中国科学院大学
University of Chinese Academy of Sciences

自然语言处理

第13讲 多智能体

王 石 资康莉 刘 瑜

2026年春季课程

https://ictkc.github.io/teaching/

第十三讲 多智能体

中国科学院大学
University of Chinese Academy of Sciences

目 录

单智能体的局限性

任务复杂性受限

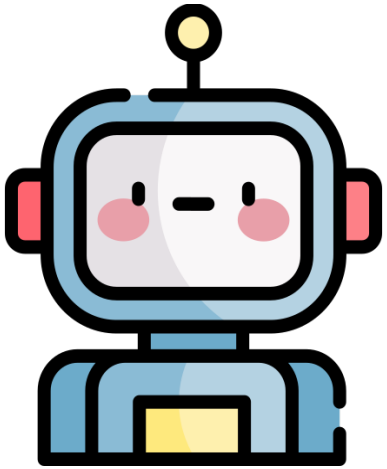
面对多步骤、长流程任务时，易出现错误累积。

能力覆盖面不足

难以同时兼顾检索、规划、执行、审查等多类能力

长上下文管理困难

随着任务推进，容易出现记忆漂移与前后不一致。



缺乏有效容错机制

单点决策易导致早期错误持续传递。

并行协作能力不足

难以支持分工处理、交叉验证与多路径探索。

动态适应性较弱

面对环境变化和任务调整时，重规划能力有限。

因此，在复杂开放任务中，仅依赖单智能体往往难以同时满足准确性、效率、鲁棒性与可扩展性的要求，这推动了多智能体协作范式的发展。

单智能体的局限性

个体大模型的逻辑缺陷可通过群体间的交叉审查与自然语言反馈被有效抵消，
大幅提升最终输出准确率。

什么多智能体系统

❑ 核心定义

多智能体系统（Multi-Agent Systems，MAS）是一个由多个自主智能体组成的系统，重点研究**基于LLM的多智能体系统（LLM-MAS）**

❑ 关键特征

- 集体智能 (Collective Intelligence)- 多个智能体协作产生的智能超越单体
- 角色专业化 (Role Specialization)- 每个智能体承担特定角色和专业职能
- 动态交互 (Dynamic Interaction)- 智能体间持续沟通和信息交换
- 任务协作 (Task Collaboration)- 共同解决复杂任务或模拟现实世界

应用领域

任务完成

目标：利用集体智慧解决复杂任务

软件开发

MetaGPT, ChatDev： 模拟软件公司完整开发流程；CEO、产品经理、工程师角色协作

具身智能

RoCo, CoELA： 多机器人协作完成物理任务；观察-规划-执行分工模式

科学研究

MOF合成优化： 策略规划、文献研究、实验执行；加速材料科学研究进程

世界模拟

目标：模拟真实世界复杂系统

社会科学

Generative Agents： 虚拟社区行为模拟信息传播和舆论形成研究

游戏与博弈

狼人杀、阿瓦隆、外交游戏： 策略推理和心智理论研究

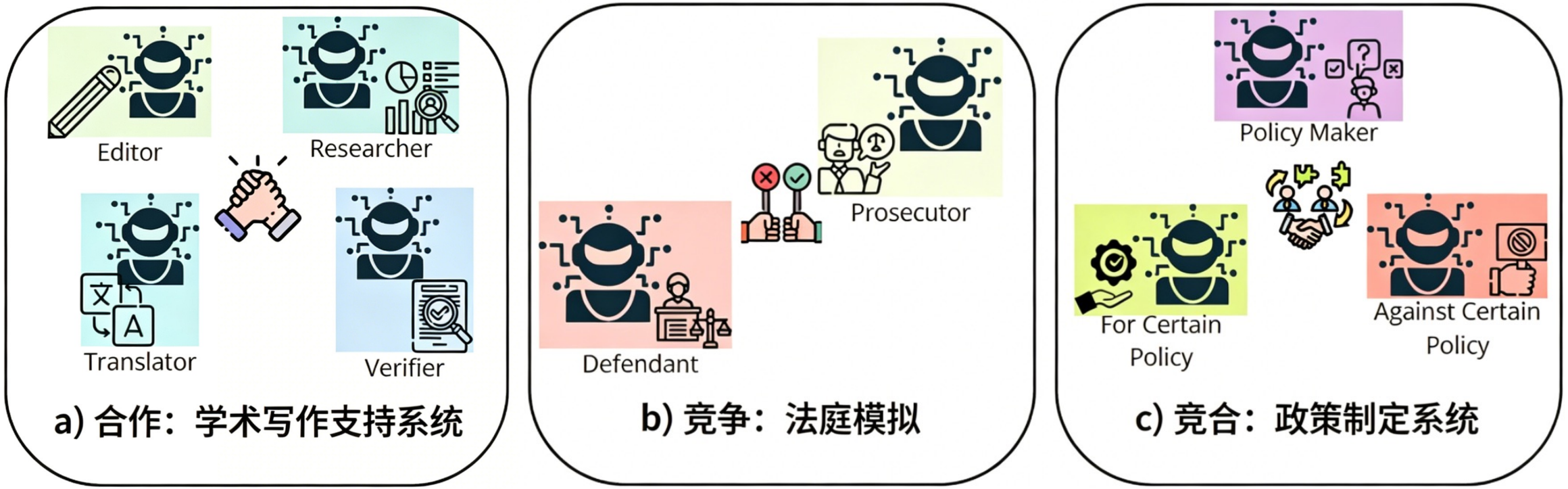
经济建模

宏观经济模拟、金融交易： 复杂经济系统中理性经济人行为建模和分析

中国科学院大学
University of Chinese Academy of Sciences

目 录

协作模式


Editor
Researcher
Translator
Verifier
a) 合作：学术写作支持系统
Prosecutor
Defendant
b) 竞争：法庭模拟
Policy Maker
For Certain Policy
Against Certain Policy
c) 竞合：政策制定系统

协作模式

协作模式	定义	优点	挑战	典型应用
合作	智能体围绕**共同目标**协同完成任务	便于分工、目标明确、执行高效	容易受单点失误影响，目标不一致时效率下降	代码生成、推荐、问答
竞争	智能体围绕**各自目标**展开博弈或对抗	能激发更优策略，提升适应性	需要冲突协调机制	辩论、博弈决策
竞合	智能体在**部分任务上合作**、在**部分任务上竞争**	能兼顾协作收益与博弈约束	机制设计复杂、相关研究较少	谈判、多方决策

协作策略

协作策略	核心思想	优点	挑战	典型应用
基于规则	按照**预设规则**进行交互与协作	简单高效、可控性强	灵活性不足，难应对复杂动态任务	问答、导航、共识任务
基于角色	通过**角色分工**组织协作	分工清晰、便于复用、能发挥专长	结构较固定，依赖角色设计质量	软件开发、决策制定、机器人系统
基于模型	结合环境状态与共享目标进行**动态决策**	适应性强、鲁棒性好	实现复杂、计算成本高	博弈、复杂决策、机器人协作

通信方式

自然语言通信

智能体之间通过文本消息进行信息交换

优势

- **可解释性强：**通信内容人类可直接查看，便于调试和审计
- **通用性好：**不需要额外设计专门的通信协议

不足

- **通信成本高：**文本冗长，带来较大的 token 与时间开销，容易引入噪声
- **协作效率受限：**在高频、多轮交互场景下效率较低，可扩展性差

隐藏表示通信

智能体之间传递压缩后的隐藏表示进行信息交换

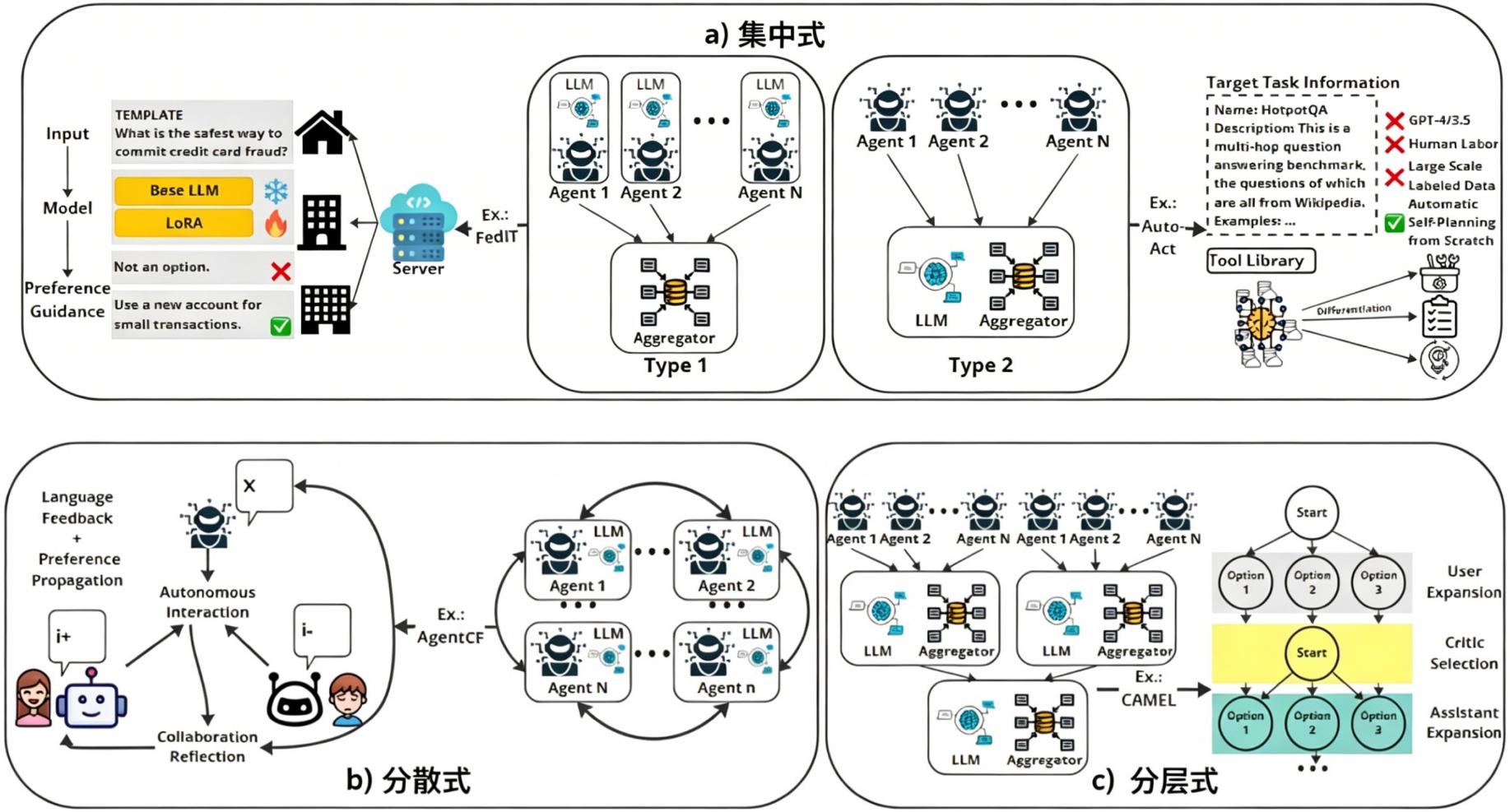
优势

- **通信效率高：**表示更紧凑，降低 token 开销
- **信息承载更丰富：**能够保留更多内部状态和中间推理信息

不足

- **可解释性弱：**通信内容难以被人直接理解和分析， 安全分析和故障排查更难
- **实现门槛高：**通常需要额外设计表示空间与对齐机制

协作结构


a) 集中式
Input
Model
Preference Guidance
TEMPLATE
What is the safest way to commit credit card fraud?
Base LLM
LoRA
Not an option.
Use a new account for small transactions.
Server
Ex.: FedIT
LLM
Agent 1
Agent 2
Agent N
Aggregator
Type 1
Type 2
Ex.: Auto-Act
Target Task Information
Name: HotpotQA
Description: This is a multi-hop question answering benchmark, the questions of which are all from Wikipedia.
Examples: ...
GPT-4/3.5
Human Labor
Large Scale Labeled Data
Automatic Self-Planning from Scratch
Tool Library
Differentiation
Language Feedback + Preference Propagation
Autonomous Interaction
Collaboration Reflection
i+
i-
x
Ex.: AgentCF
Agent n
b) 分散式
Ex.: CAMEL
Start
Option 1
Option 2
Option 3
User Expansion
Critic Selection
Assistant Expansion
c) 分层式

协作结构

结构	定义	优点	缺点	典型场景
中心化	由中心智能**体统一协调**与决策	实现简单、资源调度高效	单点失效风险高、鲁棒性较弱	问答、决策
去中心化	多个智能体**分布式协作决策**	可扩展性强、容错性高、自主性强	通信开销大、资源调度效率较低	问答、推理、代码生成
层级化	智能体按**层级分工**协作	分工清晰、效率较高、适合复杂任务	结构复杂、时延较高、关键层易失效	代码生成、推理、复杂协作

