TaskSense: A Translation-like Approach for Tasking Heterogeneous Sensor Systems with LLMs

SenSys 2025

Kaiwei Liu[1], Bufang Yang[1], Lilin Xu[1], Yunqi Guo[1], **Guoliang Xing[1]**,
Xian Shuai[2], Xiaozhe Ren[2], Xin Jiang[2], Zhenyu Yan[1]

[1]The Chinese University of Hong Kong
[2]Noah's Ark Lab, Huawei Technologies

汇报人：董兵

2025年 6月 13日

Author

Research Interests

- Sensing systems
- Embedded AI
- Cyber-physical systems

Guoliang Xing

Embedded AI and IoT Lab

The Chinese University of Hong Kong

- [1] SHADE-AD: An llm-based framework for synthesizing activity data of Alzheimer’ s patients. SenSys ' 2025
- [2] Socialmind: Llm-based proactive ar social assistive system with human-like perception for in-situ live interactions. Ubicomp ' 2025
- [3] Drhouse: An llm-empowered diagnostic reasoning system through harnessing outcomes from sensor data and expert knowledge. Ubicomp ' 2024

提纲

1 Background

2 Method

3 Evaluation

4 Conclusion

提纲

1 Background

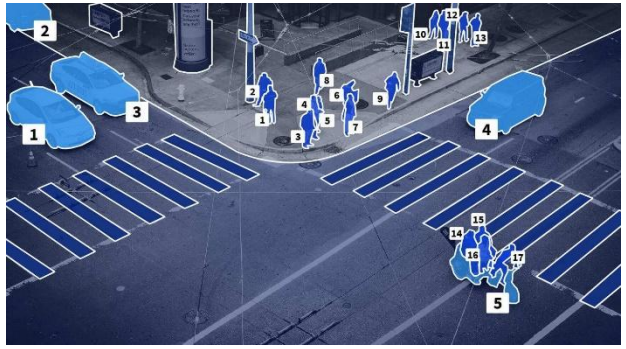
2 Method

3 Evaluation

4 Conclusion

Background：Sensor System

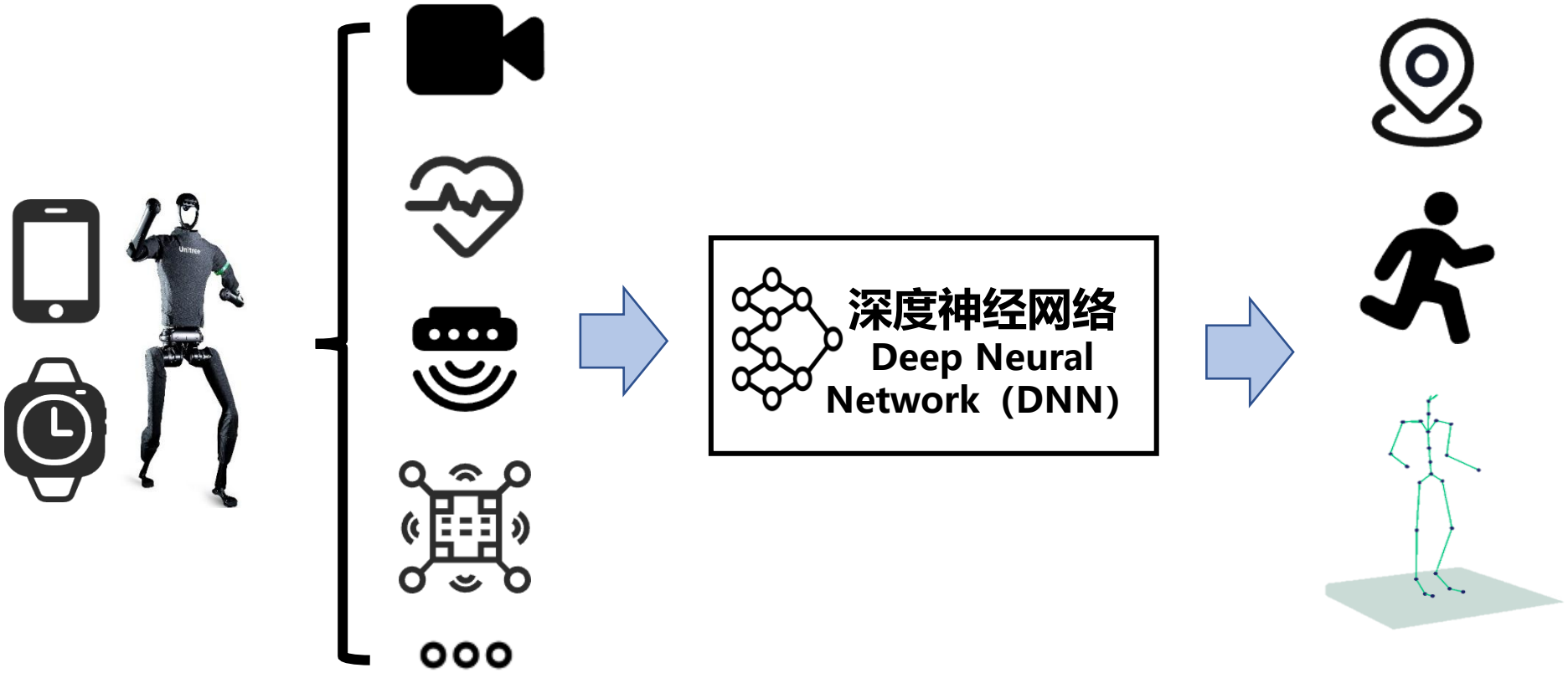
多种感知需求



传感器系统在各个领域应用广泛

Background：Sensor System

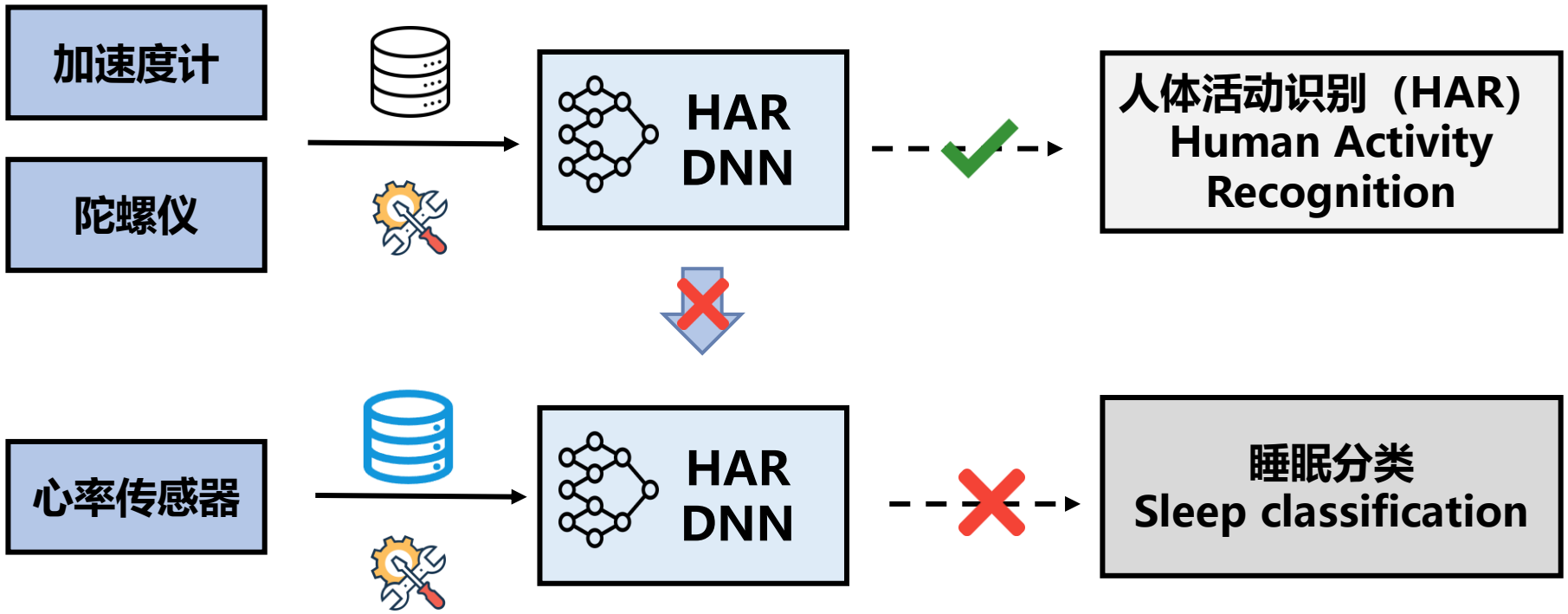
传感器系统工作流程



传感器 -> 数据采集 -> 算法处理 -> 高层次的感知信息

Related Work：Tasking Sensor Systems

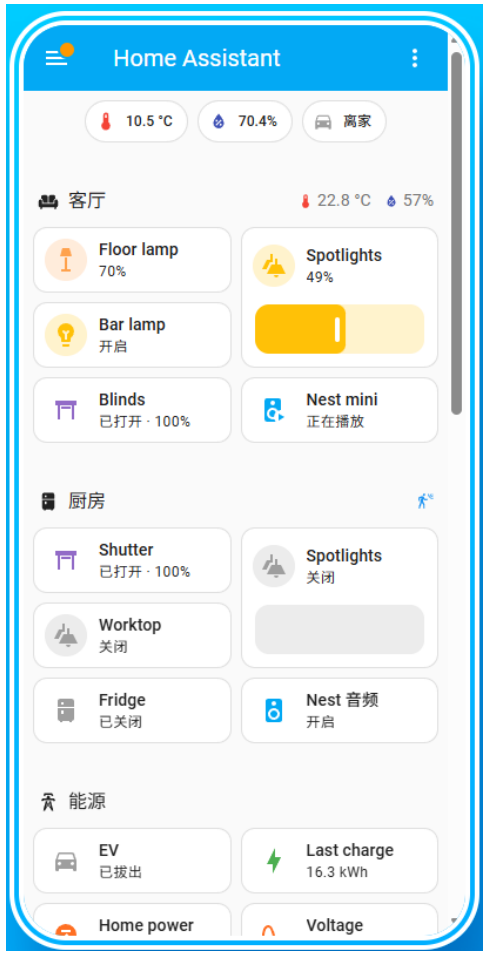
- 大多数传感器系统仅针对专门化任务设计

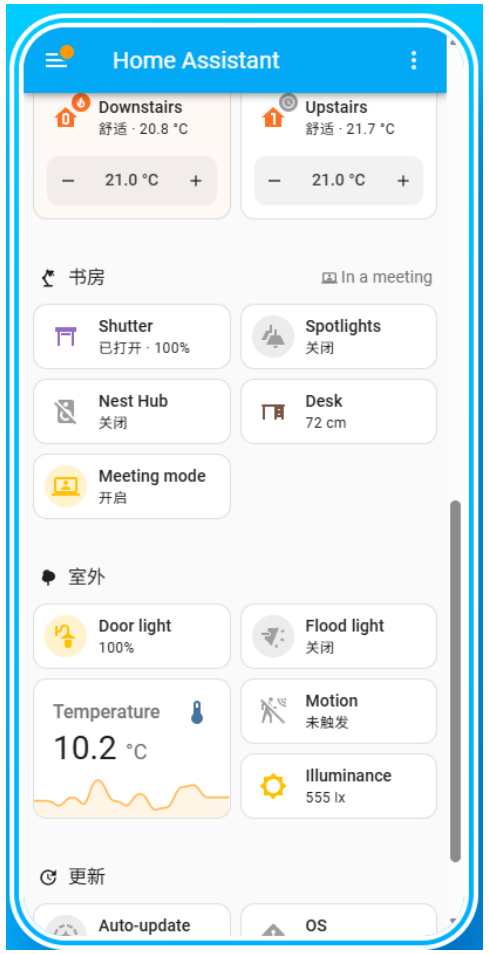


仅限于预定义且固定的任务，无法适应动态变化的用户需求

Related Work：Tasking Sensor Systems

- 大多数传感器系统仅针对专门化任务设计





Home Assistant：整合超过千种不同设备和服务，实现对来自不同厂商的异构智能设备进行统一调度，以处理复杂任务。

- **预定义的固定协作规则。**
- **面对海量潜在设备组合时，难以扩展以应对灵活多变的任务需求。**

Related Work：LLM for SensorQA

- **传感器问答：**
  - 将传感器数据映射到embeding，微调LLM（M4 MobiCom’24，OneLLM CVPR’24）

Related Work：LLM for SensorQA

- **传感器问答：**
  - 将传感器数据映射到embeding，微调LLM（M4 MobiCom’24，OneLLM CVPR’24）

取决于预训练时任务种类，无法覆盖更广泛的任务

Related Work：LLM Agent

- **传感器问答：**
  - 将传感器数据映射到embeding，微调LLM（M4 MobiCom'24，OneLLM CVPR'24）

取决于预训练时任务种类，无法覆盖更广泛的任务

- **LLM Agent：**LLM 作为控制中心，能够**灵活**调用外部工具来解决复杂任务
  - 通用工具：特定任务设计的AI模型等（HuggingGPT NeurIPS'23，ToolLLM ICLR'2024）
  - 家居设备：管理调节家居设备（Sasha UbiComp'24 ），生成IFTTT程序（ChatIoT UbiComp'24 ）

