🗓️ 10 天 VLA 学习计划：面向工业机械臂模糊指令语义消歧

> 📌 **目标**：用最少的知识点 + 最少的论文，掌握 2024-2025 年 VLA/具身智能主流技术栈 🎯 **课题方向**：面向工业机械臂的模糊指令语义消歧 📚 **已读基础**：SayCan、Inner Monologue、VIMA、VLA Survey

---

🗺️ 学习路线总览

```
Day 1-2: Grounding DINO ──→ "找出那个"
        │
Day 3:   Ask-to-Clarify ──→ "问一句"
        │
Day 4-5: OpenVLA ──────────→ "能跑的基线"
        │
Day 6:   RT-X ─────────────→ "数据怎么组织"
        │
Day 7-8: Diffusion Policy ─→ "多模态动作"
        │
Day 9-10: RT-2 / PaLM-E ──→ "VLM→Policy 范式"
```

📖 Day 1-2：Visual Grounding 模块

🎯 学习目标

掌握如何用自然语言找到图像中的目标物体，这是消歧的第一步——**先找出"那个"可能是什么**

📚 Day 1 上午：知识点学习

主题：Open-vocabulary Perception & Visual Grounding（开放词汇感知 + 视觉指代落地）

核心概念：

概念	中文	解释
Open-set Detection	开放集检测	不限于预定义类别，能检测任意物体
Referring Expression Comprehension (REC)	指代表达理解	根据文本描述找到**那个特定实例**
Grounding	落地/接地	把语言描述和视觉区域对应起来

重点理解：

- "检测" vs "指代落地"：前者找**类别**（所有螺丝），后者找**文本描述对应的那个实例**（左边那颗红色螺丝）
- 工具链：GroundingDINO（找候选）→（可选）SAM 分割 → 作为后续规划的输入

📄 Day 1 下午：论文精读

论文：Grounding DINO (2023)

- 链接：https://arxiv.org/abs/2303.05499
- 代码：https://github.com/IDEA-Research/GroundingDINO

必读部分：

- [ ] Figure 1-2：整体架构图
- [ ] Section 3：语言如何融合进检测器
- [ ] Section 4：RefCOCO/RefCOCO+/RefCOCOg 实验结果

可跳过：大规模训练的工程细节

与课题的联系：

> "把那个螺丝拧到那里" → Grounding DINO 输出 Top-K 候选螺丝和孔位

📄 Day 2 上午：论文补充

论文：Grounding DINO 1.5 (2024)

- 链接：https://arxiv.org/abs/2405.10300

快速浏览：看改进了什么、性能提升多少（30分钟即可）

🧪 Day 2 下午：代码实践

任务：跑通 Grounding DINO 的 demo

```bash
# 安装
git clone https://github.com/IDEA-Research/GroundingDINO.git
cd GroundingDINO
pip install -e .

# 下载权重
# 见 GitHub README
```

实验：

1. 准备一张工业场景图片（或网上找）
2. 输入 prompt："the red screw" 或 "左边那个螺丝"

3. 观察输出的检测框和置信度

✅ Day 1-2 检验标准

- [ ] 能解释 "检测" 和 "指代落地" 的区别
- [ ] 能对一张图片 + 模糊描述，输出 Top-K 候选物体
- [ ] 理解为什么 Grounding 是消歧的前置步骤

📖 Day 3：消歧交互模块

🎯 学习目标

掌握**当指令模糊时，系统如何决定"问一句"**，以及问什么问题最有效

📚 Day 3 上午：知识点学习

主题：Ambiguity-aware Interaction（面向歧义的交互）

核心概念：

概念	中文	解释
Ambiguity Detection	歧义检测	系统判断"我不确定"的能力
Clarifying Question	澄清问题	向用户提问以消除歧义
Information Gain	信息增益	一个问题能减少多少不确定性

重点理解：

- **何时触发澄清**：当 Top-K 候选的置信度接近时
- **问什么问题**：选择**信息增益最大**的问题（一个问题能区分最多候选）
- **评估指标**：
  - 平均澄清轮数（越少越好）
  - 澄清后成功率（越高越好）
  - 误澄清率（问了反而更糟，越低越好）

📄 Day 3 下午：论文精读

论文：Ask-to-Clarify (2025)