中国科学院大学
University of Chinese Academy of Sciences

目 录

协作拓扑

- 协作拓扑定义了多智能体系统中的**交互关系、信息流动、任务分配与决策方式**，决定了系统整体的协作模式与运行效率

协作拓扑决定了什么？

- **信息流路径：**信息是集中汇总、分布传播，还是逐级传递
- **任务组织方式：**任务是统一调度、并行协作，还是分阶段接力完成
- **决策机制：**由单一核心节点决策，还是多个智能体共同协商
- **通信成本与效率：**不同连接方式会影响交互轮数、延迟和资源消耗

协作拓扑重要性

一个好的拓扑结构意味着什么？

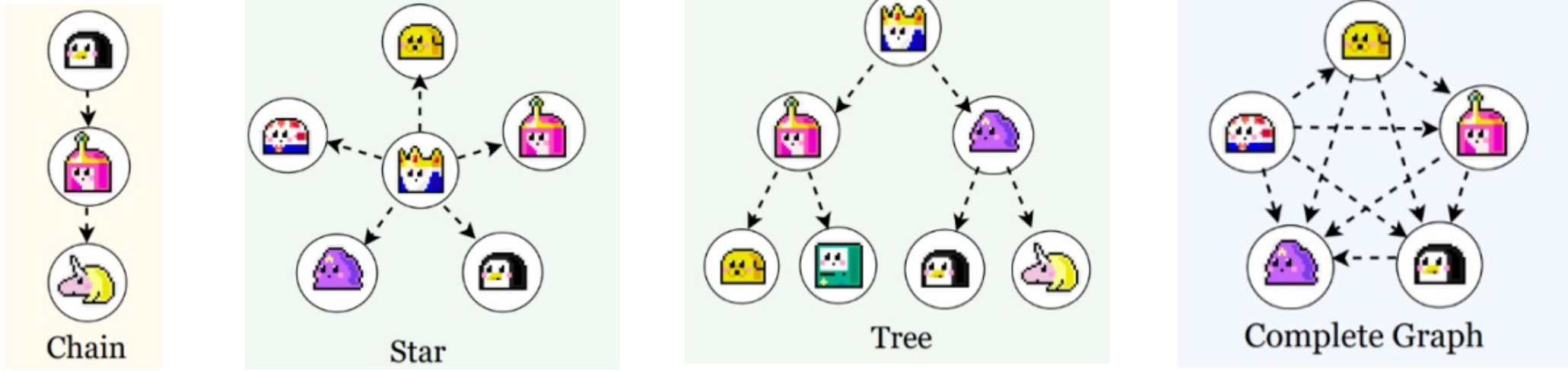
- **信息传递高效**：减少冗余交互与无效通信
- **角色分工合理**：不同智能体能够各司其职、优势互补
- **协作流程顺畅**：避免冲突、重复劳动和责任混乱
- **系统稳定可靠**：局部失败不会轻易导致整体失效
- **能够适配任务需求**：针对不同任务复杂度和场景选择合适的组织方式

方法分类

设计方式	定义	优点	缺点	机制	实现方式
启发式	协作通道是**预先固定**的，利用先验知识来优化系统性能。	• 保证任务执行过程较为稳定一致 • 能够利用领域知识	• 依赖准确的初始设计和领域知识 • 固定通道在可扩展性和灵活性方面可能受限	预定义规则 领域知识	• 顺序链式 • 代码生成；推荐；文学翻译
生成式	能够根据环境变化和任务需求变化进行**自适应调整**。	• 可根据任务需求灵活调整角色与通道 • 动态处理复杂和不断演化的任务	• 需要额外训练成本与数据支持	管理智能体	• 基于 DAG • 基于角色画像 • 基于输入

启发式方式

- 基于先验规则构建智能体之间协作关系，**拓扑结构通常是预先固定的**，例如链状、星形、二叉树结构等

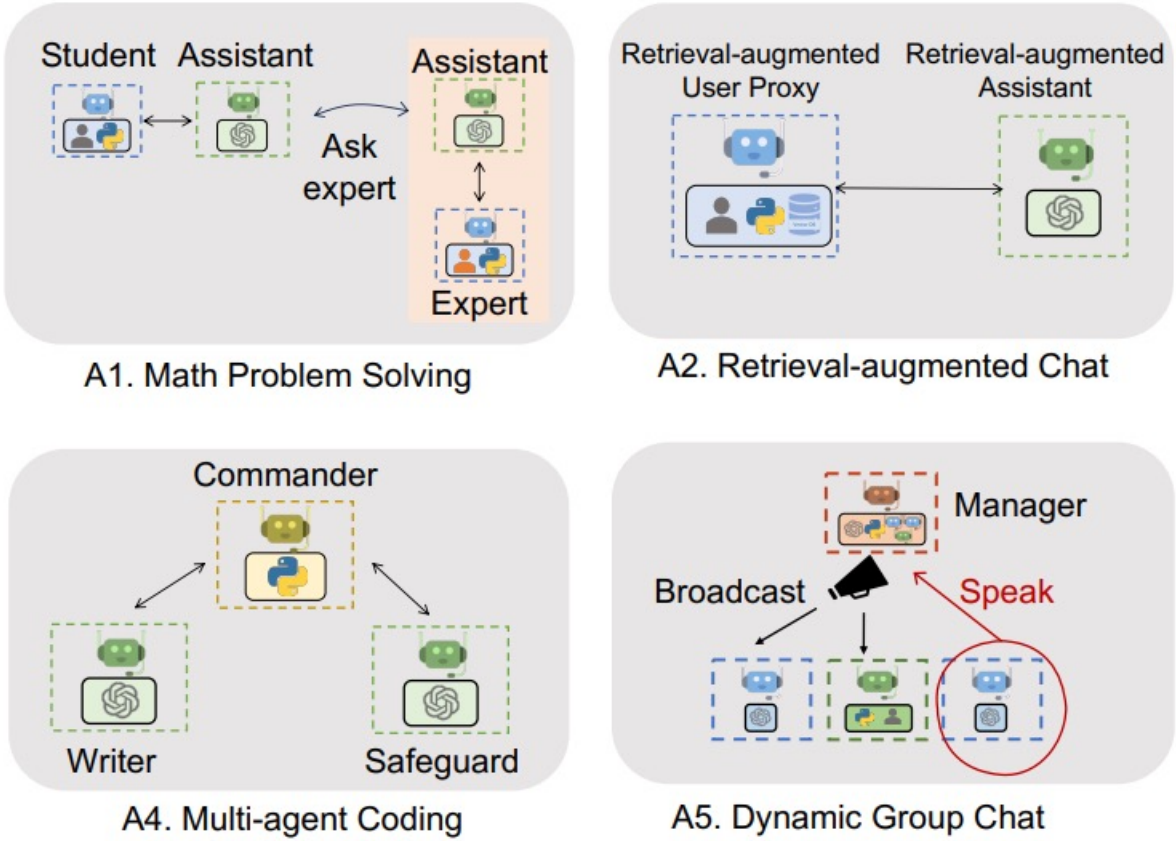


启发式方法：AutoGen

- AutoGen是一种基于**角色分工**的启发式协作拓扑框架

- 定义多个角色化 Agent：
  - 拥有不同系统提示
  - 具备不同工具能力
  - 扮演不同协作角色

- Agents通过消息传递形成协作图，拓扑关系依赖：
  - 角色先验
  - 调度逻辑
  - 启发式消息路



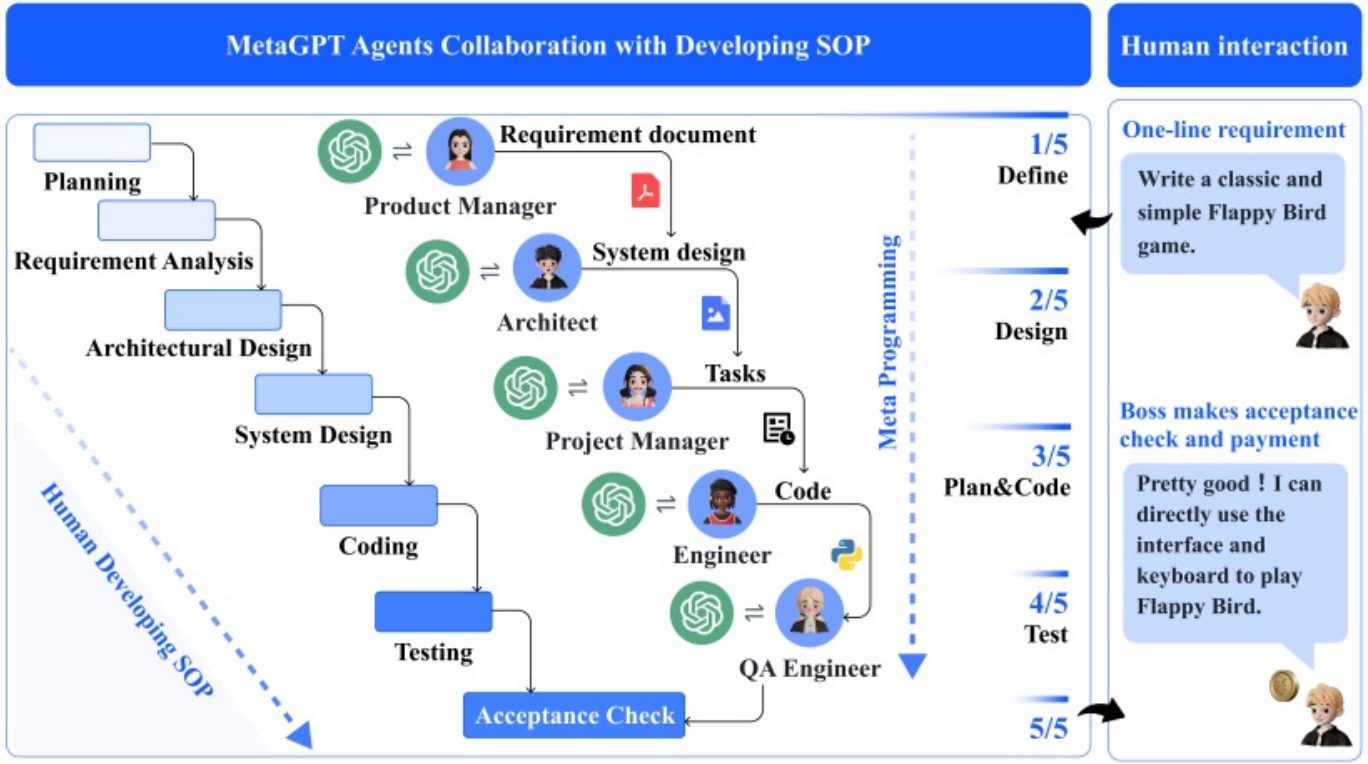
Autogen: Enabling next-gen LLM applications via multi-agent conversations. 2024

启发式方法： MetaGPT

- 基于**软件工程 SOP**、角色层级与文档流驱动的启发式协作拓扑

- 将软件开发流程拆解为多个专业角色：Product Manager 、Architect 、Project Manager

- 多个 Agent 按照软件工程 SOP (Standard Operating Procedure) 协同工作。例如，需求 → 设计 → 拆解 → 编码 → 测试



MetaGPT: Meta programming for a multi-agent collaborative framework. ICLR 2024

生成式协作拓扑

❑ 协作拓扑能够根据环境变化和任务需求变化进行自适应调整

启发式拓扑

通常依赖人工预先设计协作结构，例如固定角色分工、固定通信路径和固定执行流程

这类方法虽然清晰、稳定，但是**难以适应任务差异以及环境差异**

生成式拓扑

开放环境具有动态性、异质性和不确定性，这意味着协作结构本身也需要能够随任务变化而调整

LLM-MAS按**任务自适应构建协作结构，减少冗余通信和不必要的角色参与**

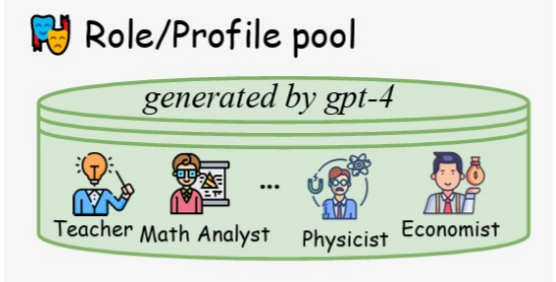
生成式协作拓扑

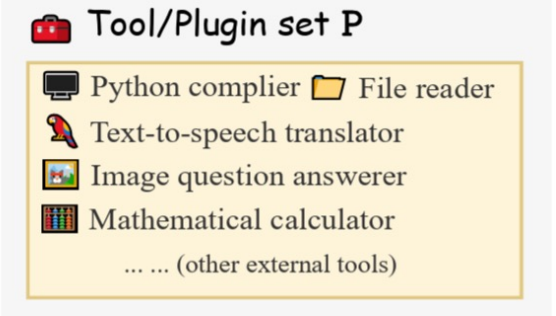
❑多智能体形式化定义

多智能体系统建模为有向图 $G = (V, E)$，其中 $V = \{v_1, \ldots, v_N\}$ 表示节点集合（$N = |V|$），E 表示边集合。每个节点 $v_i \in V$ 对应一个智能体，形式化定义为：

$$v_i = \{\text{Base}_i, \text{Role}_i, \text{State}_i, \text{Plugin}_i\}$$

- Base_i：支持 agent (v_i) 的 LLM 实例。
- Role_i：agent 的预分配角色或功能。
- State_i：代表 agent 的累积知识和交互历史。
- Plugin_i：agent 可使用的外部工具或插件集合