Related Work：LLM Agent

- **传感器问答：**
  - 将传感器数据映射到embeding，微调LLM（M4 MobiCom'24，OneLLM CVPR'24）

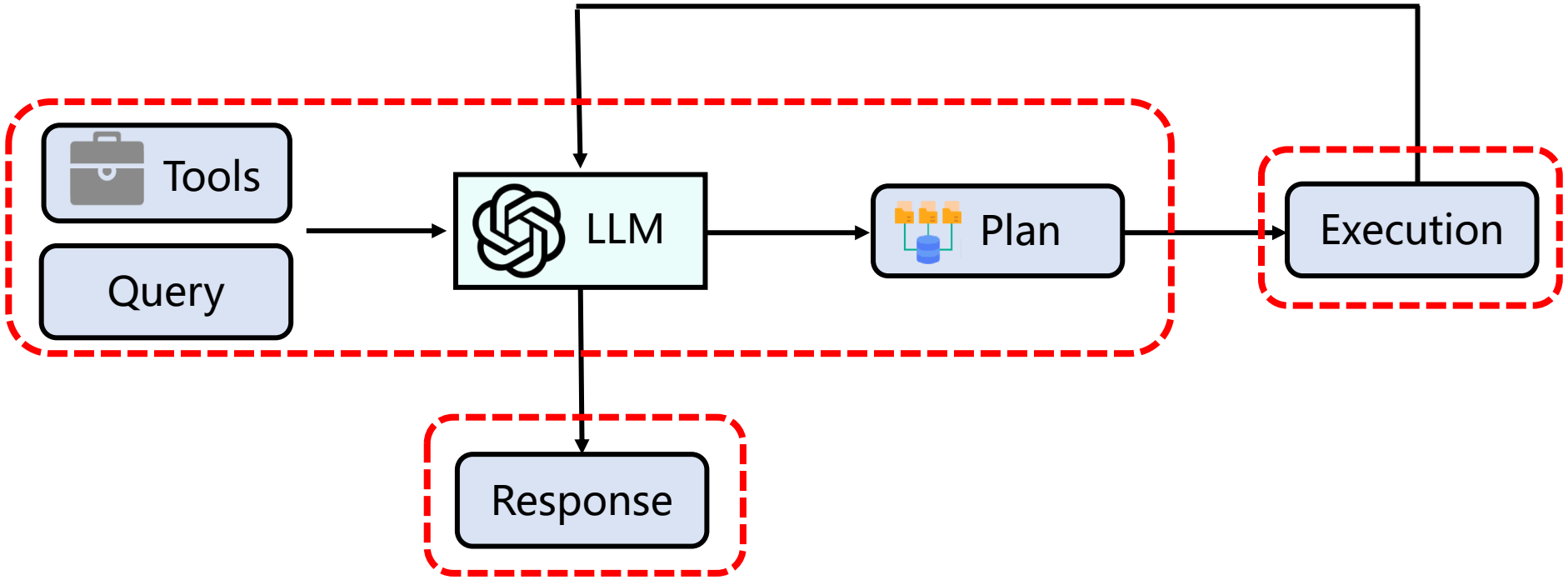
取决于预训练时任务种类，无法覆盖更广泛的任务

- **LLM Agent：**LLM 作为控制中心，能够**灵活**调用外部工具来解决复杂任务
  - 通用工具：特定任务设计的AI模型等（HuggingGPT NeurIPS'23，ToolLLM ICLR'2024）
  - 家居设备：管理调节家居设备（Sasha UbiComp'24 ），生成IFTTT程序（ChatIoT UbiComp'24 ）

未充分利用传感器系统本身所提供的信息和环境影响

Motivation：Measurement studies on LLM Agent

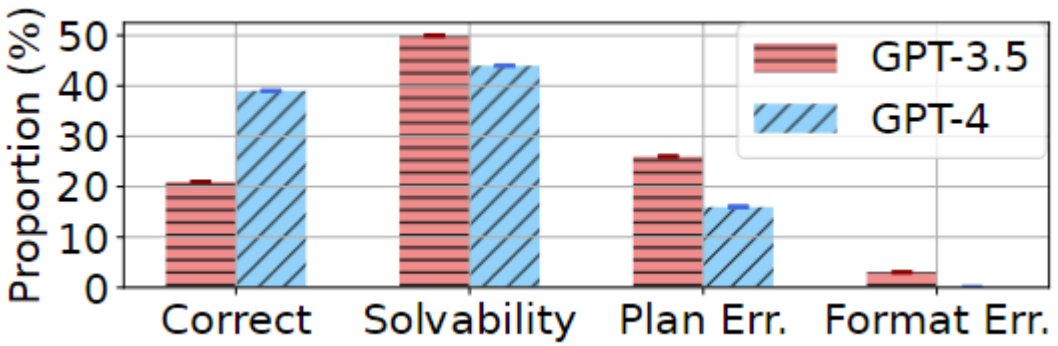
- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复



Motivation1

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

查询到规划：评估HuggingGPT在传感器系统中能力



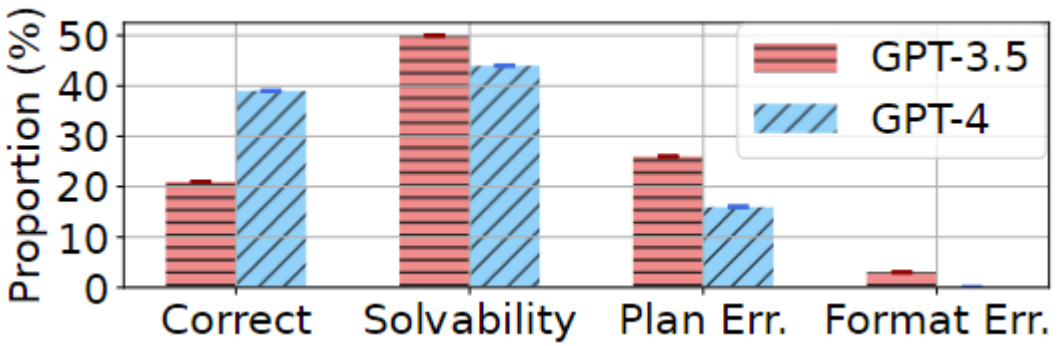
在老年人健康场景中的规划错误分布

包含监控系统、动作识别系统和声音检测系统

Motivation1

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

查询到规划：评估HuggingGPT在传感器系统中能力



在老年人健康场景中的规划错误分布

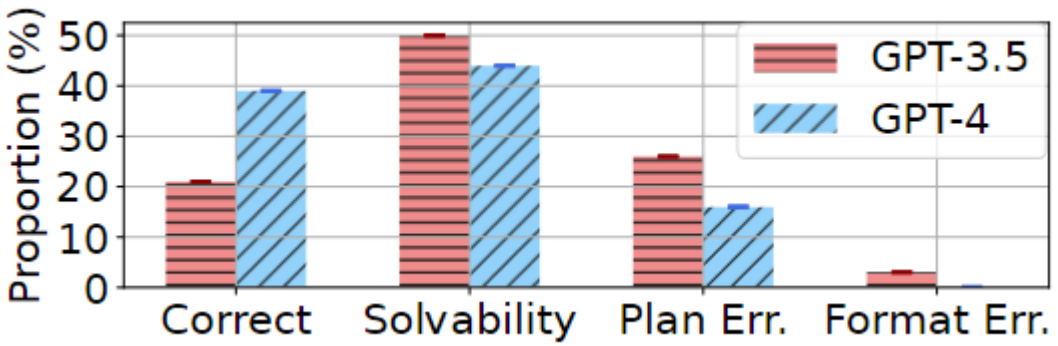
包含监控系统、动作识别系统和声音检测系统

- **Misjudging Solvability**（可解性判断错误）
  - 无法准确识别工具集的能力边界，生成超出能力范围的规划。
  - 如缺乏特定工具，工具标签未覆盖查询目标。

Motivation1

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

查询到规划：评估HuggingGPT在传感器系统中能力



在老年人健康场景中的规划错误分布

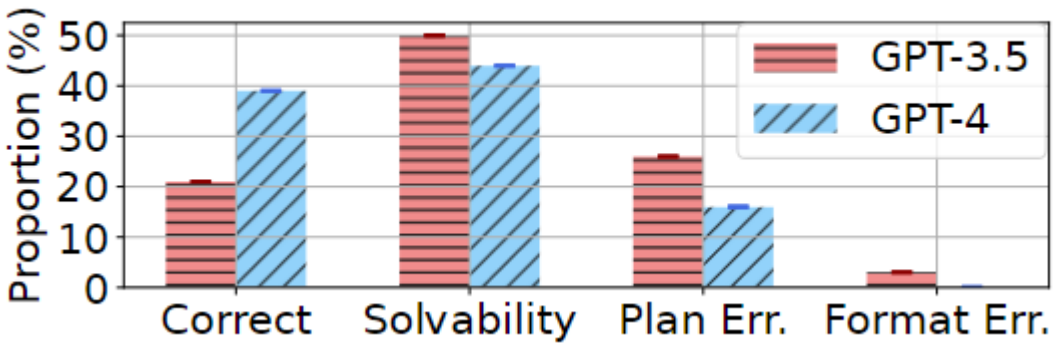
包含监控系统、动作识别系统和声音检测系统

- **Misjudging Solvability**（可解性判断错误）
  - 无法准确识别工具集的能力边界，生成超出能力范围的规划。
  - 如缺乏特定工具，工具标签未覆盖查询目标。
- **Wrong Plan**（错误规划）
  - 生成了依赖错误的规划，工具间的数据依赖受到输入和输出限制。
  - 如人脸识别需要先进行人脸检测。

