Ayush Shrivastava、Stefan Lee、Peter Anderson、Devi Parikh 和 Dhruv Batra。利用来自网络的图文对改进视觉与语言导航。载于 ECCV，2020 2, 5。Hongyuan Mei、Mohit Bansal 和 Matthew R Walter[57]。倾听、关注并行走：将导航指令映射为动作序列的神经映射。载于人工智能促进协会年会，2016年。2[58] Volodymyr Mnih、Adria Puigdomenech Badia、Mehdi Mirza、Alex Graves、Timothy Lillicrap、Tim Harley、David Silver 和 Koray Kavukcuoglu。深度强化学习的异步方法。载于 ICML，2016年。9[59] Stefan Oßwald、Henrik Kretzschmar、Wolfram Burgard 和 Cyrill Stachniss。学习根据人类示范提供路线指引。载于国际机器人与自动化会议，2014年。1，2[60] Kishore Papineni、Salim Roukos、Todd Ward 和 Wei-Jing Zhu。BLEU：一种机器翻译自动评估方法。载于 ACL，2002 6。Alexander Pashevich、Cordelia Schmid[61] 和 Chen Sun。用于视觉与语言导航的情节式 Transformer。载于 ICCV，2021年。2[62] Yuankai Qi、Zizheng Pan、Shengping Zhang、Anton van den Hengel 和 Qi Wu。用于视觉语言导航的物体与动作感知模型。载于 ECCV，2020年。6[63]Yuankai Qi、Qi Wu、Peter Anderson、Xin Wang、William Yang Wang、Chunhua Shen 和 Anton van den Hengel。REVERIE：真实室内环境中的远程具身视觉指代表达。载于 CVPR，2020年。2，5，6，9，10[64] Yanyuan Qiao、Yuankai Qi、Yicong Hong、Zheng Yu、Peng Wang 和 Qi Wu。HOP：用于视觉与语言导航的历史与顺序感知预训练。载于 CVPR，2022年。2，3，5，6，9，10[65]Alec Radford、Jong Wook Kim、Chris Hallacy、Aditya Ramesh、Gabriel Goh、Sandhini Agarwal、Girish Sastry、Amanda Askell、Pamela Mishkin、Jack Clark 等。通过自然语言监督学习可迁移视觉模型。载于 ICML，2021年。3[66] Alec Radford、Jeffrey Wu、Rewon Child、David Luan、Dario Amodei、Ilya Sutskever 等。语言模型是无监督的多任务学习者。OpenAI博客，1 8 9, 2019 4 ( ): 。Kai-Florian Richter[67] 和 Matt Duckham。最简单的指令：寻找易于描述的导航路线。载于地理信息科学国际会议，2008年。2

[68] Raphael Schumann 和 Stefan Riezler。将地图作为图到文本问题生成地标导航指令。载于 ACL，2021年。2[69] Kristina Striegnitz、Alexandre Denis、Andrew Gargett、Konstantina Garoufi、Alexander Koller 和 Mari¨et Theune。关于在虚拟环境中生成指令的第二届挑战赛（give-2.5）的报告。载于欧洲自然语言生成研讨会，2011 2。Hao Tan 和 Mohit [70] Bansal。LXMERT：从Transformer中学习跨模态编码器表示。载于 EMNLP，2019 3。Hao Tan、Licheng Yu 和 Mohit Bansal。[71]学习在未见环境中导航：利用环境丢弃进行反向翻译。载于 NAACL，2019。1，2，6，7[72]Stefanie Tellex、Thomas Kollar、Steven Dickerson、Matthew R Walter、Ashis Gopal Banerjee、Seth Teller 和 Nicholas Roy。理解用于机器人导航和移动操作的自然语言命令。载于人工智能促进协会年会，2011年。1，2[73] Jesse Thomason、Michael Murray、Maya Cakmak 和 Luke Zettlemoyer。视觉与对话导航。载于机器人学习大会，2020 1, 2。Eric J Vanetti 和 Gary L Allen。[74]传达环境知识：语言和空间能力对路线指引生成与理解的影响。《环境与行为》，20(6):667–682，1988年。