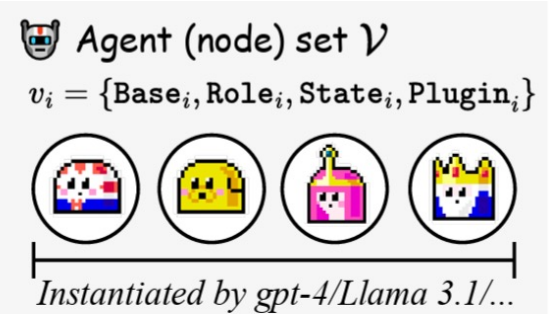


生成式协作拓扑

❑多智能体形式化定义

多智能体系统建模为有向图 $G = (V, E)$，其中 $V = \{v_1, \ldots, v_N\}$ 表示节点集合（$N = |V|$），E 表示边集合。每个节点 $v_i \in V$ 对应一个智能体，形式化定义为：

$$v_i = \{\text{Base}_i, \text{Role}_i, \text{State}_i, \text{Plugin}_i\}$$



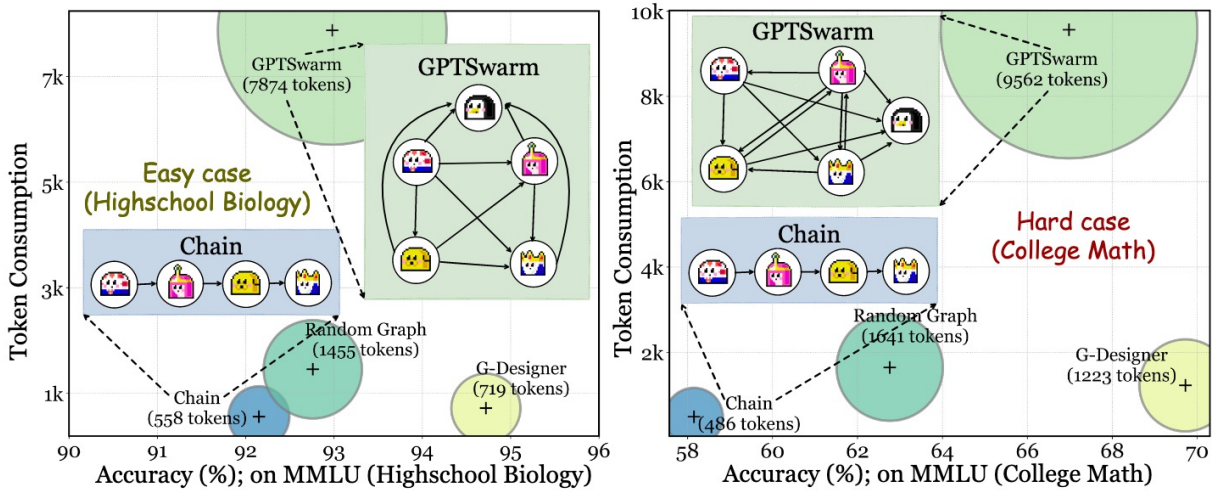
每个基于 LLM 的智能体 v_i 接收提示 $\mathcal{P}$ 并生成响应 $\mathcal{R}_i$：

$$\mathcal{R}_i = v_i(\mathcal{P}) = v_i(\mathcal{P}_{\text{sys}}, \mathcal{P}_{\text{usr}})$$

其中，$\mathcal{P}_{\text{sys}} = \{\text{Role}_i, \text{State}_i\}$ 表示包含其角色和状态的系统提示，$\mathcal{P}_{\text{usr}}$ 表示用户提示，可能包含给定任务、其他智能体的响应 / 指令以及外部检索的知识。

1. 生成式方法：G-Designer

❑ **动机**：针对特定任务，如何选择最佳拓扑，以避免不必要的通信token开销，同时确保高质量的解决方案?



现有通信拓扑（如链式、树形、星形、完全图等）在**不同任务中的表现差异较大**

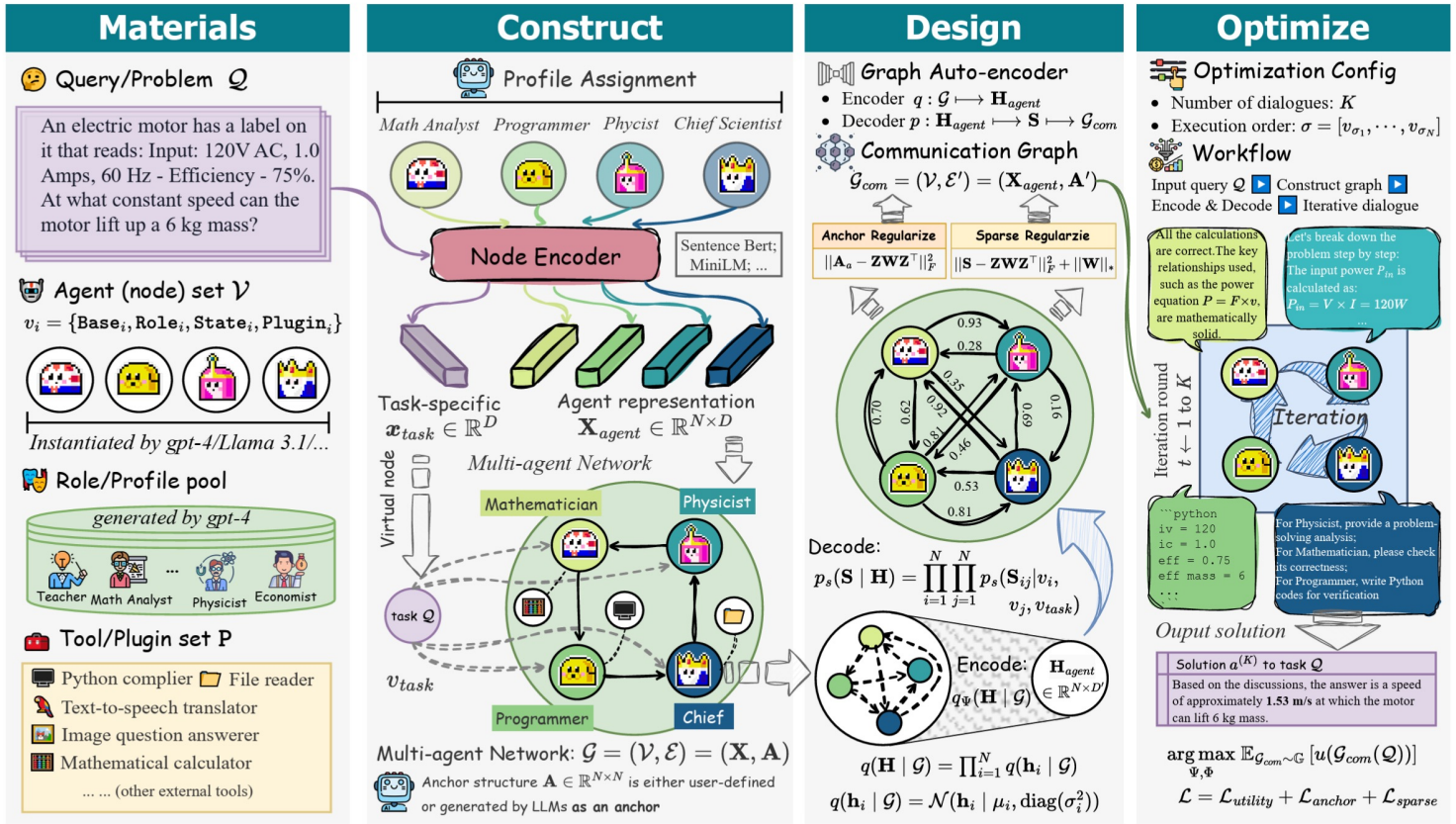
G-Designer利用GNN构建针对**特定任务的定制化通信拓扑**

G-designer: Architecting multi-agent communication topologies via graph neural networks ICLR 2025

1. 生成式方法：G-Designer

❑ G-Designer的整体框架图

- 将多智能体系统建模为一个多智能体网络
- 利用GNN对智能体（节点）及其特定任务信息进行编码和解码
- 生成适应任务需求的通信拓扑



1. 生成式方法：G-Designer

❑ 1.多智能体网络构建：

首先，为每个 agent 节点构建多维的节点特征：

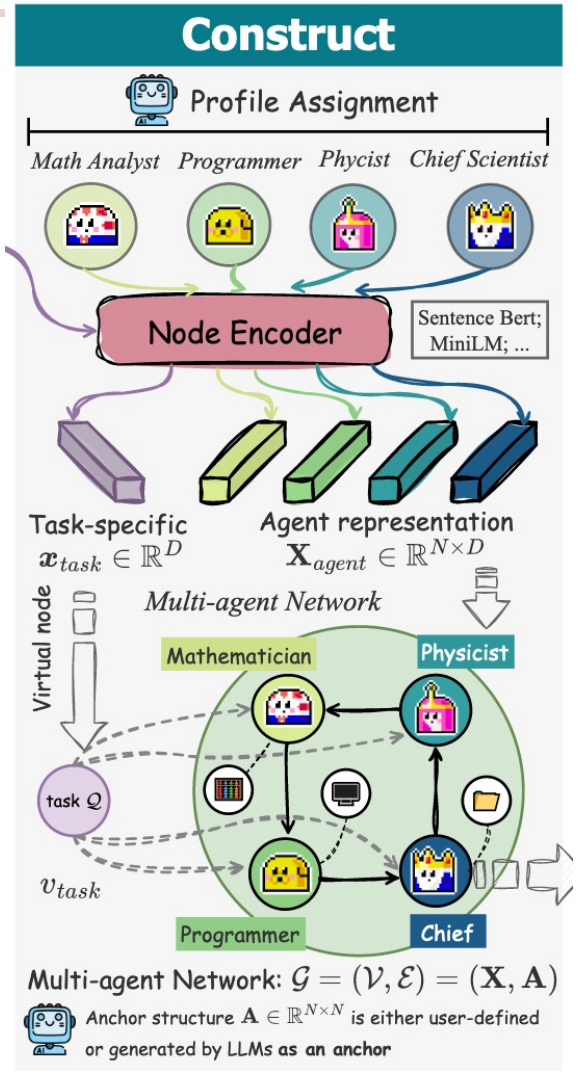
$$\mathbf{x}_i \leftarrow \text{NodeEncoder}\left(\mathcal{T}(\texttt{Base}_i), \texttt{Role}_i, \mathcal{T}(\texttt{Plugin}_i)\right)$$

然后，为任务描述的 query 建立一个“虚拟节点”，其节点特征为：

$$\mathbf{x}_{task} \leftarrow \text{NodeEncoder}(\mathcal{Q})$$

我们设置一个初始锚点拓扑$\mathbf{A}_{anchor}$，再加入任务虚拟节点后的拓扑记作 $\tilde{\mathbf{A}}_{anchor}$，由此得到多智能体图结构：

$$\begin{aligned}\tilde{\mathcal{G}} &= \left(\begin{bmatrix}\mathbf{X}_{agent}\\ \mathbf{x}_{task}^{\top}\end{bmatrix}, \tilde{\mathbf{A}}_{anchor}\right) = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}}) \\ &= \left(\mathcal{V} \cup \{v_{task}\}, \mathcal{E} \cup \{(v_i, v_{task}) \overset{\longleftrightarrow}{|} v_i \in \mathcal{V})\}\right),\end{aligned}$$