Challenge1

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

查询到规划：评估HuggingGPT在传感器系统中能力



在老年人健康场景中的规划错误分布

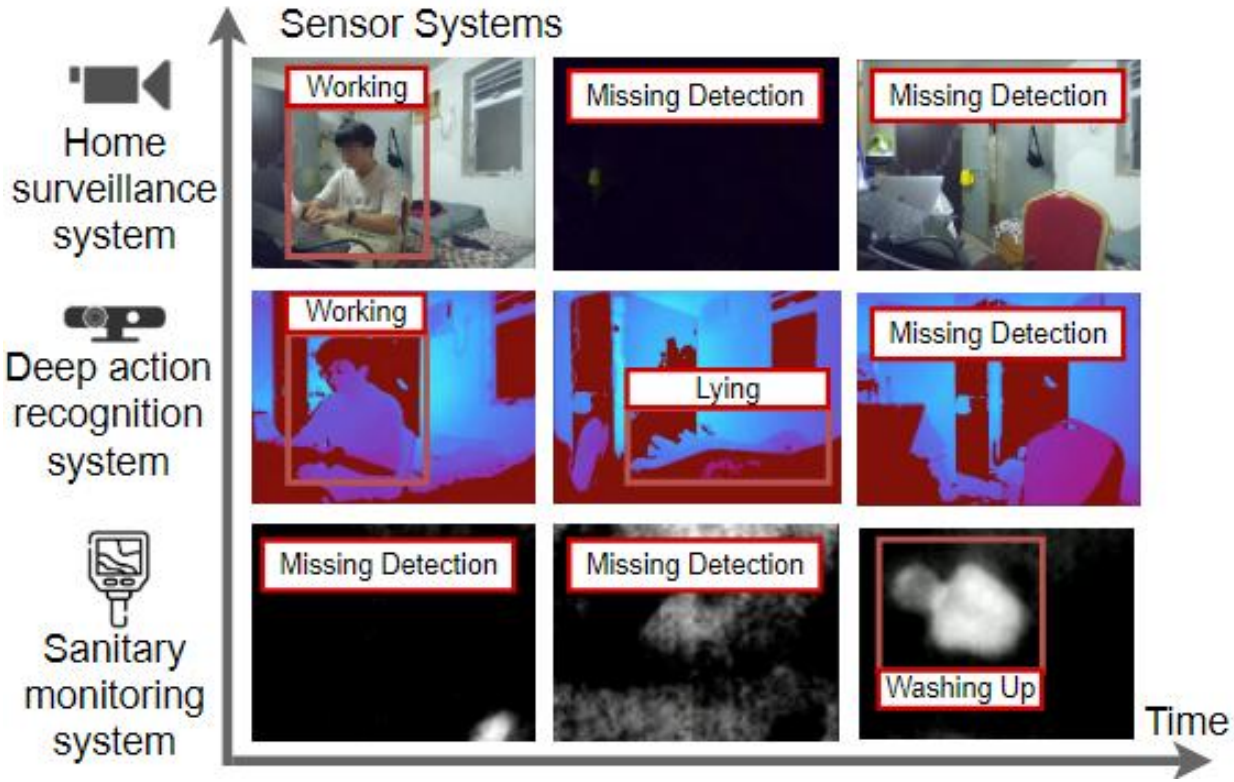
包含监控系统、动作识别系统和声音检测系统

- **Misjudging Solvability**（可解性判断错误）
  - 无法准确识别工具集的**能力边界**，生成超出能力范围的规划。
  - 如缺乏特定工具，工具标签未覆盖查询目标。
- **Wrong Plan**（错误规划）
  - 生成了依赖错误的规划，工具间的**数据依赖**受到输入和输出限制。
  - 如人脸识别需要先进行人脸检测。

Motivation2

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - **执行规划**
  - 组织执行结果生成回复

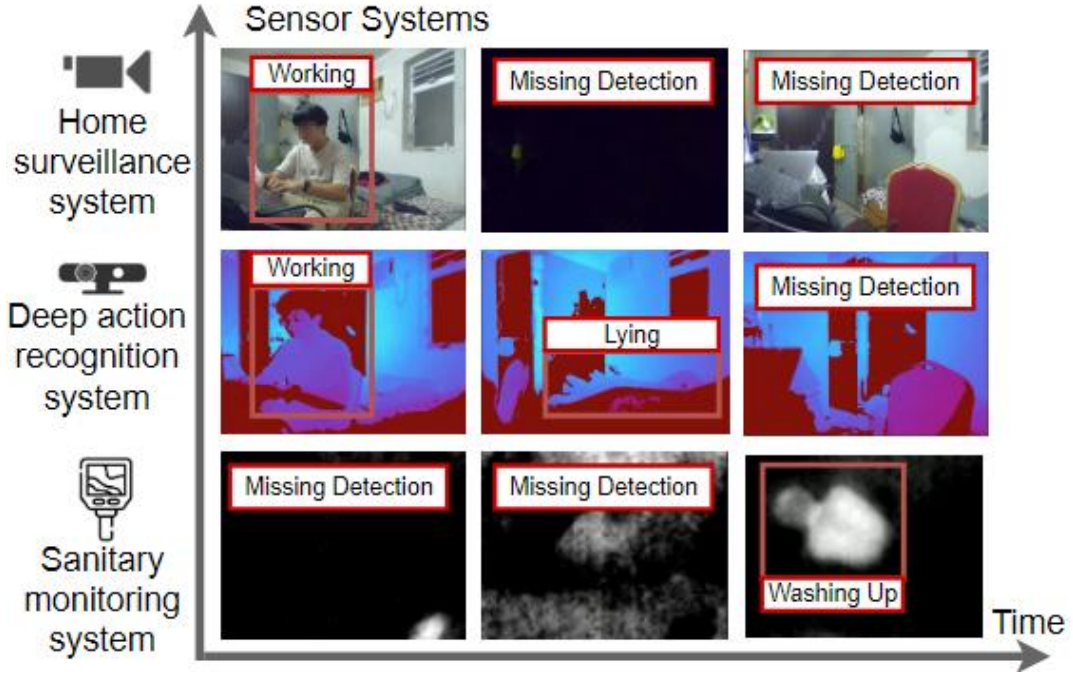
执行计划：计划本身正确，执行结果也会受到环境影响



Motivation2

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

执行规划：规划本身正确，执行结果也会受到环境影响

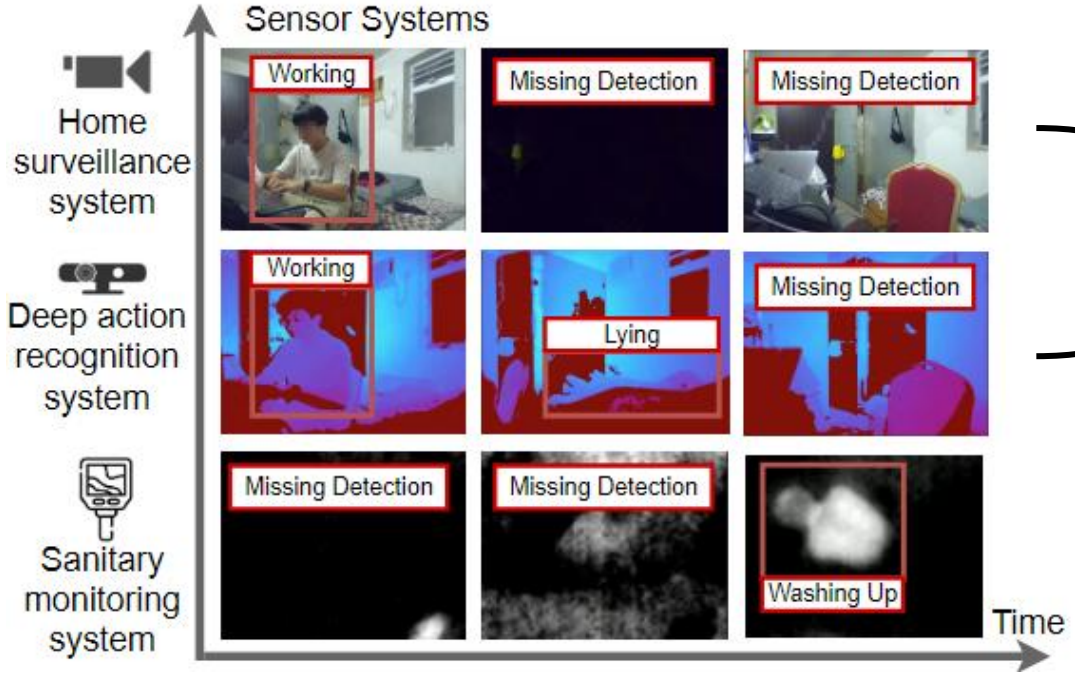


- 数据缺失：传感器因硬件故障或误关而无法记录数据。
- 数据噪声：如光照变化影响 RGB 摄像头，环境噪声干扰语音捕捉，都会降低传感器数据质量。
- 场景相关问题：例如目标移出摄像头视野、遮挡或动作姿态复杂，这类问题无法仅通过噪声水平判断