2[75] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。Attention Is All You Need。载于 NeurIPS，2017年。4[76]Ramakrishna Vedantam、C Lawrence Zitnick 和 Devi Parikh。CIDEr：基于共识的图像描述评估。载于 CVPR，2015 6。David Waller 和 Yvonne Lippa。地 [77] 标作为信标和联想线索：它们在路线学习中的作用。《记忆与认知》，35(5):910–924，2007年。2[78] Hanqing Wang、Wei Liang、Jianbing Shen、Luc Van Gool 和 Wenguan Wang。用于视觉语言导航中指令跟随与生成的反事实循环一致学习。载于 CVPR，2022年。1，2，3，5，6，7[79] Hanqing Wang、Wei Liang、Luc Van Gool 和 Wenguan Wang。迈向通用具身导航。载于 NeurIPS，2022年。2[80] Hanqing Wang、Wenguan Wang、Wei Liang、Caiming Xiong 和 Jianbing Shen。用于视觉语言导航的结构化场景记忆。载于 CVPR，2021年。2[81] Hanqing Wang、Wenguan Wang、Tianmin Shu、Wei Liang 和 Jianbing Shen。用于视觉语言导航的主动视觉信息收集。载于 ECCV，2020年。2，6[82] Su Wang、Ceslee Montgomery、Jordi Orbay、Vighnesh Birodkar、Aleksandra Faust、Izzeddin Gur、Natasha Jaques、Austin Waters、Jason Baldridge 和 Peter Anderson。少即是多：根据地标生成基于实体的导航指令。载于 CVPR，2022年。1[83] Xin Wang、Qiuyuan Huang、Asli Celikyilmaz、Jianfeng Gao、Dinghan Shen、Yuan-Fang Wang、William Yang Wang 和 Lei Zhang。用于视觉语言导航的强化跨模态匹配与自监督模仿学习。载于 CVPR，2019。2，5，6，10[84] Xin Wang、Wenhan Xiong、Hongmin Wang 和 William Yang Wang。在跃起之前先观察：通过弥合无模型与基于模型的强化学习，实现前瞻式视觉与语言导航。载于 ECCV，2018 2。Xiaohan [85] Wang、Linchao Zhu 和 Yi Yang。T vlad：用于文本-视频检索的全局-局部序列对齐。载于 CVPR，2021年。3[86] Xiaohan Wang、Linchao Zhu、Zhedong Zheng、Mingliang Xu 和 Yi Yang。对齐并描述：通过局部对齐和细粒度监督提升文本-视频检索。IEEE多媒体汇刊，2022年。3[87] Zirui Wang、Jiahui Yu、Adams Wei Yu、Zihang Dai、Yulia Tsvetkov 和 Yuan Cao。Simvlm：采用弱监督的简易视觉语言模型预训练。载于国际学习表征会议，2021 3。Shawn [88] L Ward、Nora Newcombe 和 Willis F Overton。在教堂处左转，或向北走三英里：一项关于方向指引与性别差异的研究。《环境与行为》，18 2 192 213, 1986 2 ( ): – 。Ronald J Williams 和 David [89] Zipser。一种持续运行的全循环神经网络学习算法。《神经计算》，1989年。5[90] Jiahui Yu、Zirui Wang、Vijay Vasudevan、Legg Yeung、Mojtaba Seyedhosseini 和 Yonghui Wu。Coca：对比式图像描述模型是图像-文本基础模型。arXiv 预印本 arXiv: *2205 01917*, 2022 3。 . Shuai Zhao、Linchao Zhu、 [91] Xiaohan Wang 和 Yi Yang。Centerclip：通过令牌聚类实现高效文本-视频检索。载于国际信息检索研究与开发会议，2022年。3[92] Yusheng Zhao、Jinyu Chen、Chen Gao、Wenguan Wang、Lirong Yang、Haibing Ren、Huaxia Xia 和 Si Liu。面向目标的结构化 Transformer 规划器，用于视觉语言导航。载于 ACM多媒体会议，2022 2, 3, 4。Fengda Zhu、Yi Zhu、 [93] Xiaojun Chang 和 Xiaodan Liang。通过自监督辅助推理任务实现视觉语言导航。载于 CVPR，2020年。2, 3, 6