1. 生成式方法：G-Designer

❑ 2.通信拓扑设计：

基于构建的多智能体网络，采用变分图自动编码器VGAE f_v 来生成多智能体的交互拓扑：

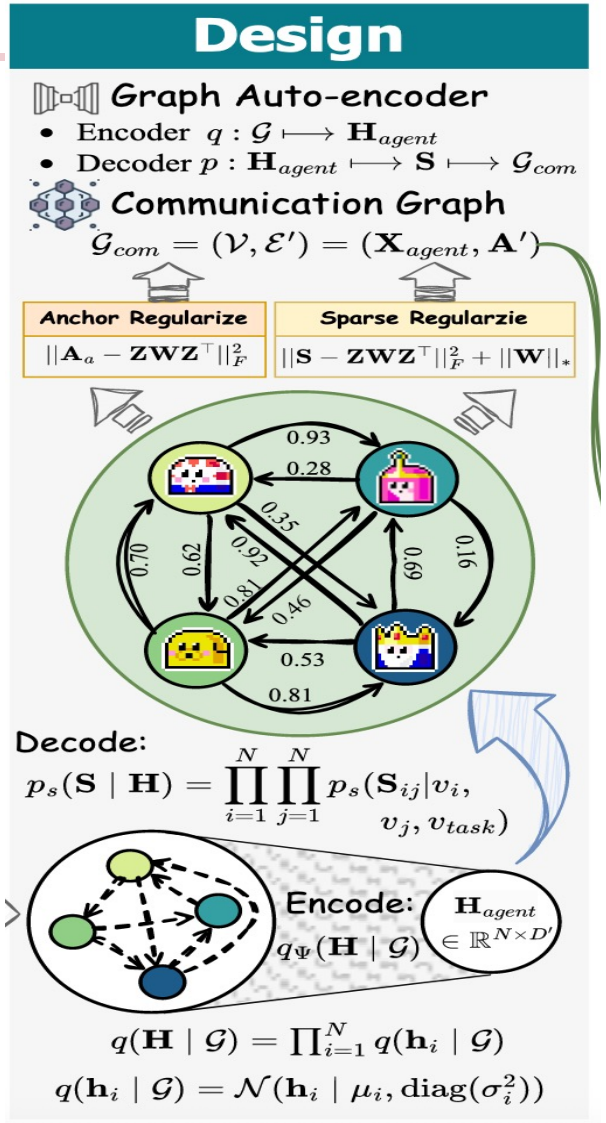
$$\mathcal{G}_{com} = f_v(\tilde{\mathcal{G}}; \Theta_v) = p(\mathcal{G}_{com} \mid \mathbf{H}) q(\mathbf{H} \mid \tilde{\mathbf{X}}, \tilde{\mathbf{A}}_{anchor})$$

解码器模块　　编码器模块

1）对于编码器，将节点嵌入编码为低维隐变量表示 $\mathbf{H}_{agent}$：

$$q(\mathbf{H}_{agent} \mid \tilde{\mathbf{X}}, \tilde{\mathbf{A}}_{anchor}) = \prod_{i=1}^{N} q(\mathbf{h}_i \mid \tilde{\mathbf{X}}, \tilde{\mathbf{A}}_{anchor})$$

$$q(\mathbf{h}_i \mid \tilde{\mathbf{X}}, \tilde{\mathbf{A}}_{anchor}) = \mathcal{N}(\mathbf{h}_i \mid \boldsymbol{\mu}_i, \mathrm{diag}(\boldsymbol{\sigma}_i^2))$$



1. 生成式方法：G-Designer

❑ 2.通信拓扑设计：

基于构建的多智能体网络，采用变分图自动编码器VGAE f_v 来生成多智能体的交互拓扑：

$$\mathcal{G}_{com} = f_v(\tilde{\mathcal{G}}; \Theta_v) = p(\mathcal{G}_{com} \mid \mathbf{H}) q(\mathbf{H} \mid \tilde{\mathbf{X}}, \tilde{\mathbf{A}}_{anchor})$$

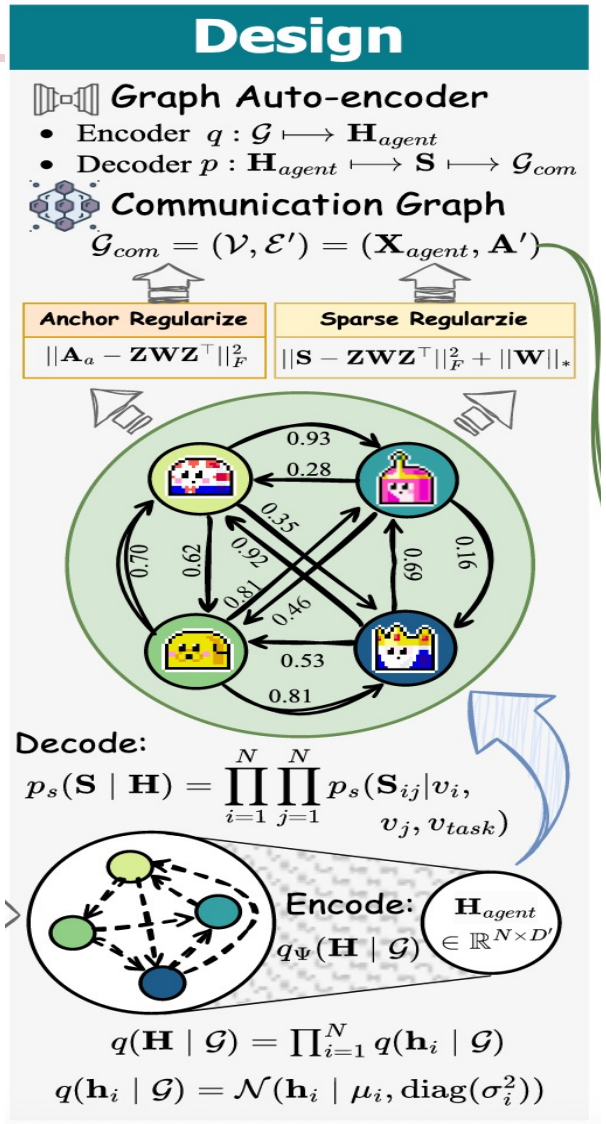
解码器模块　　编码器模块

2）对于解码器，使用隐变量生成多智能体的通信拓扑结构：

$$p(\mathcal{G}_{com} \mid \mathbf{H}_{agent}) = \int_{\mathbf{S}} p_c(\mathcal{G}_{com} \mid \mathbf{S}) p_s(\mathbf{S} \mid \mathbf{H}_{agent}) \, d\mathbf{S}.$$

最终的通信拓扑由下式给出：

$$\mathcal{G}_{com} = (\mathcal{V}, \mathcal{E}_{com}), \mathcal{E}_{com} = \{(i,j) \mid \tilde{\mathbf{S}}_{ij} \neq 0 \wedge (i,j) \in \mathcal{E}\}).$$



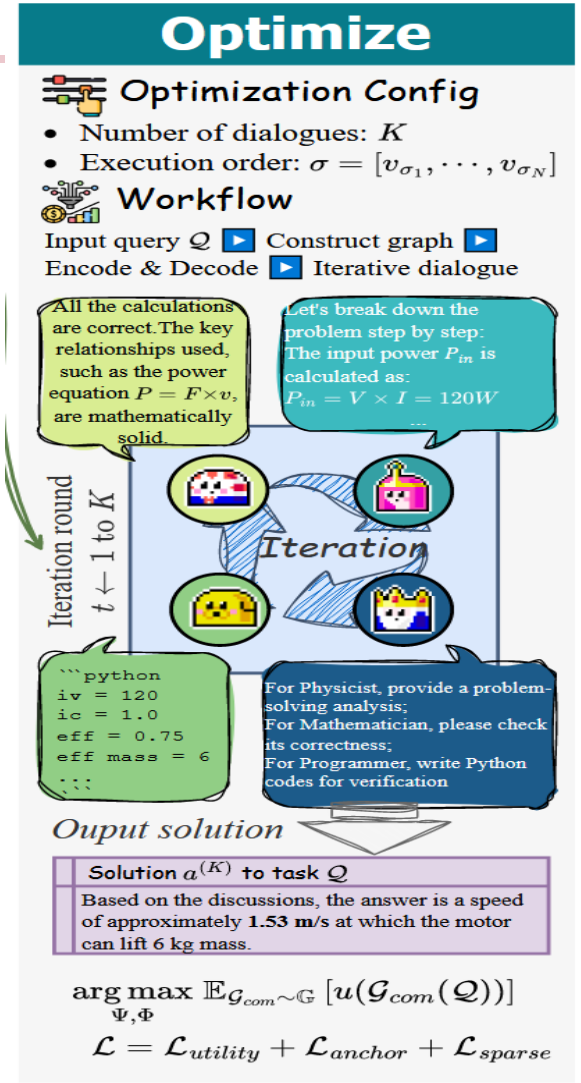
1. 生成式方法：G-Designer

❑ 3.模型优化：

基于得到拓扑结 $\mathcal{G}_{com}$ ，多智能体系统进行对话和信息交互。经过K轮交互，智能体收敛到最终解 。优化目标为：

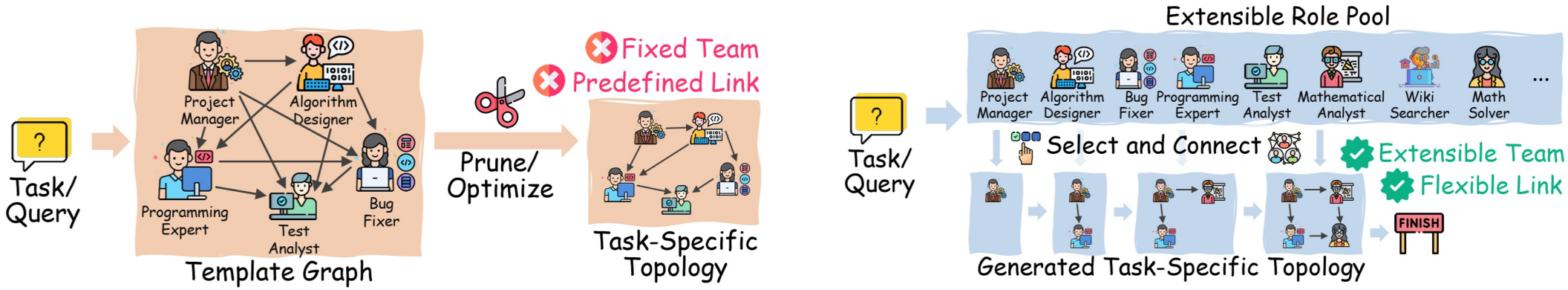
$$\arg\min_{\Theta_e,\Theta_d} \mathbb{E}_{\Theta_e,\Theta_d\sim\Omega}\left[u\left(\mathcal{G}_{com}(\mathcal{Q})\right)\right],$$

综上，G-Designer 能够自适应地为不同的任务和领域设计出高效、鲁棒的通信拓扑，为自组织、自进化的多智能体系统的建立和部署提供了一种全新的解决方案



2. 生成式方法：ARG-Designer

❑ **动机：**传统方法通常采用固定模版，如何让智能体系统像人类专家团队一样，动态决定由哪些智能体以及协作方式?



(a) Template Graph Modification Paradigm (Existing)

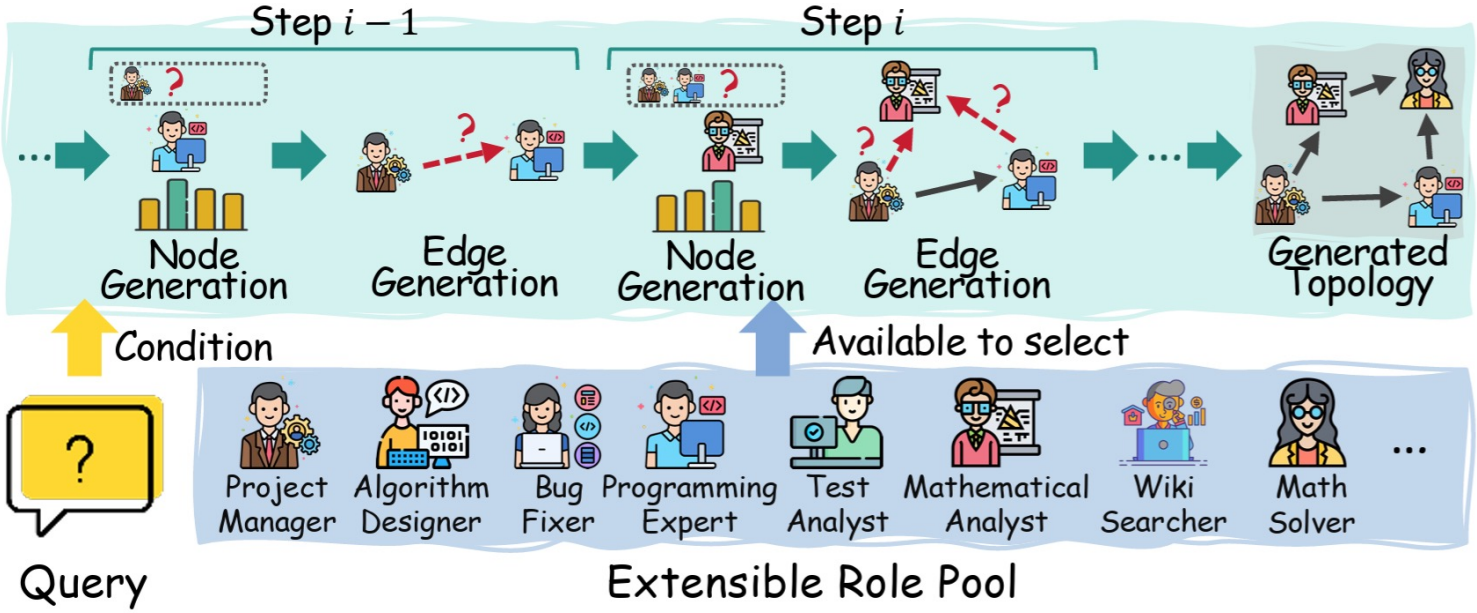
(b) Autoregressive Graph Generation Paradigm (Ours)