Motivation2

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

执行规划：规划本身正确，执行结果也会受到环境影响

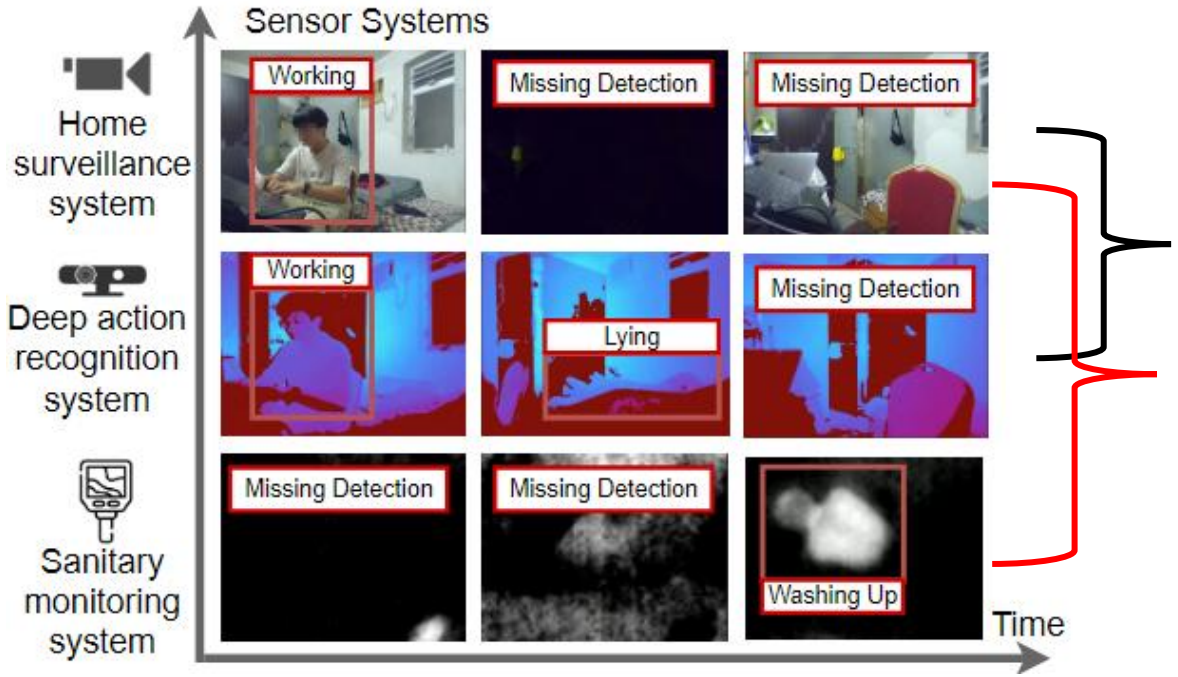


不同规划受到环境影响不同

Motivation2

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

执行规划：规划本身正确，执行结果也会受到环境影响



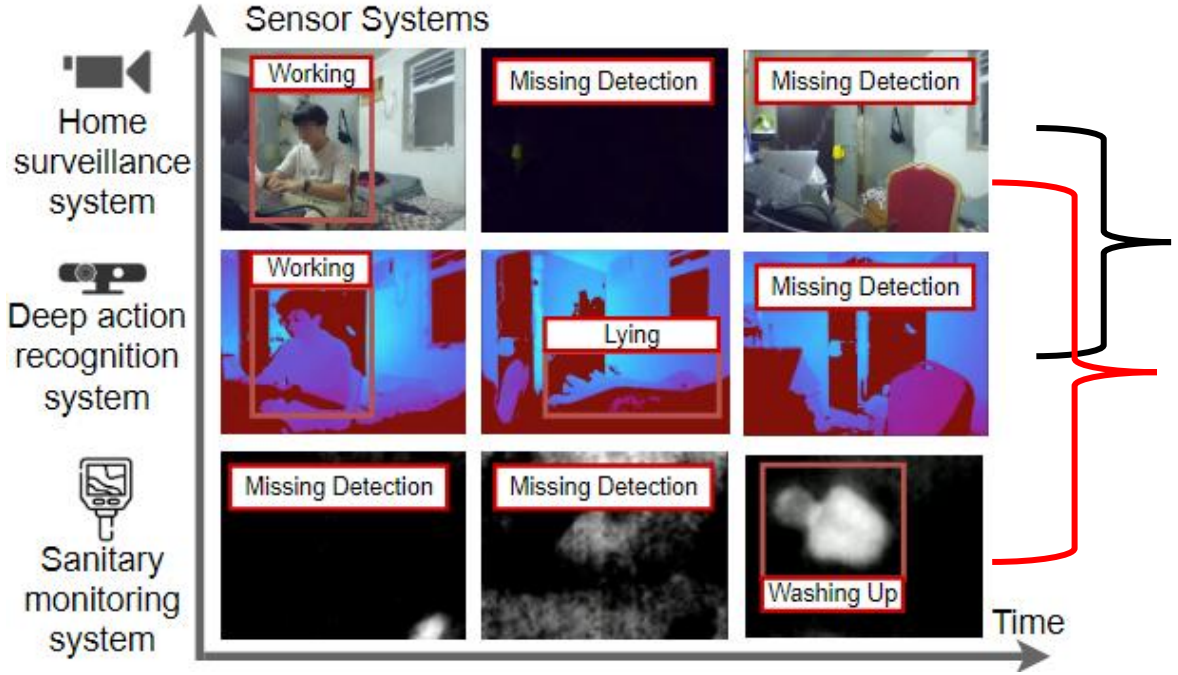
不同规划受到环境影响不同

互补性：多个传感器系统，根据位置切换，保证持续监测

Challenge2

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

执行规划：规划本身正确，执行结果也会受到环境影响



不同规划受到环境影响不同

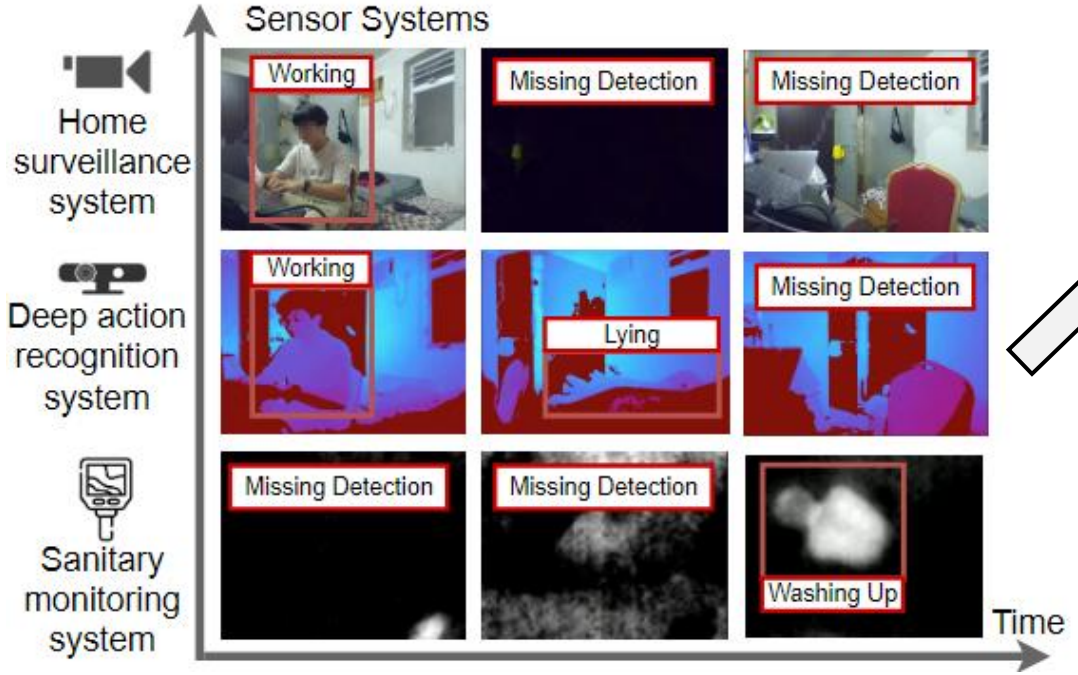
互补性：多个传感器系统，根据位置切换，保证持续监测

环境变化时如何自适应切换执行路径

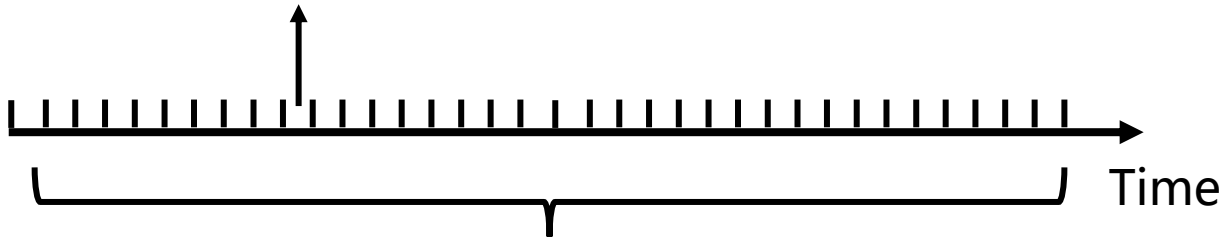
Challenge3

- 传感器系统进行任务调度：
  - 基于用户查询生成工具调用规划
  - 执行规划
  - 组织执行结果生成回复

执行规划：规划本身正确，执行结果也会受到环境影响



每一时间片执行一次 -> **海量数据**



LLM对长序列数据推理时，容易产生“遗忘”和“数据关联错误”等幻觉问题

提纲

1 Background

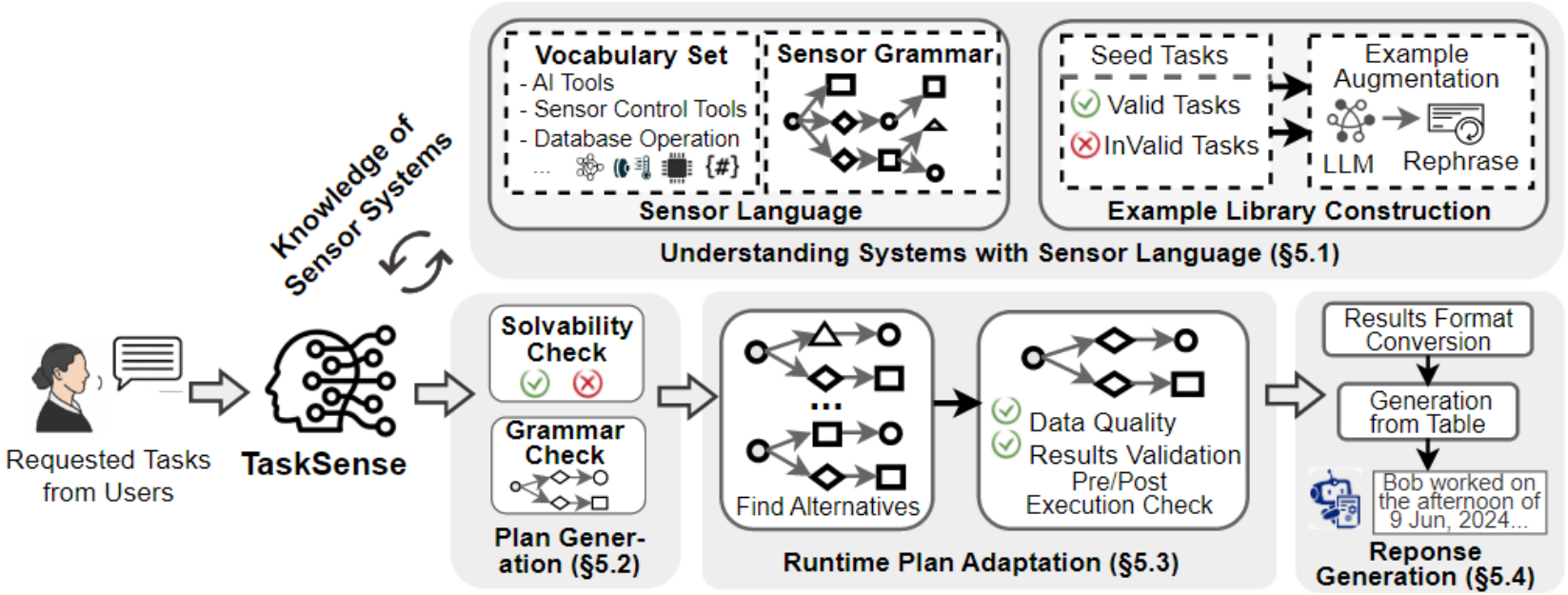
2 Method

3 Evaluation

4 Conclusion

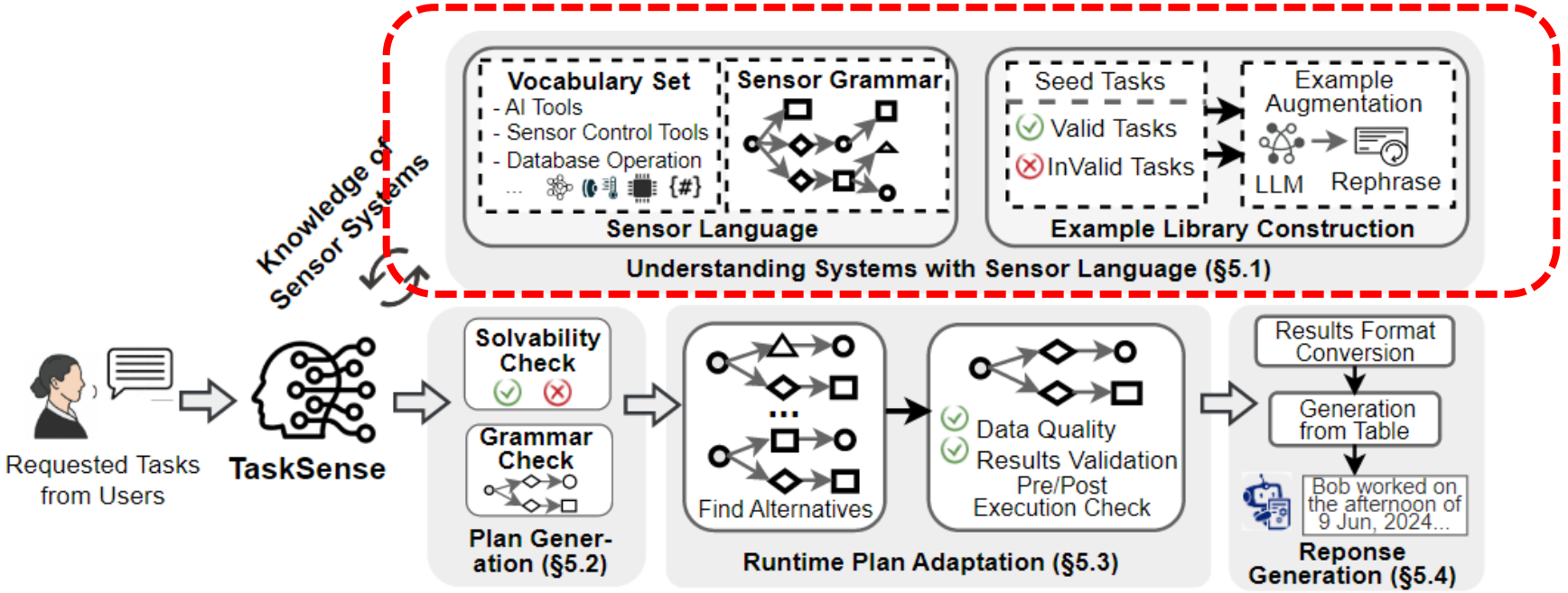
Method：Overview of TaskSense

- **Understanding Systems with Sensor Language**：工具描述和工具间依赖关系图，构建可解示例与不可解示例
- **Plan Generation**：工具能力边界判断 - 可解性检查，工具间依赖关系检查
- **Runtime Plan Adaptation**：运行时，根据执行反馈动态切换可替代路径
- **Response Generation**：将输出结果转换为对象及其标签（如活动、性别、身份）来存储，回复时提取关键信息输入LLM

