**2025.1月周报**

**Related Paper研读和idea**

  主要是关于LLM、RL应用于游戏测试、场景探索的工作。

* 1. 瀚阳师兄的近期工作（还没发表，技术细节就不提了）

  发现没有LLM的基础对Software Reliability这块很痛苦，正在开始恶**补LLM基础**

* 1. （FSE '21）*GLIB: Towards Automated Test Oracle for Graphically-Rich Applications*

文章是检测Unity游戏中的UI Glitches异常（属于GUI Testing）。先明确了8种画面异常以及引起它们的4种code级别的原因，通过手动截异常图+代码注入方法增强数据集，训练CNN模型去分辨UI Glitches屏幕截图，通过梯度图得到显著图的方式帮助程序员定位画面异常。这篇工作侧重画面异常bug，并且结合了cv的方法。

  文章是先做了bug characteristic的实证研究，分析代码级别原因，然后手动注入bug；**可以借鉴先收集大量bug然后做bug characteristic的实证分析，再做taxonomy**

* 1. *★*（ASE '23）*VRGuide: Efficient Testing of Virtual Reality Scenes via Dynamic Cut Coverage*二刷 。

  重点**回顾了下作者的Evalutaion Metrics**，用的是**可交互物体覆盖率** + **Method Coverage**。这两个指标可以用到现在的VRAgent工作里；此外，还想加上比如 **物理交互覆盖率**、路径探索覆盖率等指标，这里有些灵感，先记录一下（物理交互矩阵 + 主成分分析降维优化）

* 1. （ASE '19）*Wuji: Automatic Online Combat Game Testing Using Evolutionary Deep Reinforcement Learning*

这篇工作的对象是多人在线对战游戏，方法是演化强化学习。首先对bug characteristic进行实证分析分成四类；在测试时忽略了GUI，重点关注比如游戏技能释放和移动作为状态；然后通过手动注入个位数的bug作为ground truth，通过代码覆盖率、状态覆盖率评估，最后采取2个网易游戏作为case study

* + 1. **代码注入bug作为ground truth**还是比较普遍的
    2. 代码覆盖率在游戏测试中作为指标的不合理性
    3. **忽略GUI**这个方法，可以借鉴；**简化RL问题**
  1. （A-Test '20）*Navigation and Exploration in 3D-Game Automated Play Testing*

探讨3D游戏中的寻路算法问题，包括动态避障和精细控制（比如有选择性的走入危险区域），也提到了决策序列问题动态避障问题在Unity NavmeshAgent中可以启发式的解决，精细控制我认为需要加上强化学习，决策序列这里只提到了类似于简单的FSM有限状态机。

  这篇文章比较早，但是在寻路这块也提到了现在的问题，可以作reference

*★* (A-Test '21) *Using an Agent-Based Approach for Robust Automated Testing of Computer Games*

这篇工作是上一篇的延续，实现了一个名为 iv4XR的框架（java-based）。该框架通过对游戏目标序列化，对每个策略给出action的组合实现，是对action和task的基于序列化的形式建模。

  本文的对action和task形式化建模（需要手动建模）；我的idea是如果能通**过LLM对代码的学习、游戏场景的学习，能够自动生成action和task序列**，是可以更上一层楼的，还没调研到类似这种。

  下面是文章给出的形式化建模在Java游戏中的实现，实际上比较类似有限状态机FSM.....

  在数据集中选取代表性项目，尝试分析一下FSM，并进行设计和建模有限的空间，针对性的进行explore以及detect bugs

  $$\begin{aligned} & \text{var testingTask = SEQ(} \\ & \quad \text{GoalLib.entityInteracted("button1"),} \\ & \quad \text{GoalLib.entityStateRefreshed("door1"),} \\ & \quad \text{GoalLib.entityInvariantChecked(testAgent,} \\ & \quad \quad \text{"door1",} \\ & \quad \quad \text{"door1 should be open",} \\ & \quad \quad \text{(WorldEntity e) } \rightarrow \text{ e.getBooleanProperty("isOpen"))} \\ & \text{);} \end{aligned}$$

**2024年**