ARG-Designer采用**生成式方式按需构建**高效的协作网络

Assemble your crew: Automatic multi-agent communication topology design via autoregressive graph generation. AAAI 2026

2. 生成式方法：ARG-Designer

- ARG-Designer把拓扑设计重构为**自回归图生成任务**
  - **分配智能体角色**：从可扩展的角色池中为各位置选择合适的专家
  - **建立通信链接**：确定新节点与已有成员之间的通信关系并建立连接



(a) Pipeline of ARG-DESIGNER Generation

2. 生成式方法：ARG-Designer

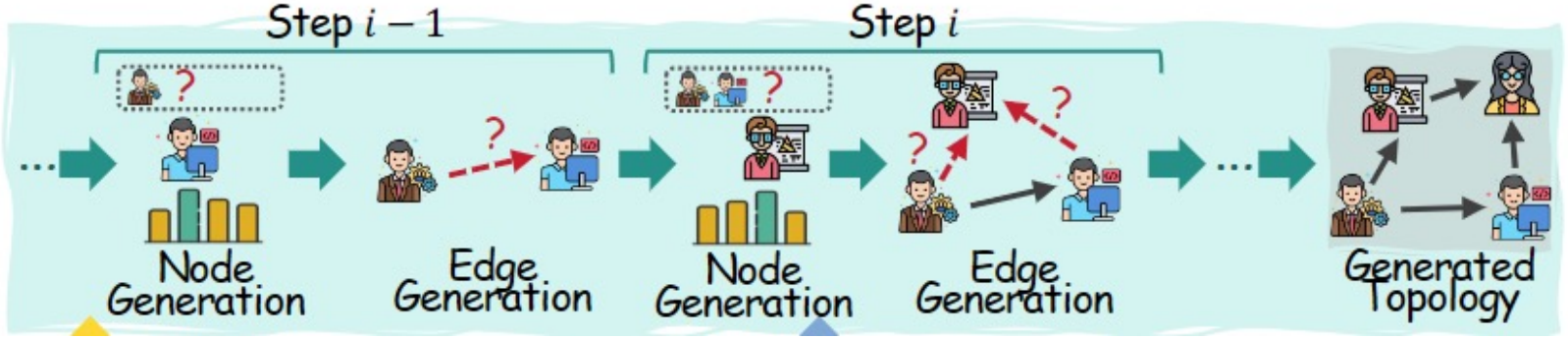
❑ 模块一：分配智能体角色（**节点生成**）

历史嵌入 $\mathbf{f}_{\text{hist}}^{(i)} = \text{GRU}_{\text{prev}}([\mathbf{z}_{r_1}, \mathbf{z}_{r_2}, \ldots, \mathbf{z}_{r_{i-1}}])$ → 智能体角色信息

上下文嵌入 $\mathbf{f}_{\text{cont}}^{(i)} = (1-\mathbf{g}_i)\cdot\mathbf{f}_{\text{hist}}^{(i)} + \mathbf{g}_i\cdot\mathbf{f}_{\mathcal{Q}}, \quad \mathbf{g}_i = \sigma\left(\frac{\mathbf{f}_{\text{hist}}^{(i)}\cdot\mathbf{f}_{\mathcal{Q}}}{\sqrt{d}}\right)$

生成条件 $\mathbf{h}_{\text{node}}^{(i)} = \text{GRU}_{\text{node}}(\text{MLP}_{\text{node}}([\mathbf{f}_{\text{cont}}^{(i)}, \mathbf{f}_{\text{edge}}^{(i)}]), \mathbf{h}_{\text{node}}^{(i-1)})$

节点预测分数 $\mathbf{s}_{\text{node}}^{(i)} = \text{MLP}_{\text{pred\_n}}(\mathbf{h}_{\text{node}}^{(i)}) \cdot \text{MLP}_{\text{role}}([\mathbf{Z}, \mathbf{z}_{\text{end}}])$



2. 生成式方法：ARG-Designer

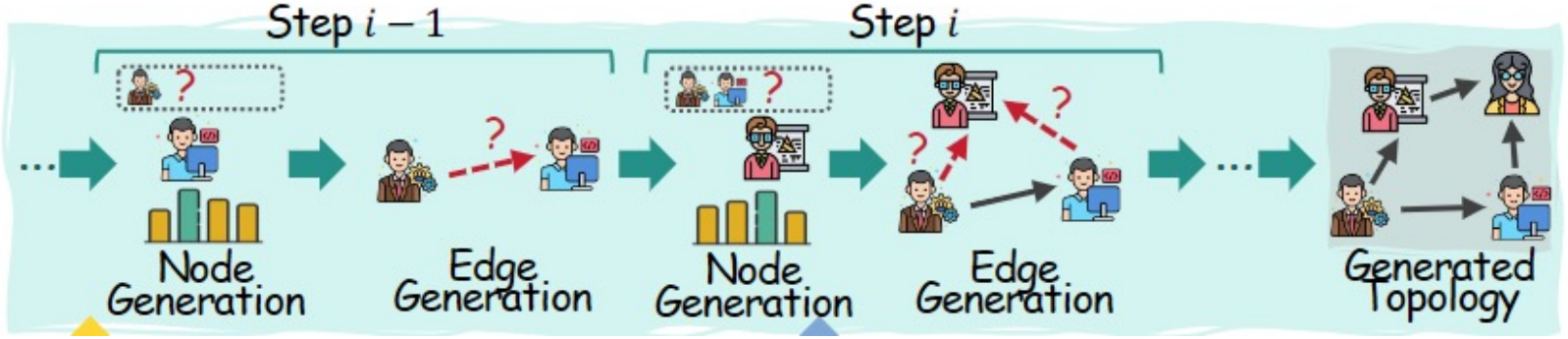
❑ 模块二：建立通信链接（**边生成**）

初始隐藏状态 $\mathbf{h}_{\text{edge}}^{(i,0)} = \text{MLP}_{\text{node2edge}}(\mathbf{h}_{\text{node}}^{(i)})$

更新隐藏状态 $\mathbf{h}_{\text{edge}}^{(i,j)} = \text{GRU}_{\text{edge}}(\text{MLP}_{\text{edge}}(\mathbf{e}^{(j-1,i)}), \mathbf{h}_{\text{edge}}^{(i,j-1)})$

边预测分数 $s_{\text{edge}}^{(i,j)} = \text{MLP}_{\text{pred\_e}}(\mathbf{h}_{\text{edge}}^{(i,j)})$

one-hot 向量，表示从前一个决策是否从节点 v^{j-1} 到 v^{i} 形成边的决策



2. 生成式方法：ARG-Designer

❑ 模型训练：构造数据集

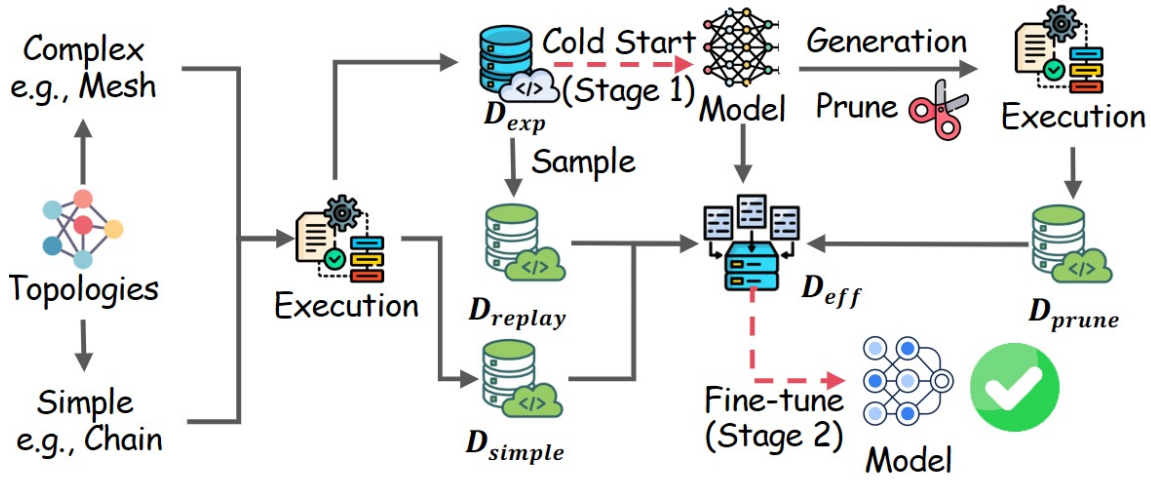
目的：先模型先学会生成正确、丰富的拓扑，再学会把拓扑做得更精简、更高效

- **构造探索集**

  根据人工设定的配置模板造出候选图，然后把任务和复杂图配置配对，执行验证，只保留那些经验上成功的样本

- **构造效率集**

  对上述成功图进行剪枝，例如删除节点或边，如果删除后任务依然成功，则说明原结构存在冗余，于是保留更稀疏的图作为训练样本



(b) Pipeline of Model Training

2. 生成式方法：ARG-Designer

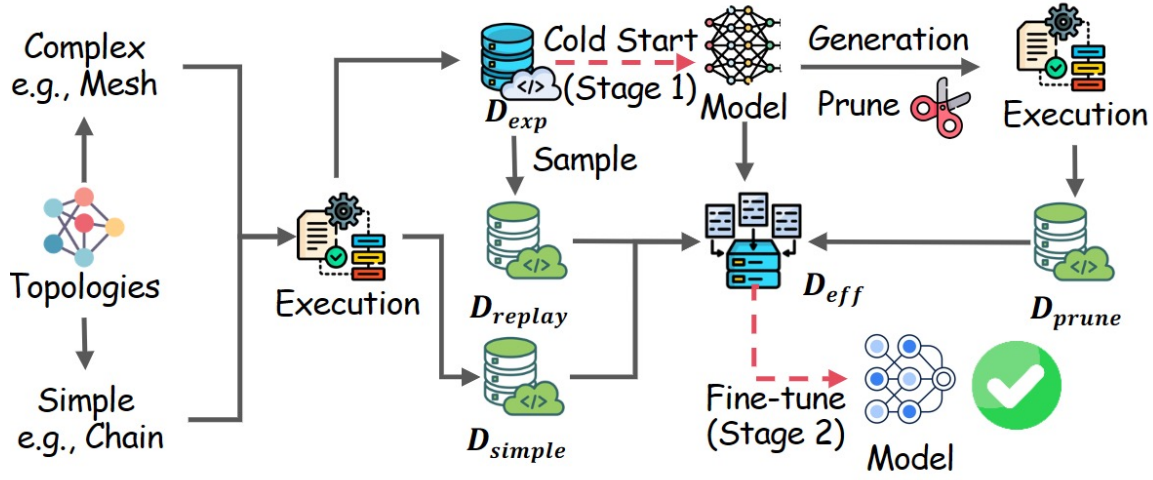
❑ 模型训练-目标优化

在整个训练过程中采用教师强制策略，在每一步向模型输入真实结构，以稳定和加速学习。ARG-Designer按照两阶段过程训练 。首先在 D_{exp}上进行冷启动训练，然后在D_{eff} 上进行微调

$$\mathcal{L}_{\text{node}} = -\sum_{(\mathcal{G},\mathcal{Q})\in\mathcal{D}}\sum_{i=1}^{|\mathcal{V}|}\log P_\theta(v_i|\mathcal{G}_{<i},\mathcal{Q},\mathcal{R}),$$

$$\mathcal{L}_{\text{edge}} = -\sum_{(\mathcal{G},\mathcal{Q})\in\mathcal{D}}\sum_{i=1}^{|\mathcal{V}|}\sum_{j=1}^{i-1}\log P_\theta(e_{ji}|v_i,\mathcal{G}_{<i},\mathcal{Q}).$$