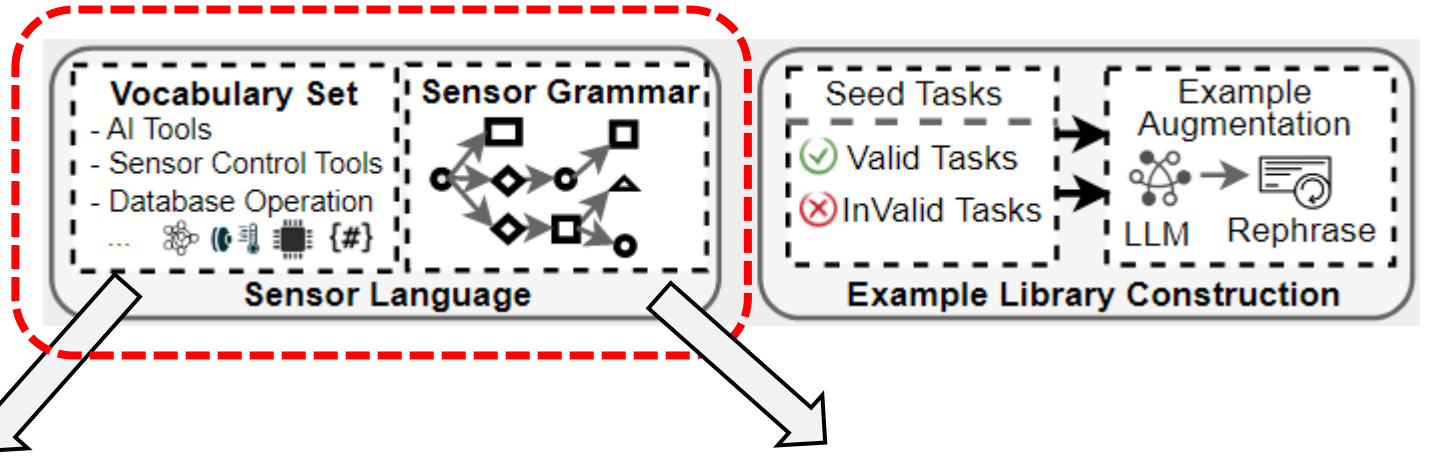


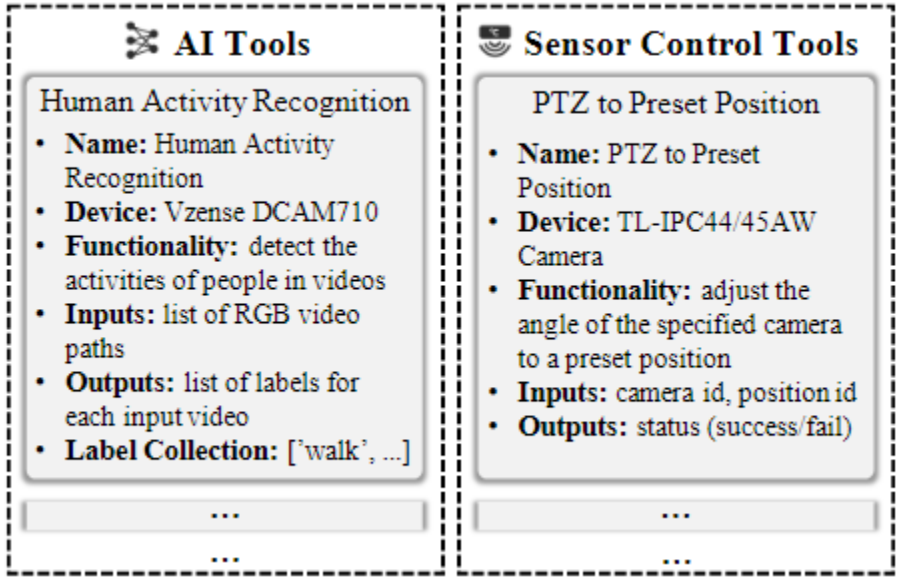
Method：Overview of TaskSense

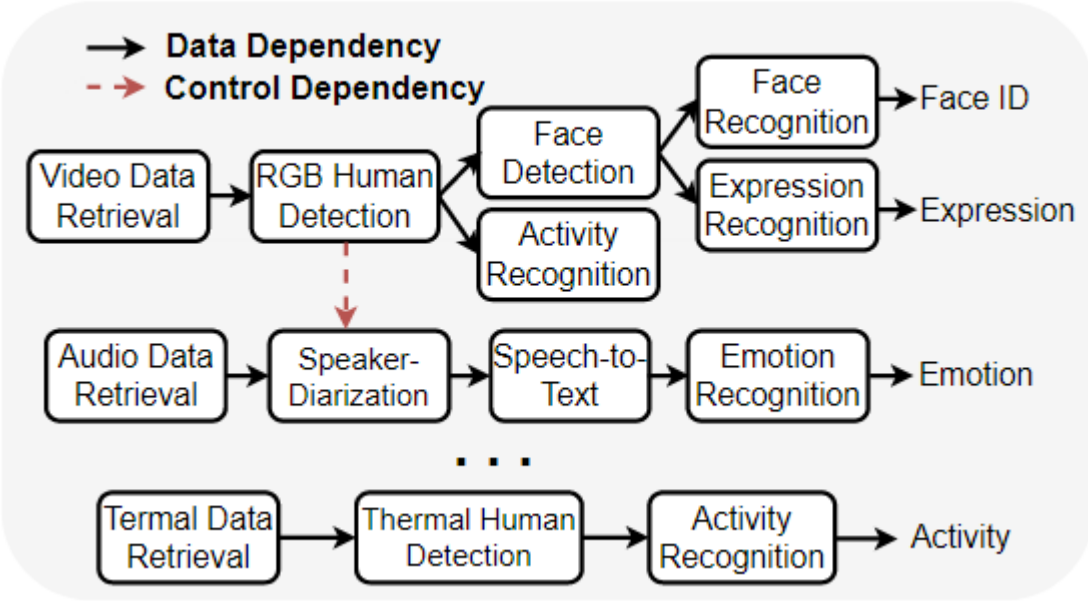
- **Understanding Systems with Sensor Language**：工具描述和工具间依赖关系图，构建可解示例与不可解示例
- **Plan Generation**：工具能力边界判断 - 可解性检查，工具间依赖关系检查
- **Runtime Plan Adaptation**：运行时，根据执行反馈动态切换可替代路径
- **Response Generation**：将输出结果转换为对象及其标签（如活动、性别、身份）来存储，回复时提取关键信息输入LLM



Understanding Systems with Sensor Language



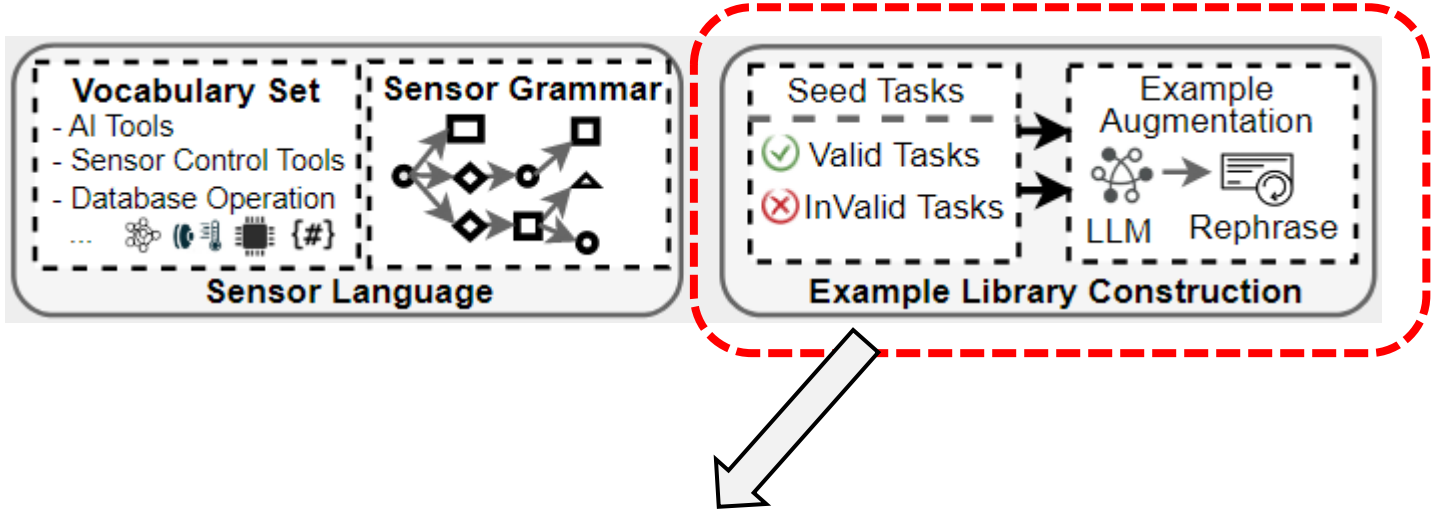




Vocabulary set：**工具结构化描述**

Sensor grammar：**工具间的依赖关系图**

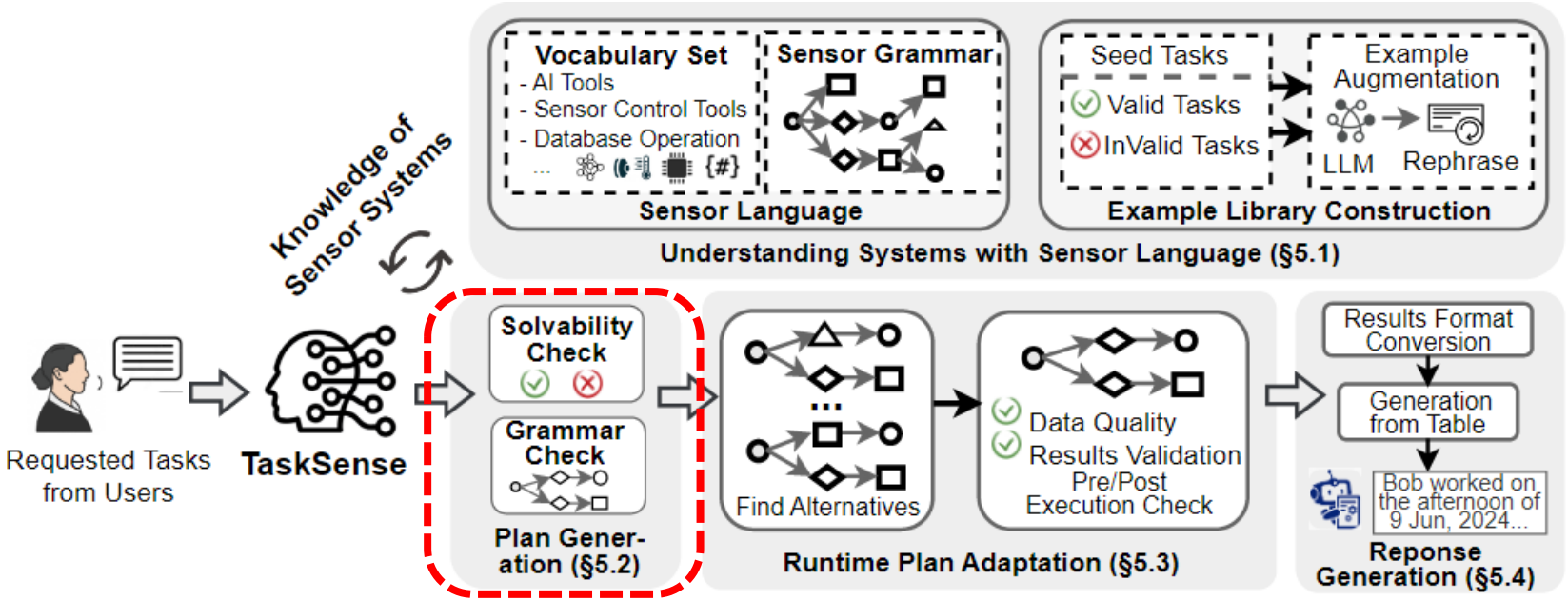
Understanding Systems with Sensor Language