- 链接：https://arxiv.org/abs/2509.15061

必读部分：

- [ ] 澄清触发条件：什么时候系统该问
- [ ] 问句生成策略：怎么生成一个好问题
- [ ] 澄清结果如何喂回 action generation

与课题的联系：

> 这就是你课题"模糊指令语义消歧"的**核心参考**：先问清楚再动手

✅ Day 3 检验标准

- [ ] 能说出 3 种触发澄清的条件
- [ ] 能设计一个简单的澄清问题（如"你指的是左边还是右边那颗？"）
- [ ] 理解"信息增益"的直觉含义

📖 Day 4-5：VLA 基线模块

🎯 学习目标

理解 VLA 的核心训练范式，并**跑通一个能用的基线**

📚 Day 4 上午：知识点学习

主题：Offline Imitation Learning + Behavior Cloning（离线模仿学习 + 行为克隆）

核心概念：

概念	中文	解释
Behavior Cloning (BC)	行为克隆	直接模仿专家动作，监督学习
Distribution Shift	分布偏移	模型犯错后进入"没见过"的状态
Compounding Error	累积误差	小错误滚雪球变成大错误

BC 的目标函数：

$$L_{BC} = -\mathbb{E}_{(s,a)\sim D}[\log \pi_\theta(a|s)]$$

即：最大化专家动作的似然 = 最小化负对数似然（NLL）

评估指标：

- Success Rate（成功率）
- 子目标成功率
- 交互轮次（消歧任务很关键！）

📄 Day 4 下午 + Day 5 上午：论文精读

论文：OpenVLA (2024)

- 链接：https://arxiv.org/abs/2406.09246
- 代码：https://github.com/openvla/openvla

必读部分：

- [ ] Figure 2：模型架构（DINOv2 + SigLIP + Llama）
- [ ] Section 3：训练数据来源、数据格式
- [ ] Section 4：微调方法（LoRA 等参数高效微调）
- [ ] Section 5：评估任务和指标

可跳过：所有 baseline 的详细对比数据

与课题的联系：

> OpenVLA 是你做实验的**强基线**：开源、能跑、有微调代码

🧪 Day 5 下午：代码实践

任务：跑通 OpenVLA 推理

```python
# 参考 GitHub README
from openvla import OpenVLA

model = OpenVLA.from_pretrained("openvla-7b")
action = model.predict(image, instruction="pick up the red block")
```

实验：

1. 准备一张简单场景图片
2. 输入不同的语言指令
3. 观察输出的动作

✅ Day 4-5 检验标准

- [ ] 能写出 BC 的损失函数并解释每一项
- [ ] 能解释"分布偏移"为什么是 BC 的核心问题
- [ ] 能跑通 OpenVLA 的推理 demo

📖 Day 6：数据范式模块

🎯 学习目标

理解**多机器人数据如何统一格式**，为你未来构建工业数据集做准备

📄 Day 6 上午：论文精读

论文：Open X-Embodiment & RT-X (2023)

- 链接：https://arxiv.org/abs/2310.08864
- 项目页：https://robotics-transformer-x.github.io/

必读部分：

- [ ] Section 2：数据集统一格式（不同机器人的 demonstration 怎么对齐）
- [ ] Section 3：Cross-robot transfer 为什么成立
- [ ] Table 1：数据集统计

可跳过：具体模型的 ablation 实验

与课题的联系：

> 消歧任务需要你**自己构建数据**，RT-X 给你"数据组织与标准化"的参考范式

📝 Day 6 下午：思考与规划

任务：为你的工业场景设计数据格式

思考这些问题：

1. 你的输入是什么？（图像、语言指令、机器人状态）
2. 你的输出是什么？（动作序列、澄清问题、目标物体）
3. 如何标注"模糊指令"？（哪些指令是模糊的、正确答案是什么）
4. 如何标注"消歧成功"？（什么叫成功消歧）

输出：画一个你数据集格式的草图

✅ Day 6 检验标准

- [ ] 能说出 RT-X 数据格式的核心字段
- [ ] 能画出你工业数据集的初步格式设计

📖 Day 7-8：扩散策略模块

🎯 学习目标

理解**扩散模型如何生成动作**，以及为什么它**天然适合处理模糊指令**

📚 Day 7 上午：知识点学习

主题：Diffusion Policy（扩散策略）

核心概念：

概念	中文	解释
Denoising	去噪	从纯噪声逐步"擦干净"变成动作
Multi-modal Distribution	多模态分布	同一个输入可能对应多种合理输出
Receding Horizon Control	滚动时域控制	每次只执行一小段，然后重新规划

为什么 Diffusion 适合模糊指令：

```
指令："把那个东西放过去"
    │
    ├──→ 动作1：拿红色螺丝放左边  (概率 40%)
    ├──→ 动作2：拿蓝色螺丝放右边  (概率 35%)
    └──→ 动作3：拿扳手放中间      (概率 25%)

传统 BC：只能输出一个"平均"动作（可能哪个都不对）
Diffusion：能表示这个多峰分布，采样时选择其中一个模式
```

📄 Day 7 下午 + Day 8 上午：论文精读

论文：Diffusion Policy (2023)

- 链接：https://arxiv.org/abs/2303.04137
- 项目页：https://diffusion-policy.cs.columbia.edu/