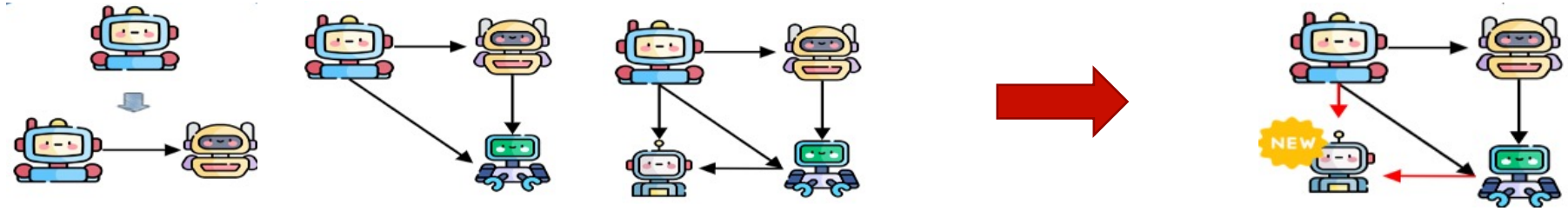
$$\mathcal{L}_{\text{total}} = \alpha\cdot\mathcal{L}_{\text{node}} + (1-\alpha)\cdot\mathcal{L}_{\text{edge}}$$



(b) Pipeline of Model Training

3. 生成式方法—OFA-MAS

- 动机：现有生成式拓扑设计大多仍是“one-for-one”范式——每个任务域单独训练一个专用拓扑生成模型，缺乏“**one-for-all**”的统一模型，用一个生成器面向多域任务生成协作拓扑

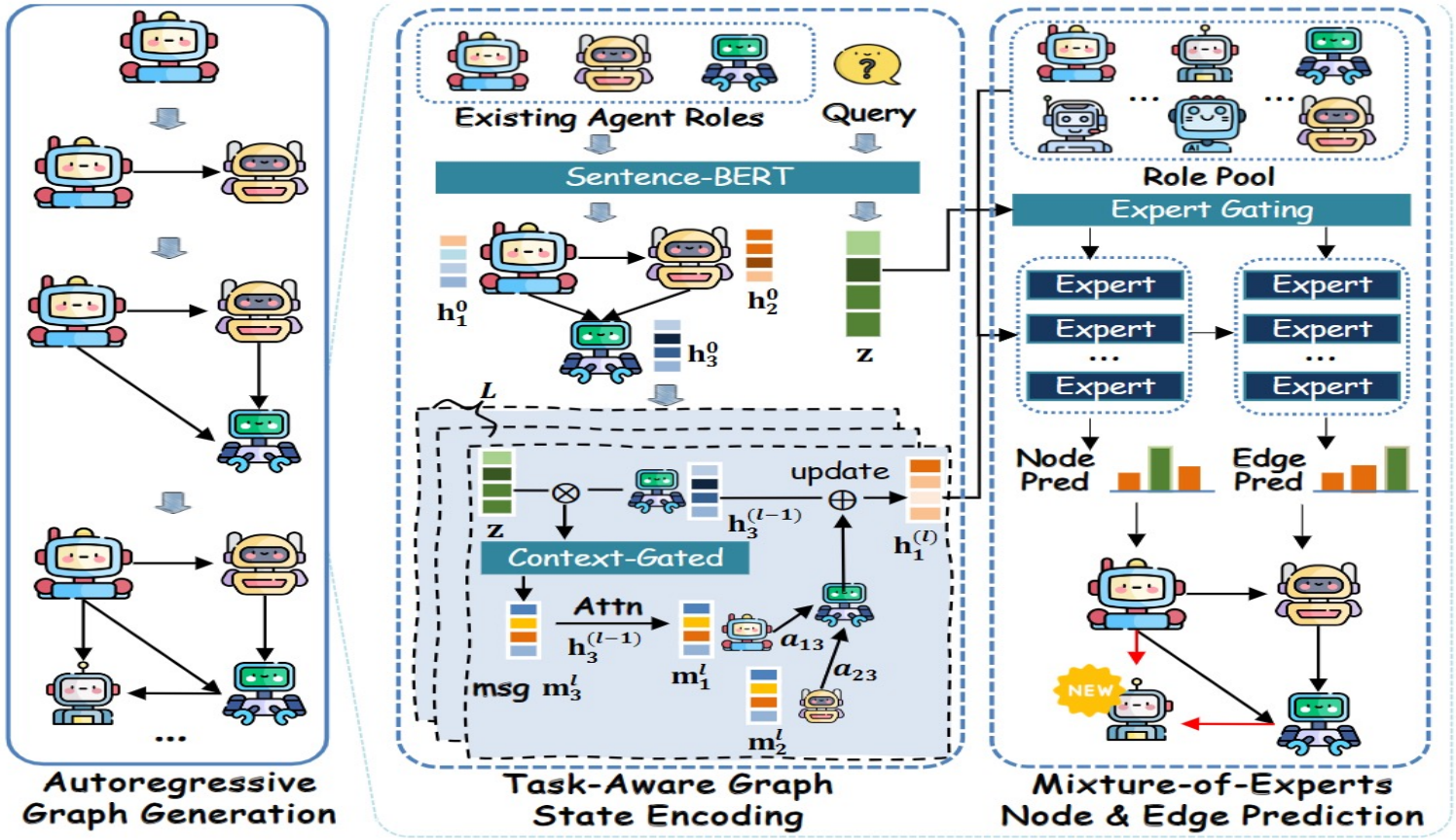


OFA-MAS从单任务专用拓扑生成，走向**跨任务共享的统一拓扑生成模型**

OFA-MAS: One-for-All Multi-Agent System Topology Design based on Mixture-of-Experts Graph Generative Models. WWW 2026

3. 生成式方法—OFA-MAS

❑ 模型框架图：任务感知图状态编码器和专家混合生成模块



A Survey of Large Language Models. 2023.

3. 生成式方法—OFA-MAS

❑ 模块一：任务感知图状态编码器

融合当前任务信息与邻域结构信息，得到任务相关的节点表示，再用于后续的节点和边预测

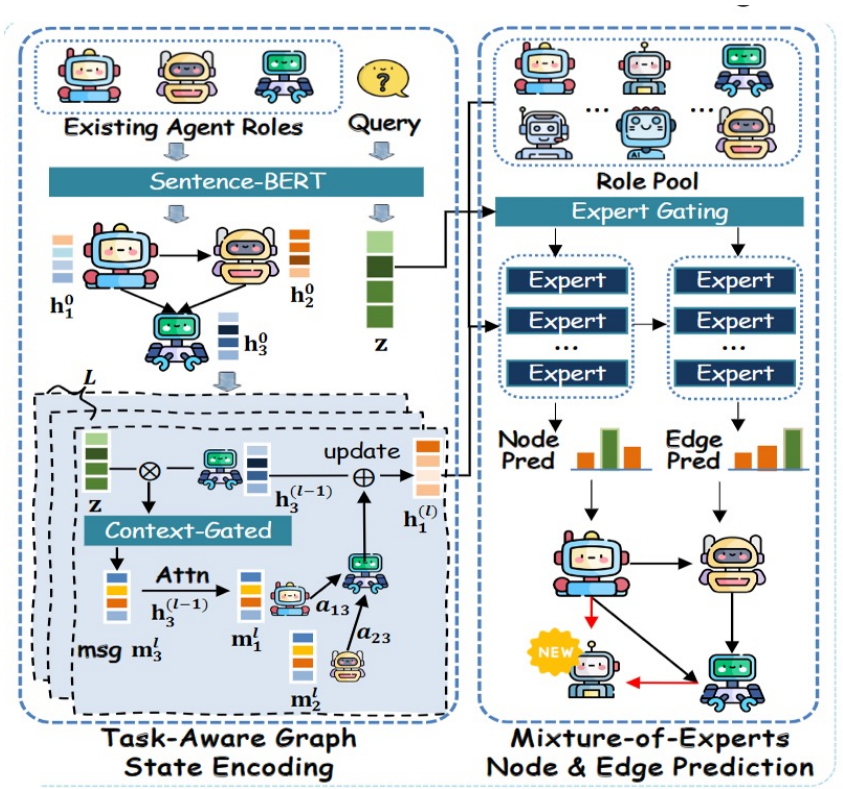
- 融合任务信息和邻域信息：

$$\mathbf{m}_v^{(l)} = \sigma(\mathbf{W}_g[\mathbf{h}_v^{(l-1)} \| \mathbf{z}]) \odot \mathrm{ReLU}(\mathbf{W}_m \mathbf{h}_v^{(l-1)}),$$

- 计算节点表示：

$$\mathbf{h}_v^{(l)} = \frac{1}{2}\left(\mathbf{h}_v^{(l-1)} + \hat{\mathbf{m}}_v^{(l)}\right) \qquad \hat{\mathbf{m}}_v^{(l)} = \sum_{u \in \mathcal{N}_v} \alpha_{uv}^{(l)} \mathbf{m}_u^{(l)}$$

$$\alpha_{uv}^{(l)} = \frac{\exp(e_{uv}^{(l)})}{\sum_{j \in \mathcal{N}_v} \exp(e_{jv}^{(l)})} \qquad e_{uv}^{(l)} = \mathrm{LeakyReLU}(\mathbf{a}^T[\mathbf{W}_k \mathbf{h}_u^{(l-1)} \| \mathbf{W}_q \mathbf{m}_v^{(l)}])$$



3. 生成式方法—OFA-MAS

❑ 模块二：专家混合生成模块

让不同专家子网络学不同的拓扑生成偏好，再由路由机制针对具体任务动态选专家

- 节点生成：

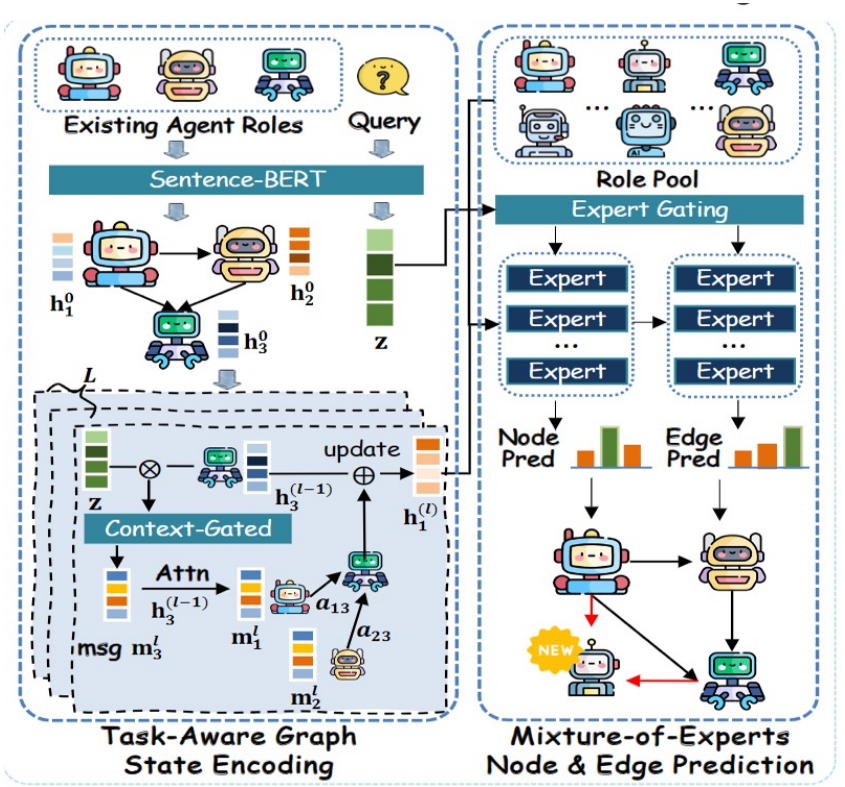
通过节点表示的池化操作获得的局部图的全局表示

$$P(\text{role}_t|\mathcal{G}_{<t}, \mathbf{z}) = \sum_{k=1}^{K} w_k \cdot \text{Expert}_k^{\text{node}}(\text{GlobalPool}(\mathcal{G}_{<t}), \mathbf{z})$$

- 边生成：

$$P(\text{edge}_{j\to t}|\mathcal{G}_{<t}, \mathbf{z}) = \sum_{k=1}^{K} w_k \cdot \text{Expert}_k^{\text{edge}}([\mathbf{h}_j \| \mathbf{h}_t \| \mathbf{z}])$$