- 构建“查询-规划”示例库
  - **Seed Example Categorization**：为确保示例多样性和覆盖性，根据应用将查询分类，为每个类别创建示例，每个类别内种子示例尽可能互不相同。
  - **Solvable and Unsolvable Examples**：可解示例包含可处理的查询及对应计划，不可解示例包含不可处理的查询和一个空规划。
  - **Example Library Augmentation**：用户往往具有多样的表达风格，利用LLM生成若干语义等价但风格不同的新查询。

Method：Overview of TaskSense

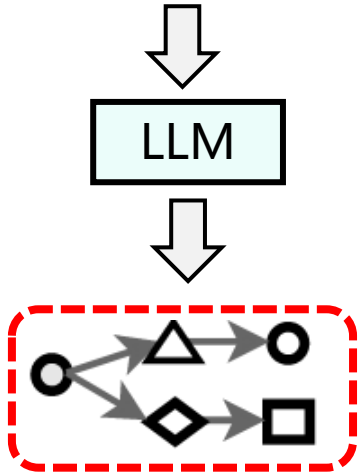
- **Understanding Systems with Sensor Language**：工具描述和工具间依赖关系图，构建可解示例与不可解示例
- **Plan Generation**：工具能力边界判断 - 可解性检查，工具间依赖关系检查
- **Runtime Plan Adaptation**：运行时，根据执行反馈动态切换可替代路径
- **Response Generation**：将输出结果转换为对象及其标签（如活动、性别、身份）来存储，回复时提取关键信息输入LLM



Plan Generation

Challenge 1: 从查询到规划中的可解性错误，工具依赖错误

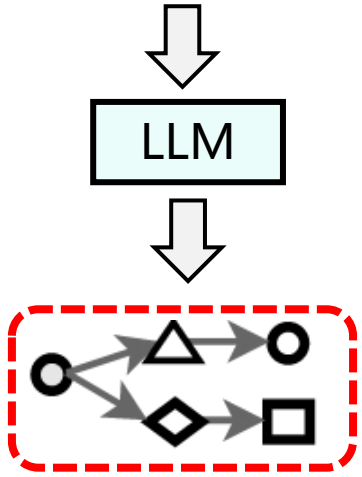
- 查询-规划：
  - 在示例库中基于相似度检索，并在每个查询类别中均匀抽取示例
  - 工具结构化描述（Vocabulary set），工具间的依赖关系图（Sensor grammar）
  - 用户查询

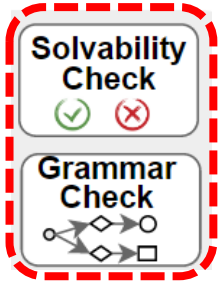


Plan Generation

Challenge 1: 从查询到规划中的可解性错误，工具依赖错误

- 查询-规划：
  - 在示例库中基于相似度检索，并在每个查询类别中均匀抽取示例
  - 工具结构化描述（Vocabulary set），工具间的依赖关系图（Sensor grammar）
  - 用户查询



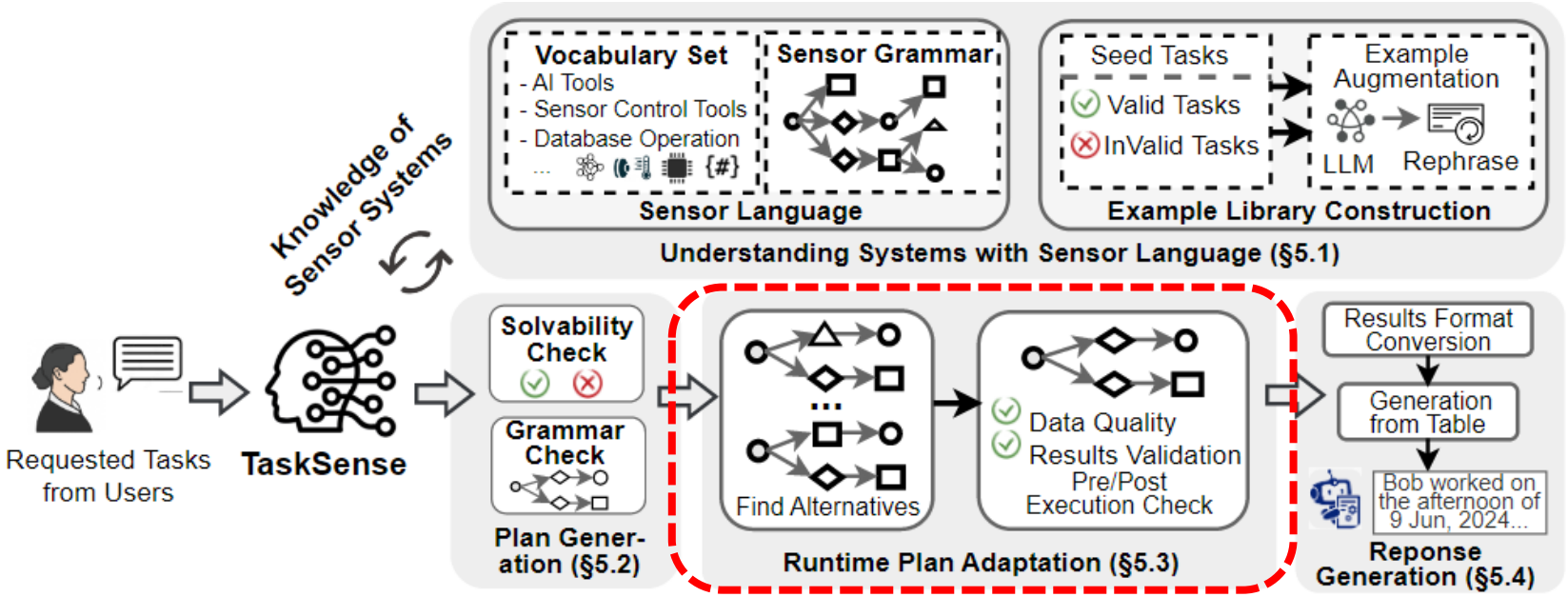


先判断工具集中是否有对应能力的工具，在判断工具的标签是否包含目标标签

子图匹配验证：生成的规划DAG 与 系统中完整的工具依赖图 匹配验证

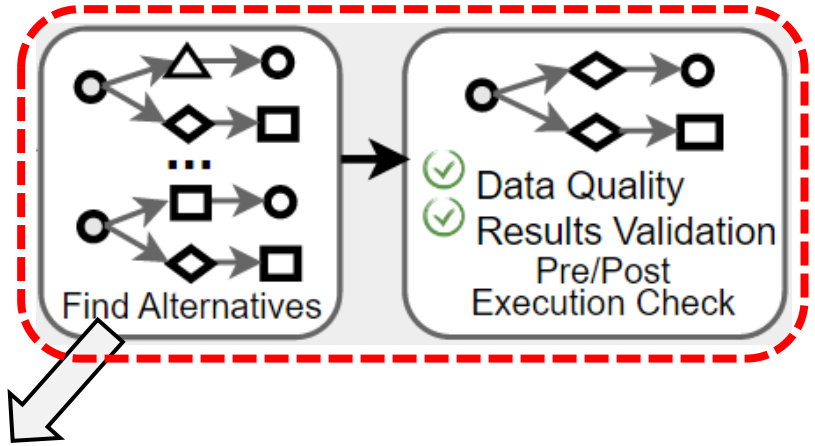
Runtime Plan Adaptation

- **Understanding Systems with Sensor Language**：工具描述和工具间依赖关系图，构建可解示例与不可解示例
- **Plan Generation**：工具能力边界判断 - 可解性检查，工具间依赖关系检查
- **Runtime Plan Adaptation**：运行时，根据执行反馈动态切换可替代路径
- **Response Generation**：将输出结果转换为对象及其标签（如活动、性别、身份）来存储，回复时提取关键信息输入LLM



Runtime Plan Adaptation

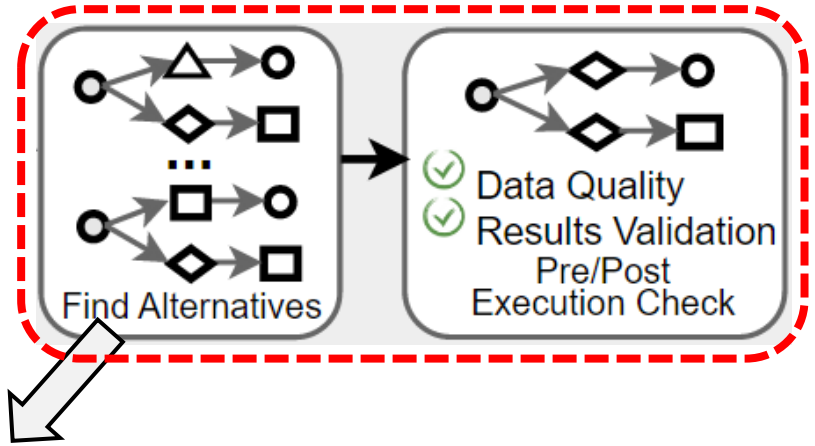
Challenge 2: 环境变化时如何动态切换执行路径，保证执行准确率



- Alternative Path：将功能相似的工具归为一组，对于每个组内的每个工具，构造一条以该工具为终点、只包含其依赖的最小必要工具的执行路径。

Runtime Plan Adaptation

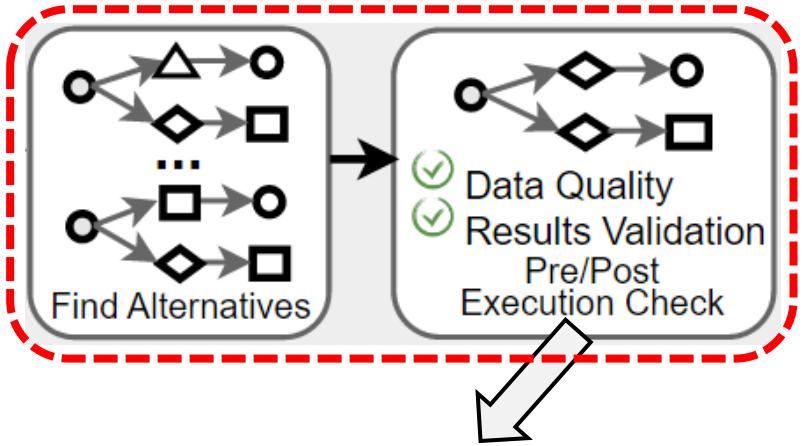
Challenge 2: 环境变化时如何动态切换执行路径，保证执行准确率



- Alternative Path：将功能相似的工具归为一组，对于每个组内的每个工具，构造一条以该工具为终点、只包含其依赖的最小必要工具的执行路径。
- LLM-generated plan -> **Adaptable groups** and **irreplaceable parts**:
  - 选取Alternative Path能完全嵌入到当前规划中的组，如果多个组都能匹配，选取路径最长的组；
  - 将匹配的部分从原规划中剥离出来，标记为一个可替换模块；
  - 对剩余部分重复操作，直至无法匹配，形成多个**可替换模块 + 不可替换**的部分。