必读部分：

- [ ] Figure 1-2：整体流程图（必须看懂！）
- [ ] Section 3.1：Diffusion 如何条件化在视觉观测上
- [ ] Section 3.2：Receding Horizon 怎么做闭环执行
- [ ] Section 4：与 BC/Transformer policy 的对比实验

可跳过：扩散推导里过细的数学符号（先抓直觉）

与课题的联系：

> "语言模糊 → 多种合理动作"，扩散天然适合表示多模态动作分布

🧪 Day 8 下午：代码实践

任务：跑通 Diffusion Policy 仿真 demo

```bash
# 参考项目页的安装说明
git clone https://github.com/real-stanford/diffusion_policy.git
cd diffusion_policy
# ... 按 README 安装
```

实验：

1. 在 Push-T 或其他简单任务上训练
2. 观察生成的动作轨迹

✅ Day 7-8 检验标准

- [ ] 能用自己的话解释"去噪生成动作"的过程
- [ ] 能解释为什么 Diffusion 比 BC 更适合多模态分布
- [ ] 能解释 Receding Horizon 如何避免误差累积

📖 Day 9-10：大模型范式模块

🎯 学习目标

理解**VLM 如何直接变成 Policy**，这是当前 VLA 的主流范式

📚 Day 9 上午：知识点补充

主题：Affordance / Value Function 再补一层

你已从 SayCan 理解了"能不能做"，这里补充**工业场景的特殊约束**：

约束类型	例子
夹具限制	当前夹爪能不能抓这个形状的物体
工具约束	这个任务需要扳手，但手里拿的是螺丝刀
安全区	这个区域机械臂不能进入
力控约束	这个零件太脆弱，不能用大力

与消歧的联系：

> 两个候选目标都符合语言描述，但只有一个**物理上可抓取** → 用 affordance 消歧

📄 Day 9 下午：论文精读 (1)

论文：RT-2 (2023)

- 链接：https://arxiv.org/abs/2307.15818
- 博客：https://deepmind.google/blog/rt-2

必读部分：

- [ ] Figure 1：核心 idea 一图流
- [ ] Section 2：如何把动作表示成文本 token
- [ ] Section 4.3：Emergent Capabilities（涌现能力）

可跳过：一些离散化动作的工程实现细节

与课题的联系：

> RT-2 证明了 VLM 的**常识**（"苹果是红色的"）可以帮助机器人理解指令

📄 Day 10 上午：论文精读 (2)

论文：PaLM-E (2023)