源节点和目标节点的表示



3. 生成式方法—OFA-MAS

❑ 模型训练-三阶段：

- **无条件图预训练**

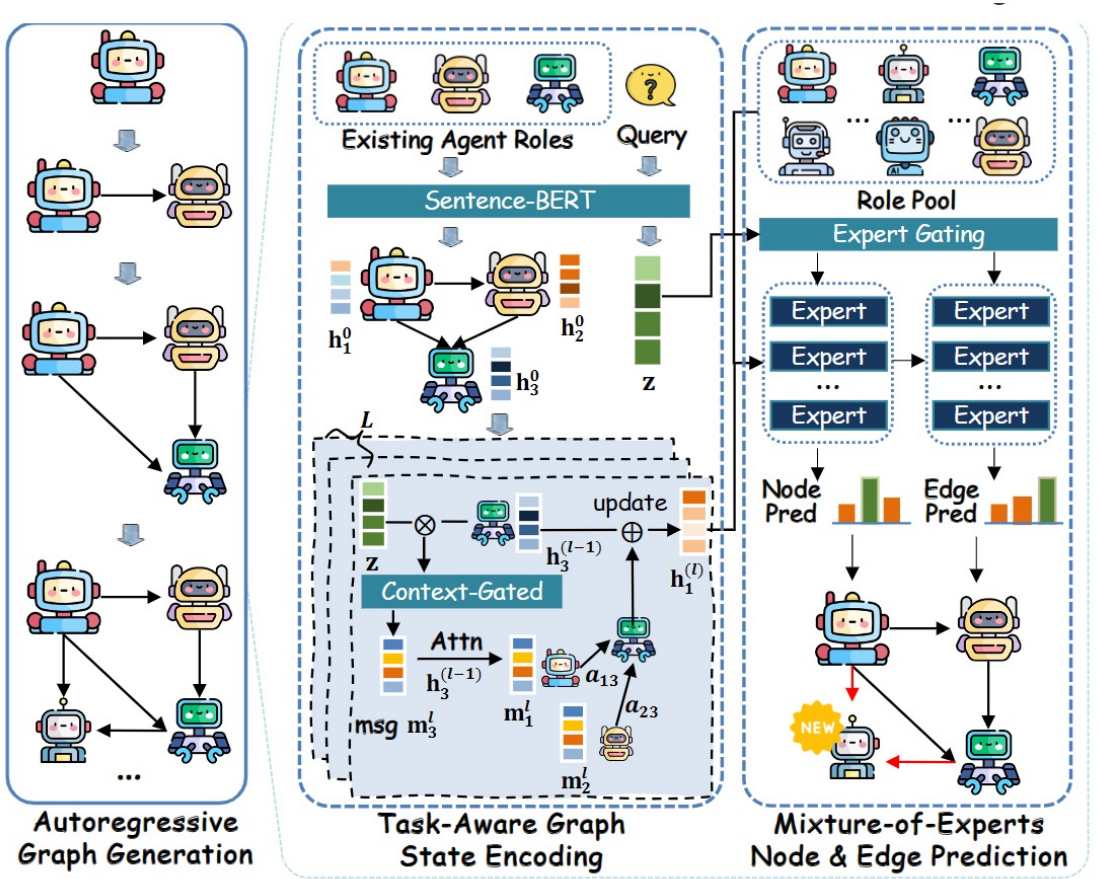
  先在常规拓扑（星型、链式等）上做无条件预训练，让模型学习基本的结构先验

- **任务拓扑对齐**

  在LLM生成的数据上做条件预训练，学习从“任务描述”到“拓扑结构”的映射

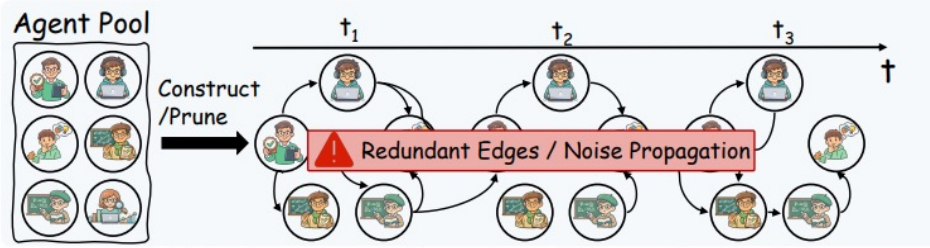
- **监督微调**

  精心构建数据集，其中包含（任务查询，高性能拓扑）

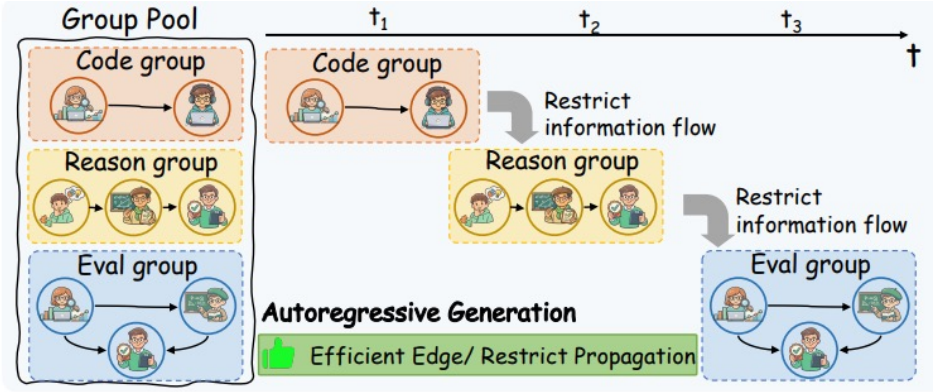


4. 生成式方法—GoAgent

- 动机：现有生成式拓扑方法多以单个智能体为单位构造通信图，但复杂任务更需要先形成协作小组，再进行组间协同
- 若群组结构仅靠局部连边隐式形成，容易导致协作不足与通信冗余，因此需要将协作群组显式作为拓扑生成的基本单元



(a) Node-centric Paradigm (Exist Methods)



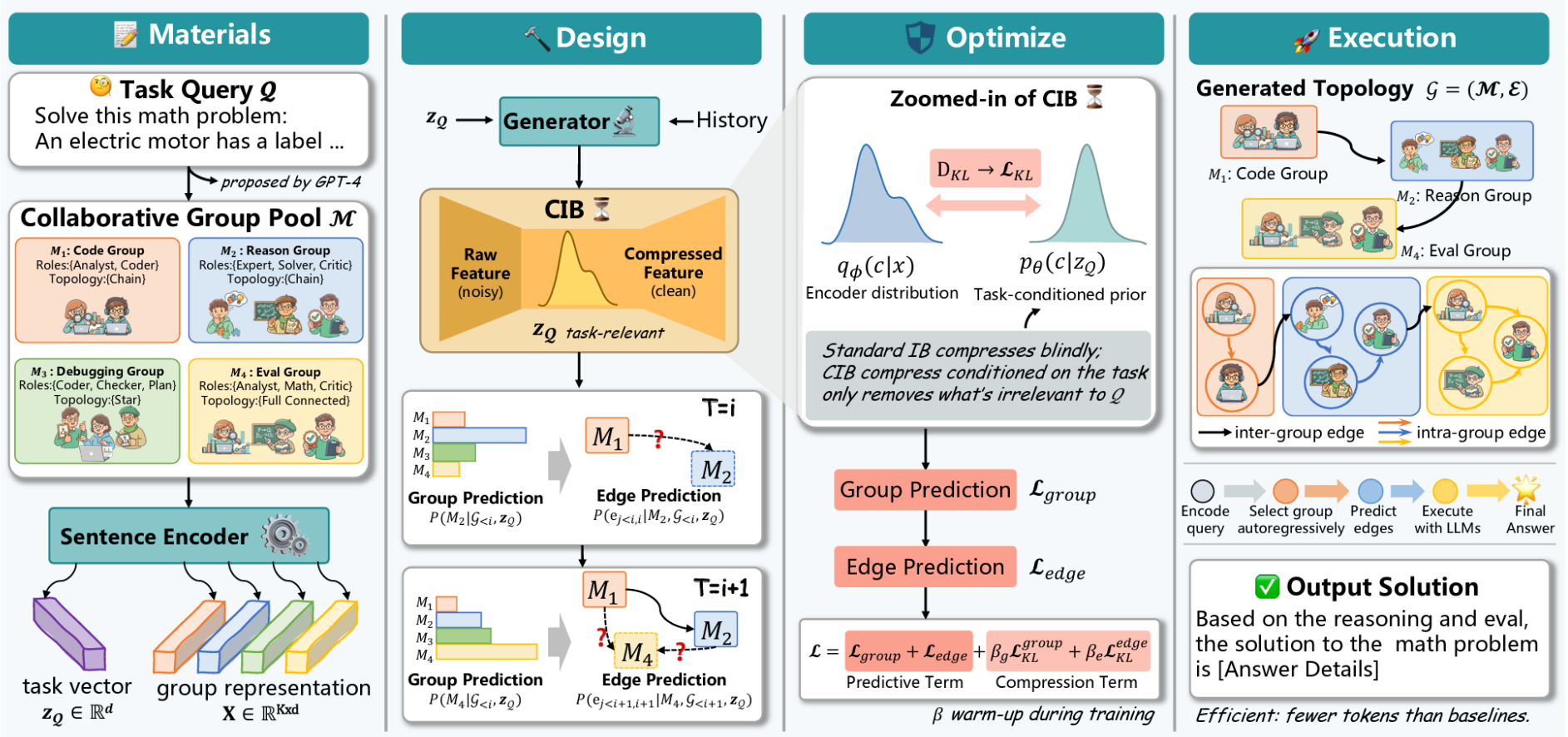
(b) Group-centric Paradigm (Ours)

GoAgent从节点级拓扑生成，走向**群组级协作拓扑生成**

GoAgent: Group-of-Agents Communication Topology Generation for LLM-based Multi-Agent Systems. 2026

4. 生成式方法—GoAgent

- 整体框架：候选群组生成、群组拓扑生成、拓扑优化与执行



4. 生成式方法—GoAgent

❑ 模块一：候选群组生成

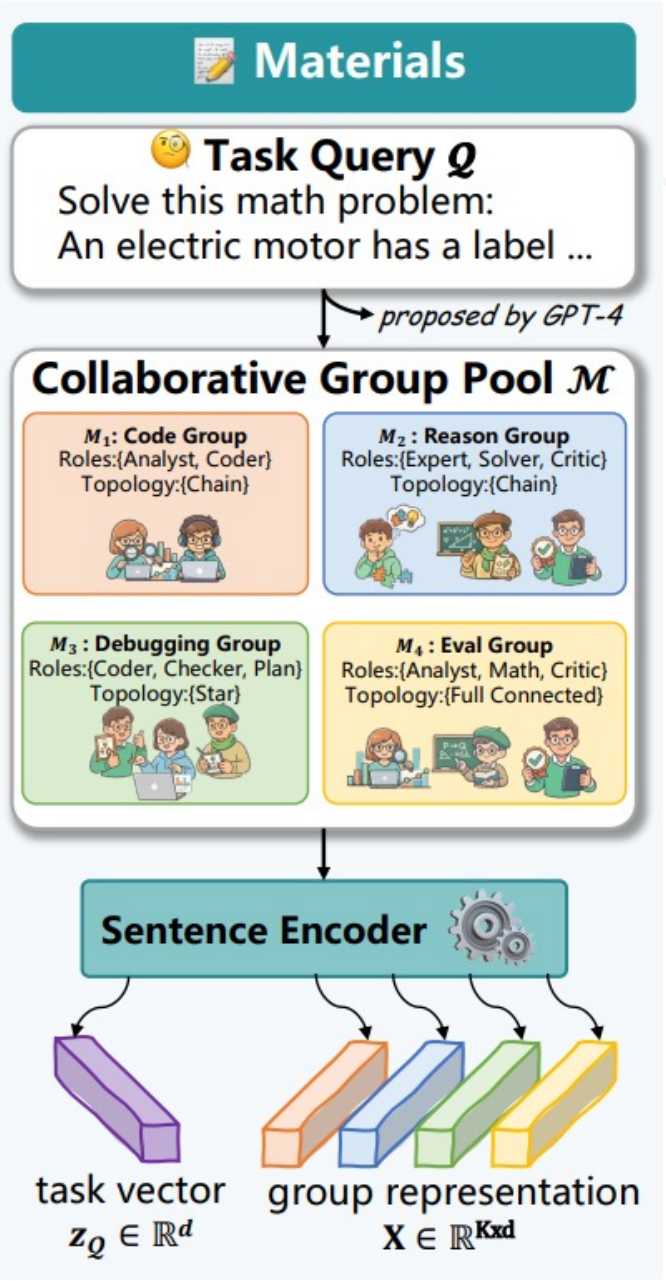
- 协作群组发现：

  由 LLM 从任务相关的候选群组池中发现并定义带固定组内拓扑的协作群组

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_K] \in \mathbb{R}^{K \times d}$$

- 任务编码：

$$\mathbf{z}_{\mathcal{Q}} = \mathrm{FFN}(\mathrm{SentenceEncoder}(\mathcal{Q}))$$



4. 生成式方法—GoAgent

❑ 模块二：群组拓扑生成

- 历史聚合：

先前生成的组嵌入

$$\mathbf{h}_{\text{his}}^{(t)} = \text{GRU}([\mathbf{x}_{M_1}, \ldots, \mathbf{x}_{M_{t-1}}])$$

$$g^{(t)} = \sigma\left(\frac{\mathbf{h}_{\text{his}}^{(t)} \cdot \mathbf{z}_{\mathcal{Q}}}{\sqrt{d}}\right) \quad \mathbf{h}_{\text{comb}}^{(t)} = (1 - g^{(t)})\mathbf{h}_{\text{his}}^{(t)} + g^{(t)}\mathbf{z}_{\mathcal{Q}} + \mathbf{e}_{\text{pos}}^{(t)}$$