Runtime Plan Adaptation

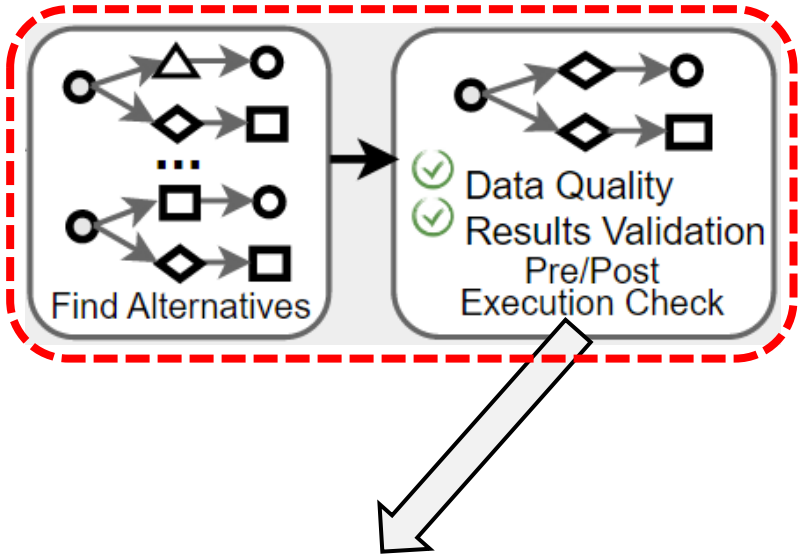
Challenge 2:环境变化时如何动态切换执行路径，保证执行准确率



- Pre-execution Filtering：在执行规划之前，对数据进行依赖检查和质量评估
  - 时间划分：将传感器数据的时域区间按固定长度划分；
  - 依赖检查：对于每个时间段，检查每条Alternative Path的传感器数据来源，若缺失，则跳过；
  - 质量评估：对于能够获得的数据的，评估信噪比等指标，低于阈值的跳过。

Runtime Plan Adaptation

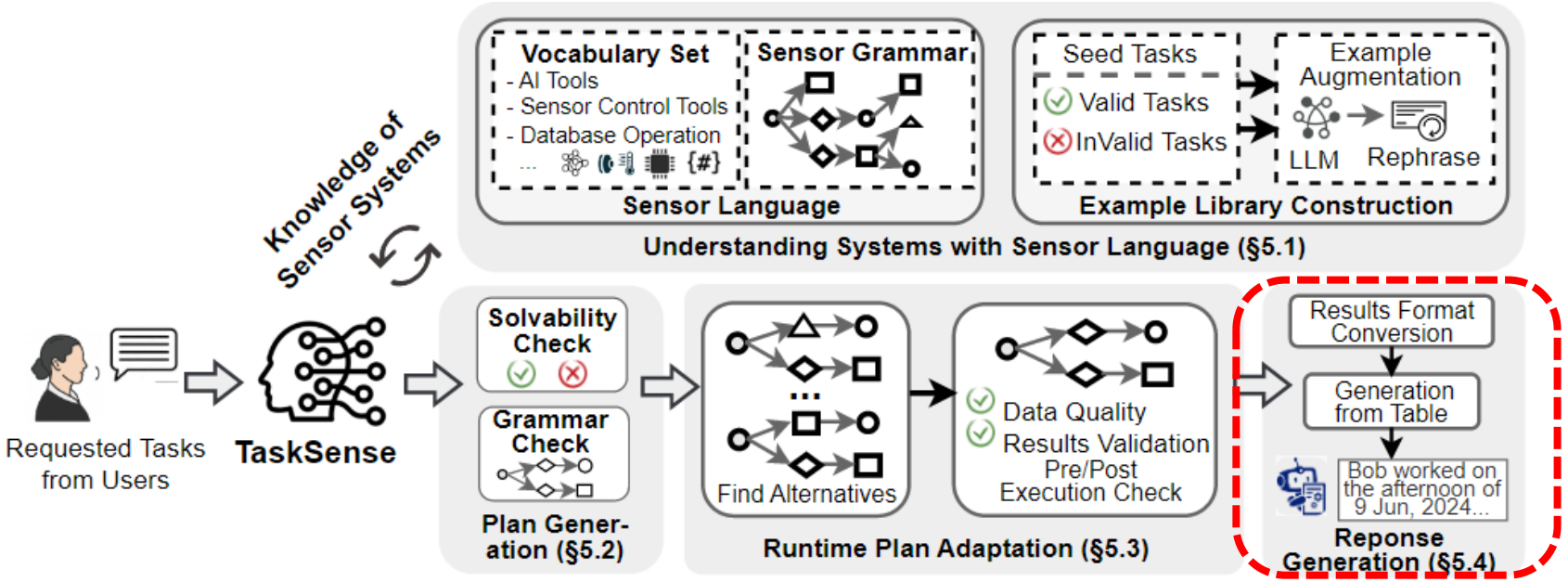
Challenge 2:环境变化时如何动态切换执行路径，保证执行准确率



- Post-execution Selection：在执行规划之后，如物体遮挡、目标移出镜头等无法通过数据质量提前判断的情况
  - 执行结果反馈，若输出为空或明显不合理，则切换到Alternative Path。
  - 缓存机制：存储每个工具在特定trace上的执行结果，在执行前可先查询缓存，若有相同的输入则可直接复用结果。

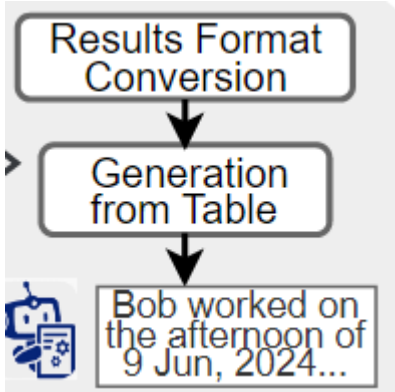
Method：Overview of TaskSense

- **Understanding Systems with Sensor Language**：工具描述和工具间依赖关系图，构建可解示例与不可解示例。
- **Plan Generation**：可解性检查，依赖关系检查。
- **Runtime Plan Adaptation**：运行时，根据执行反馈动态切换可替代路径
- **Response Generation**：将输出结果转换为对象及其标签（如活动、性别、身份）来存储，回复时提取关键信息输入LLM



Response Generation

Challenge 3: 传感器系统产生海量数据结果，LLM对长序列数据推理时，容易产生“遗忘”和“数据关联错误”等幻觉问题



➢ 将工具输出结果格式化为对象属性表，回复时提取关键信息输入：

- **Outcome-wise ID**：为每条工具生成一个随机字符串ID，将工具及其输出ID作为下一个工具的前置ID（pre-IDs），构建一个Result Tree来维护不同工具输出之间的对应关系
- **Results Format Conversion**：利用公共祖先匹配算法识别对象及其对应属性标签，从而构建成对象表。
  - 对象 x1：情绪=“高兴”，动作=“跑步”
  - 对象 x2：情绪=“中性”，动作=“站立”

提纲

1 Background

2 Method

3 Evaluation

4 Conclusion

Evaluation：Setup

工具类型: 专用AI模型，数据库操作工具（如检索传感器数据），传感器控制工具.

Baselines: HuggingGPT（ NeurIPS’ 23）、Sasha（UbiComp’ 24）

Datasets：

- In-lab HAR：RGB + 深度）摄像头系统，6 个工具，DAHLIA（开源）：（RGB + 深度）摄像头系统，6 个工具
- Synthetic Dataset： LLM根据“虚拟对象 + 时间区间 + 工具列表”生成结果并缓存。
- Real-world Dataset：三个房间中部署 8 套异构传感器系统，收集 12 名个体的数据，其中 3 人用于评估

Metrics：

- 规划准确率：生成规划与标准规划是否完全一致
- 规划评分： $S_{\text{score}} = \frac{1}{n}\sum_{i=1}^{n}\frac{2\,PR_i \cdot RC_i}{PR_i + RC_i}$, $RC_i = \frac{|g_i \cap p_i|}{|g_i|}$, $PR_i = \frac{|g_i \cap p_i|}{|p_i|}$, g_i： ground-truth endings , p_i： predicted endings
- 执行准确率: 工具执行输出正确的比例
- 响应准确率：最终形成自然语言回答的准确率

Evaluation： Overall performance

Using GPT4 as base LLM

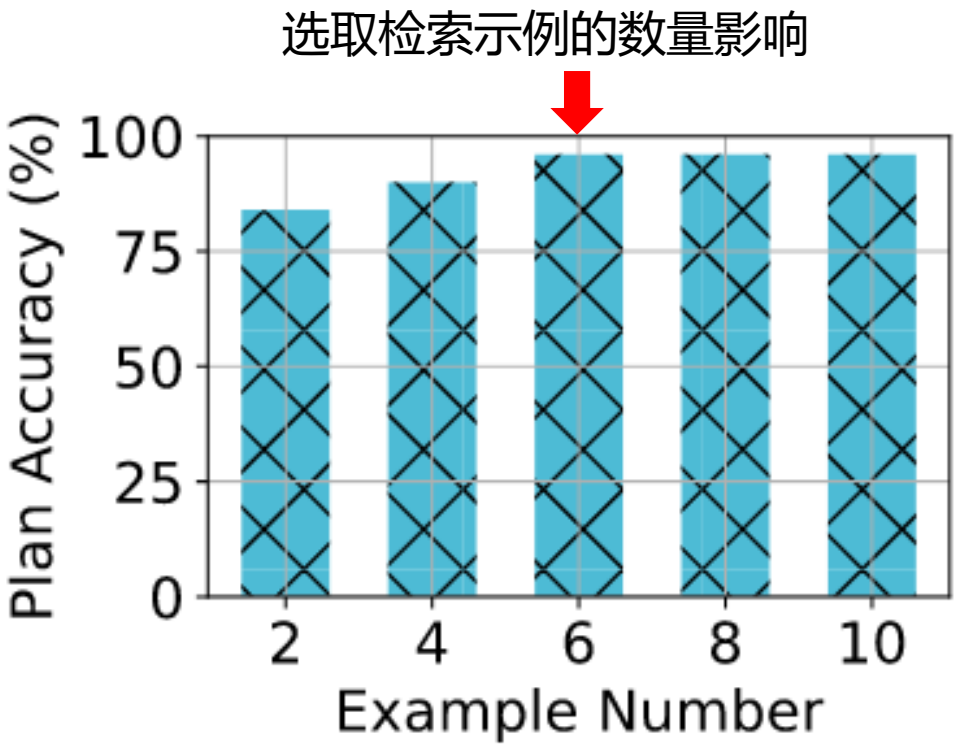
Dataset	Methods	Planning Accuracy	Execution Accuracy	Response Accuracy
In-lab	HuggingGPT	0.80	0.55	0.47
	Sasha	0.86	0.56	0.50
	Ours	**0.97**	**0.75**	**0.73**
DAHLIA	HuggingGPT	0.86	0.43	0.50
	Sasha	0.76	0.42	0.40
	Ours	**0.93**	**0.60**	**0.70**
Synthetic	HuggingGPT	0.66	0.59	0.50
	Sasha	0.32	0.67	0.48
	Ours	**0.96**	**0.72**	**0.74**
Real-world	HuggingGPT	0.55	0.35	0.55
	Sasha	0.66	0.37	0.62
	Ours	**0.82**	**0.64**	**0.75**

In-lab dataset using different LLM

LLM	Methods	Planning Accuracy	Execution Accuracy	Response Accuracy
GPT-4o	HuggingGPT	0.83	0.51	0.60
	Sasha	0.83	0.47	0.63
	Ours	**0.97**	**0.72**	**0.70**
Claude-3-Opus	HuggingGPT	0.87	0.55	0.57
	Sasha	0.60	0.23	0.53
	Ours	**0.97**	**0.65**	**0.77**
Claude-3.5-Sonnet	HuggingGPT	0.73	0.54	0.50
	Sasha	0.83	0.48	0.73
	Ours	**0.93**	**0.67**	**0.70**
Llama 3 70B Instruct (Open-source)	HuggingGPT	0.47	0.27	0.37
	Sasha	0.63	0.48	0.37
	Ours	**0.77**	**0.54**	**0.57**
Mistral Large (Open-source)	HuggingGPT	0.73	0.58	0.47
	Sasha	0.40	0.18	0.30
	Ours	**0.97**	**0.74**	**0.70**