- 链接：https://arxiv.org/abs/2303.03378
- 项目页：https://palm-e.github.io/

必读部分：

- [ ] Figure 2：如何把连续传感/状态塞进 LLM
- [ ] Section 3：多任务联合训练与正迁移

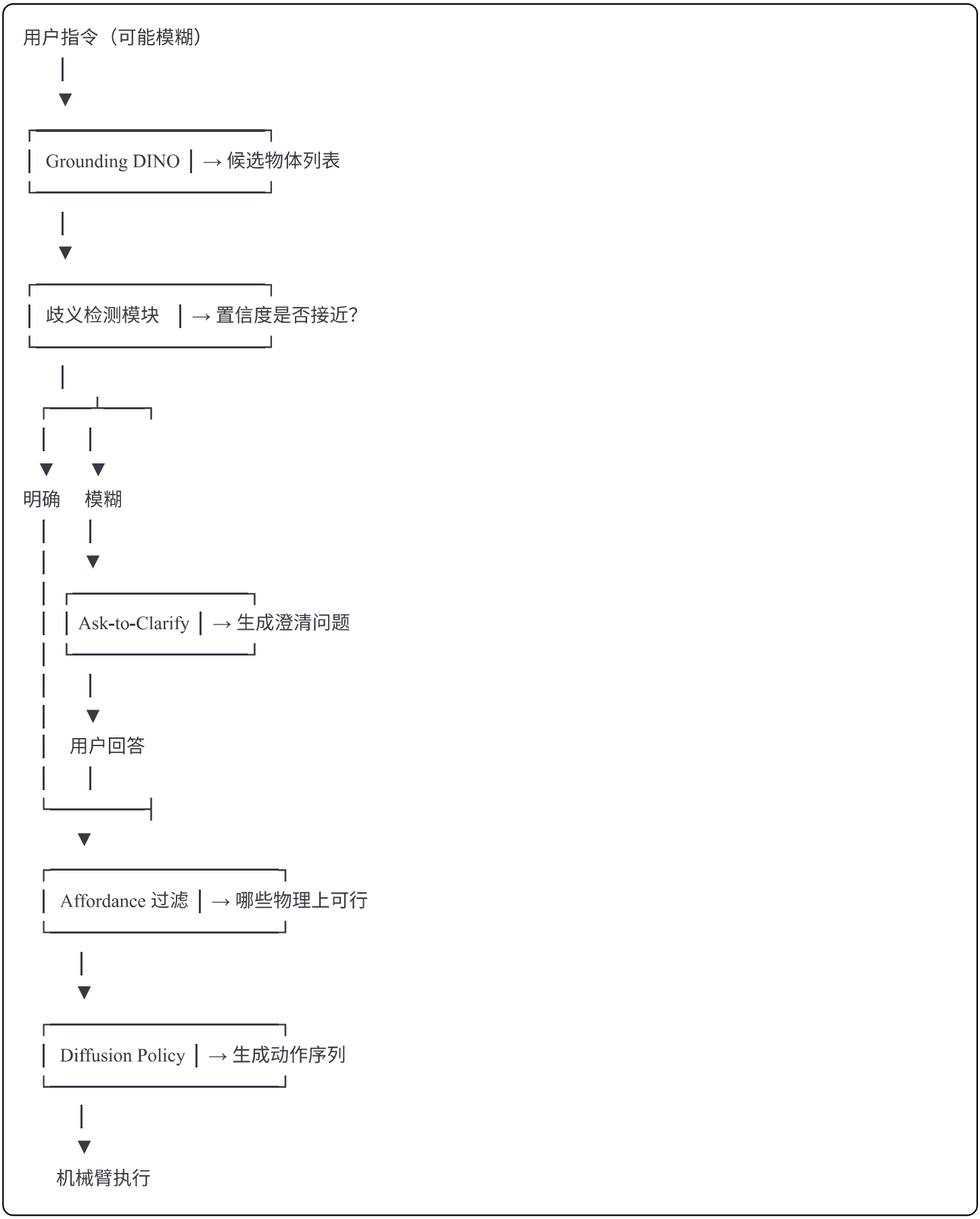
可跳过：超大模型规模的训练细节

与课题的联系：

> 工业机械臂有很多**状态信息**（夹爪开合、力矩、工位编号），PaLM-E 展示了如何把这些纳入 LLM

📝 Day 10 下午：总结与规划

任务：画出你课题的技术路线图



Day 9-10 检验标准

- [ ] 能说出 RT-2 和 PaLM-E 的核心区别
- [ ] 能画出你课题的技术路线图
- [ ] 能列出接下来要做的 3 个"小而硬"实验

🧪 学完后立刻能做的 3 个实验

实验 1：候选生成实验（Grounding）

目标：对工业场景图片 + 模糊指令，输出 Top-K 候选实体

输入：

- 图片：工业工作台（有多颗螺丝、扳手、孔位）
- 指令："把那个螺丝拧到那里"

输出：

- 候选螺丝列表 + 置信度
- 候选孔位列表 + 置信度

实验 2：澄清问句实验（Disambiguation）

目标：当 Top-K 置信度接近时，自动生成澄清问题

触发条件：Top-1 置信度 < 0.6 或 Top-1 和 Top-2 差距 < 0.1

输出示例：

- "你指的是左边还是右边那颗螺丝？"
- "你想拧到上面的孔还是下面的孔？"

实验 3：执行对比实验（Policy）

目标：比较不同方法的效果

方法	评估指标
BC (自己写)	成功率、澄清轮数、误执行率
OpenVLA (微调)	成功率、澄清轮数、误执行率
Diffusion Policy	成功率、澄清轮数、误执行率

⏰ 每日节奏建议

时段	活动	时长
🌅 上午	📚 知识点 / 📄 论文精读	2-3 小时
🌆 下午	📄 论文续读 / 💉 代码实践	2-3 小时
🌙 晚上	📝 笔记整理 + 思考"和课题怎么联系"	1 小时

📎 资源链接汇总

论文/工具	arXiv	代码/项目页
Grounding DINO	2303.05499	GitHub
Grounding DINO 1.5	2405.10300	-
Ask-to-Clarify	2509.15061	-
OpenVLA	2406.09246	GitHub
RT-X	2310.08864	项目页
Diffusion Policy	2303.04137	项目页
RT-2	2307.15818	博客
PaLM-E	2303.03378	项目页

✅ 10 天总检验清单

- [ ] 能跑通 Grounding DINO，对模糊描述输出候选物体
- [ ] 能设计澄清问题的触发条件和问句模板
- [ ] 能跑通 OpenVLA 推理
- [ ] 能画出你工业数据集的格式草图
- [ ] 能解释 Diffusion Policy 为什么适合多模态分布
- [ ] 能说出 RT-2 和 PaLM-E 的核心区别
- [ ] 能画出你课题的完整技术路线图
- [ ] 能列出接下来要做的 3 个实验

> 💪 **加油！10 天后你就能从"看论文"进入"做实验"阶段了！**
> 有任何问题随时问我 🎓