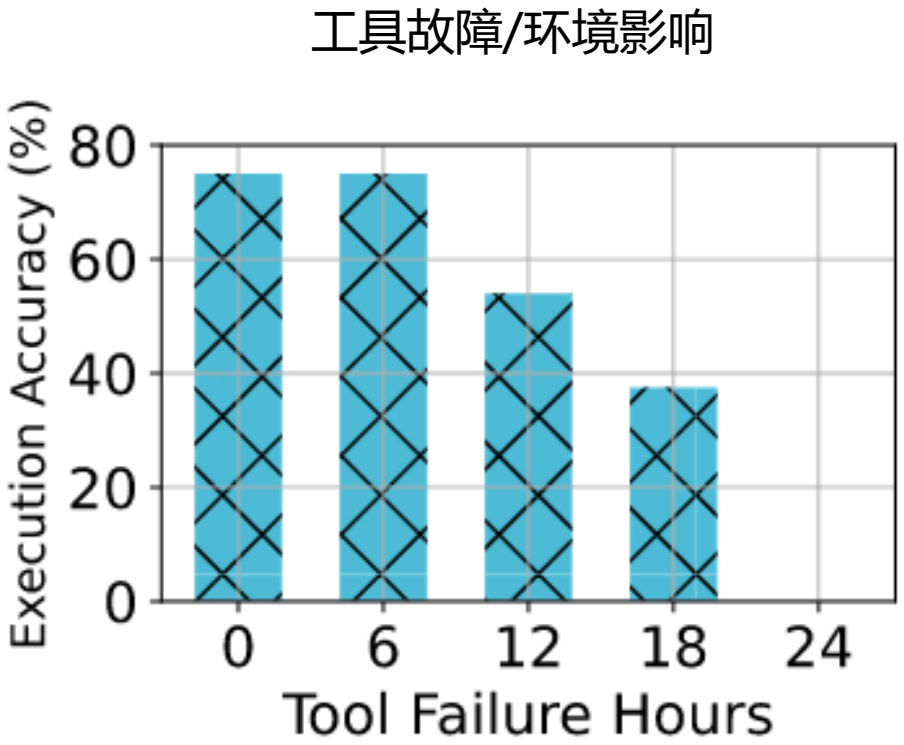
Evaluation：

- In-lab dataset



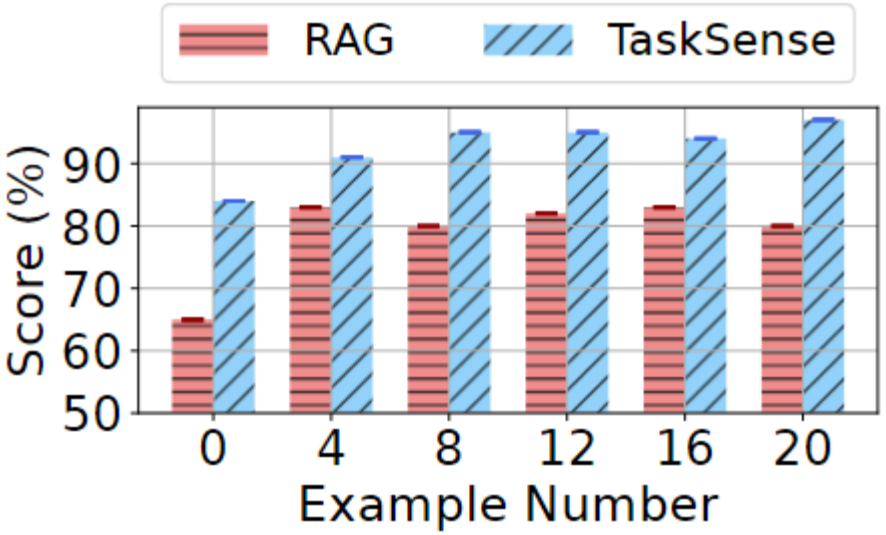
Evaluation：

- Synthetic dataset
  - 模拟一天中不同长度的“工具失效时间”，如RGB 摄像头夜间受光线影响失效，深度摄像头因遮挡而输出异常等



Evaluation：

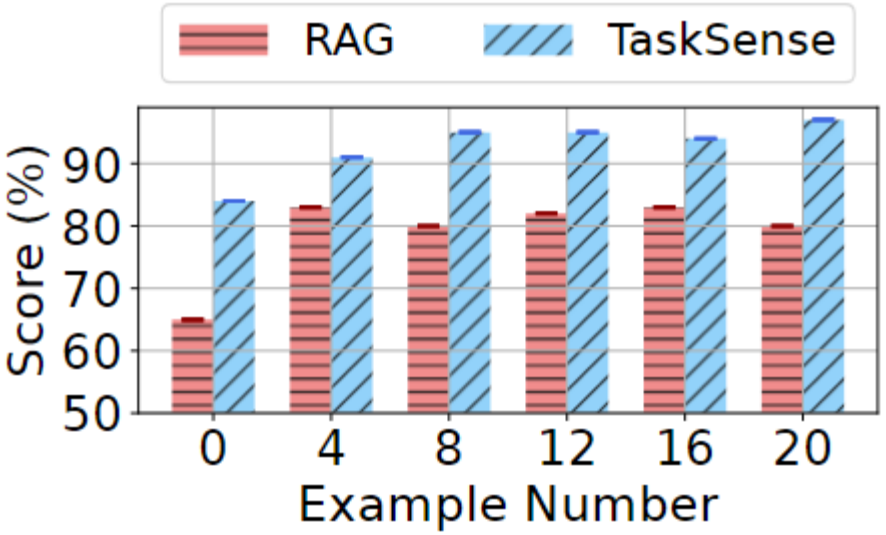
- In-lab dataset
  - Planning example RAG：仅保留示例检索，删去可解性和依赖检查



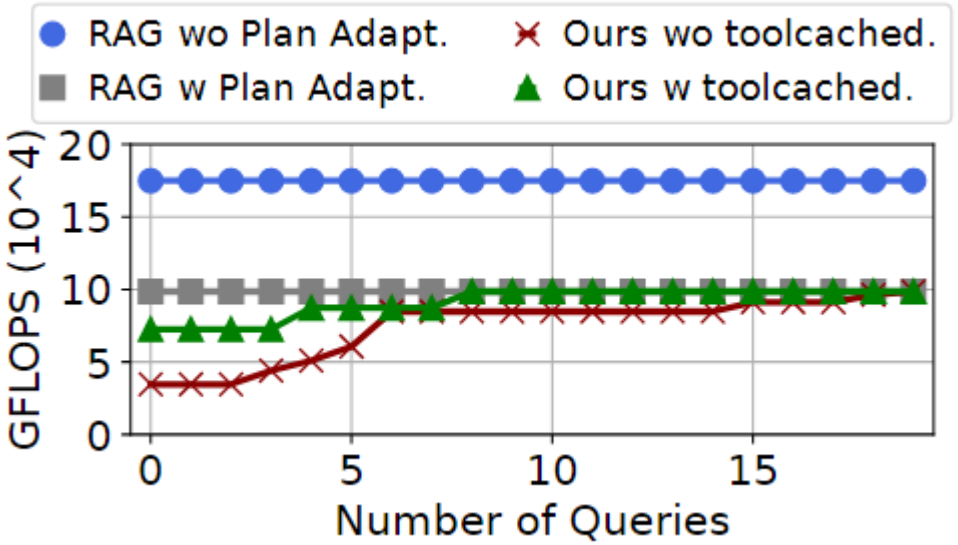
Planning Score

Evaluation：

- In-lab dataset
  - Planning example RAG：仅保留示例检索，删去可解性和依赖检查
  - Tool output RAG：对工具持续执行，并将每次执行结果缓存（全部预执行，查询直接检索结果）



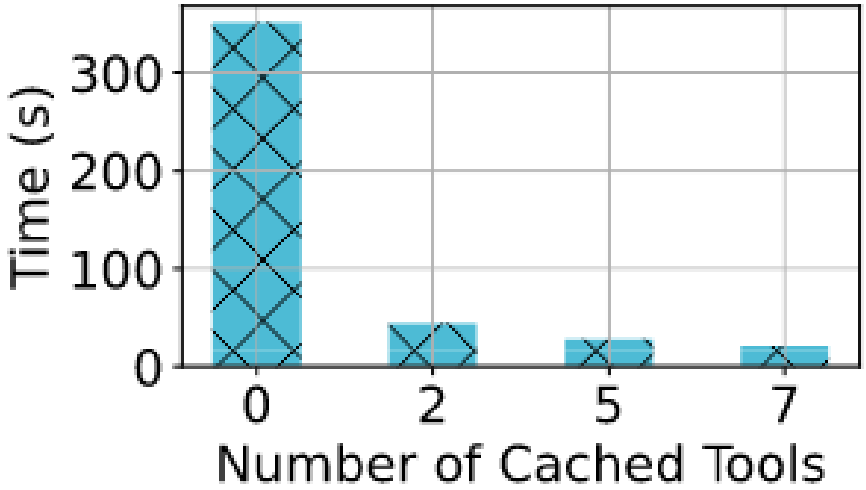
Planning Score

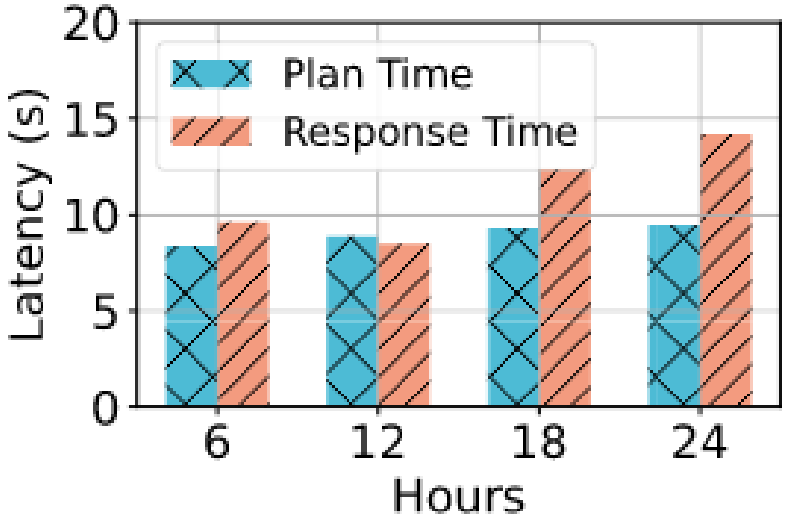


系统计算负载

Evaluation:

- In-lab dataset





System Overhead

提纲

1 Background

2 Method

3 Evaluation

4 Conclusion

Conclusion

- 解决了规划生成中的可解性错误和数据依赖错误
  - Sensor Language + Sensor grammar
  - Example Library
  - Solvability Checking + Grammar Checking

- 利用传感器系统之间的互补性，运行时动态切换可替代的执行路径消除环境影响，并记录工具的输入和输出缓存防止重复执行带来的开销
  - Dynamic Plan Adaptation

- 由于传感器系统易产生大量数据的特性，将输出结果组织成对象及其属性的表格，提取需要的信息输入LLM，减少长序列数据带来的幻觉问题
  - Response Generation

Thinking

- 能否提高
  - 示例库训练到模型里，通过规则强化学习加强reasoning来判断可解性和数据依赖
  - 文章考虑了传感器之间的互补性，但是未考虑传感器系统本身输出的信息，比如光照等是可以通过光照传感器提前得知的状态，作为提示词输入
  - 成本考虑，当光照正常或者恢复到正常时，活动识别应优先使用普通的摄像头

- 泛化：文章的规划思想是从通用的工具学习泛化到传感器系统上

- 用到我们的idea中：sensor grammar和Alternative Path适合用来构建数据集

東南大學
SOUTHEAST UNIVERSITY

恳请各位老师与同学批评指正！