- 群组预测

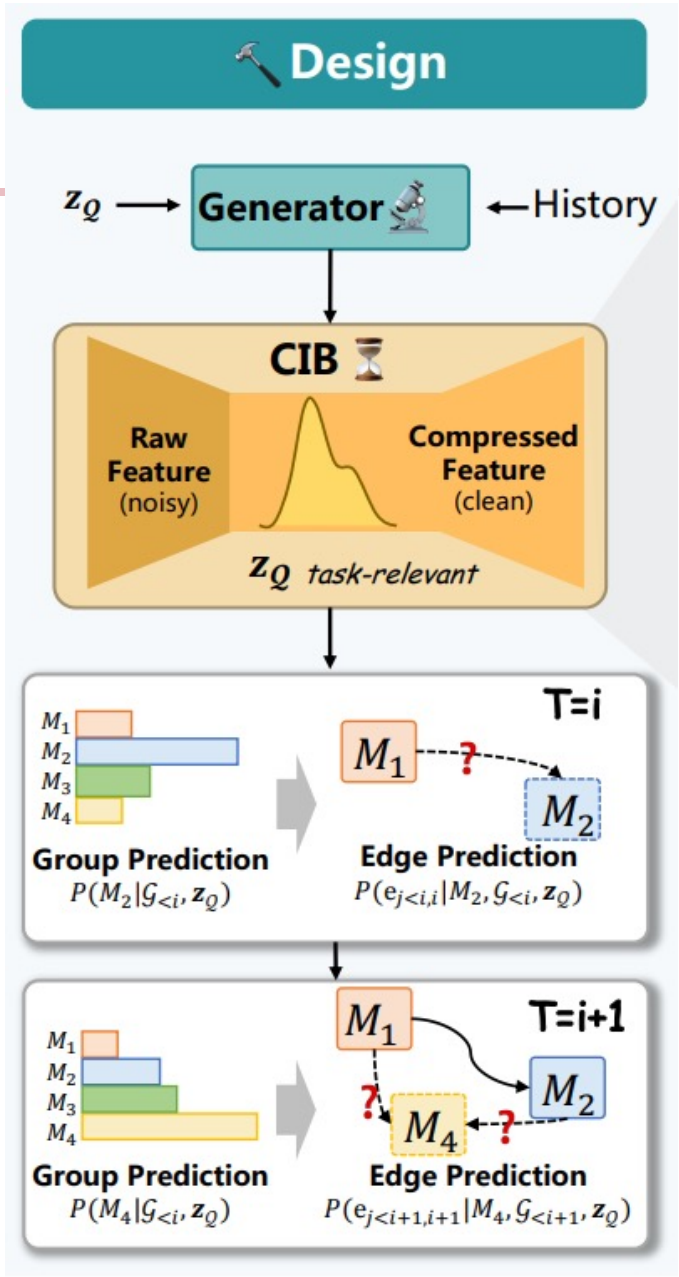
$$h_{comb}^{(t)} \xrightarrow{\text{GRU编码}} x_{group}^{(t)} \xrightarrow{\text{CIB压缩与去噪}} c_{group}^{(t)}$$

$$P(M_t \mid G_{<t}, z_Q) = \text{Softmax}(c_{group}^{(t)} X^\top)$$

- 连边预测

$$x_{edge}^{(i,t)} = [\, h_{comb}^{(i)} \,\|\, x_{M_t} \,\|\, z_Q \,] \xrightarrow{\text{CIB压缩与去噪}} c_{edge}^{(i,t)}$$

$$P(e_{i,t} = 1 | M_i, \mathcal{G}_{<t}, \mathbf{z}_{\mathcal{Q}}) = \sigma(\text{MLP}(\mathbf{c}_{\text{edge}}^{(i,t)}))$$



4. 生成式方法—GoAgent

❑ 模块二：群组拓扑生成

由于历史通信图中会不断累积无关信息，因此作者利用信息瓶颈思想，只保留对当前预测真正有帮助的信息

$$\min \mathcal{L}_{\text{CIB}} = \underbrace{-I(\mathbf{c}; y|\mathbf{z}_Q)}_{\text{Predictive Term}} + \beta \underbrace{I(\mathbf{x}; \mathbf{c}|\mathbf{z}_Q)}_{\text{Compression Term}}$$

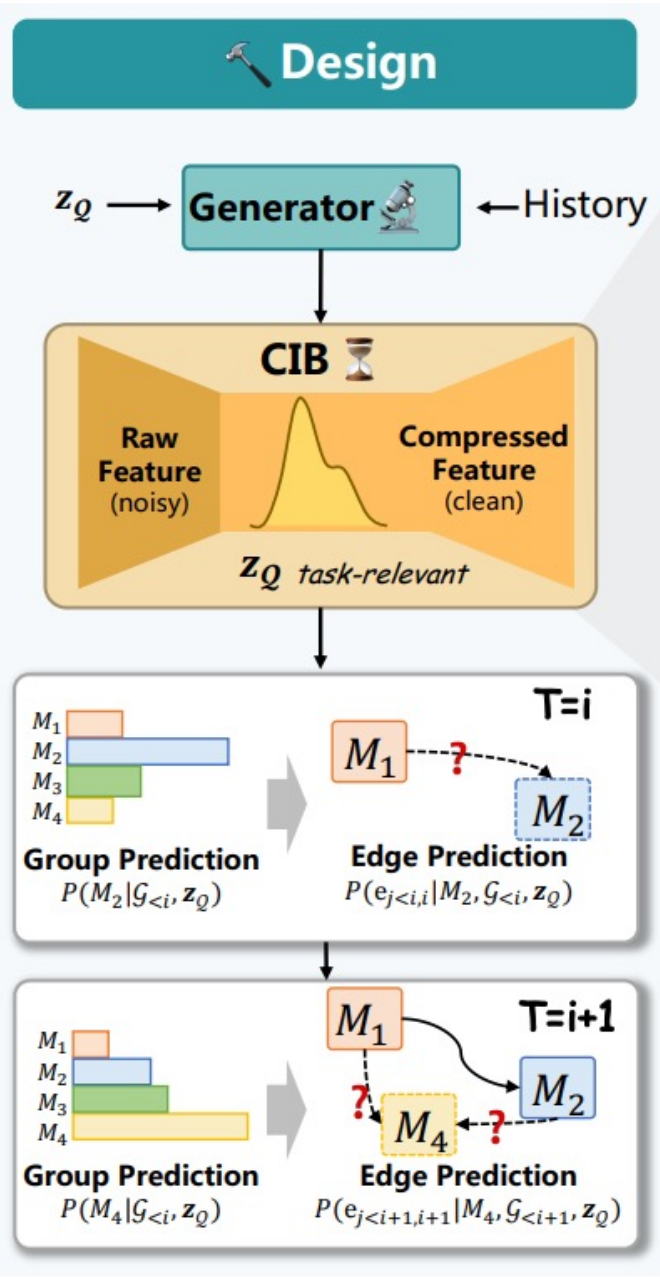
压缩后的表示 c和目标 y 在给定任务的条件下尽量相关

限制从原始特征 x到压缩表示 c的信息流

$$-I(c; y|z_Q) \leq \mathbb{E}_{c\sim q_\phi(c|x)}[-\log p_\psi(y|c, z_Q)] \triangleq L_{task}$$

$$I(x; c|z_Q) \leq \mathbb{E}_{x,z_Q}\left[D_{KL}(q_\phi(c|x)\|p_\theta(c|z_Q))\right] \triangleq L_{KL}$$

$$L_{CIB} \leq L_{task} + \beta L_{KL}$$



4. 生成式方法—GoAgent

❑ 模块三：拓扑优化

- 数据集构建： $\mathcal{D} = \{(\mathcal{Q}, \mathcal{G}^*)\}$

  Step1：对训练任务采样很多候选拓扑

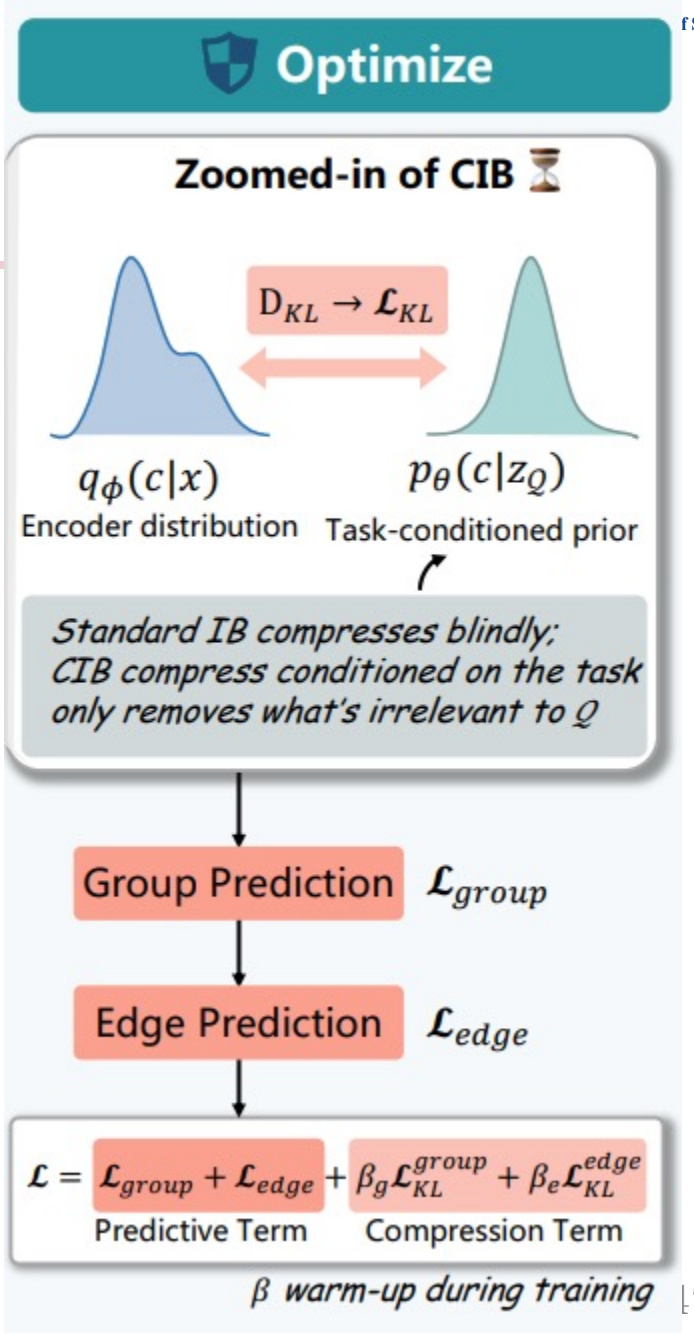
  Step2：用这些候选拓扑去真实执行任务

  Step3：保留成功图并选最小可行拓扑

- 优化目标：

$$L_{group} = -\mathbb{E}_D \left[\sum_t \log P(M_t^* | G_{<t}^*, z_Q) \right] \quad L_{edge} = -\mathbb{E}_D \left[\sum_t \sum_{i<t} \log P(e_{i,t}^* | M_i^*, G_{<t}^*, z_Q) \right]$$

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{group}} + \mathcal{L}_{\text{edge}}}_{\text{Predictive Task}} + \underbrace{\beta_g \mathcal{L}_{\text{KL}}^{\text{group}} + \beta_e \mathcal{L}_{\text{KL}}^{\text{edge}}}_{\text{Information Compression}}$$



中国科学院大学
University of Chinese Academy of Sciences

目 录

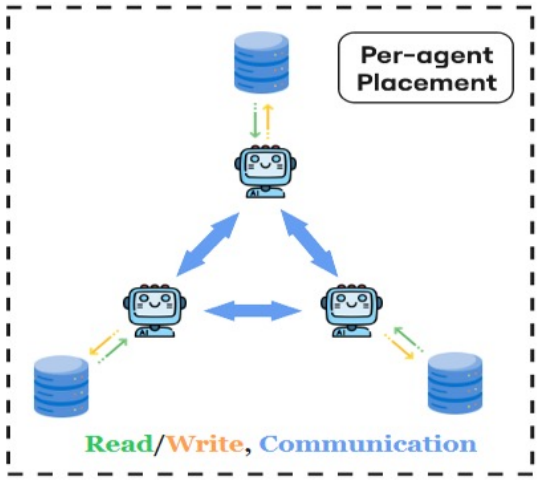
为什么LLM-MAS需要记忆

- 让多智能体系统从“即时协作”走向“持续协作”，从“信息交换”走向“经验积累”，从而支撑更高效、更一致、更智能的群体行为

  - **持续协作能力**：支持跨轮次、跨阶段的任务推进
  - **高效协同能力**：减少冗余通信与重复劳动
  - **全局一致性**：帮助不同智能体共享目标、事实与状态
  - **经验积累能力**：支持知识沉淀、策略复用与长期优化
  - **群体智能涌现**：推动系统从局部交互走向集体认知

LLM-MAS记忆分类

本地记忆

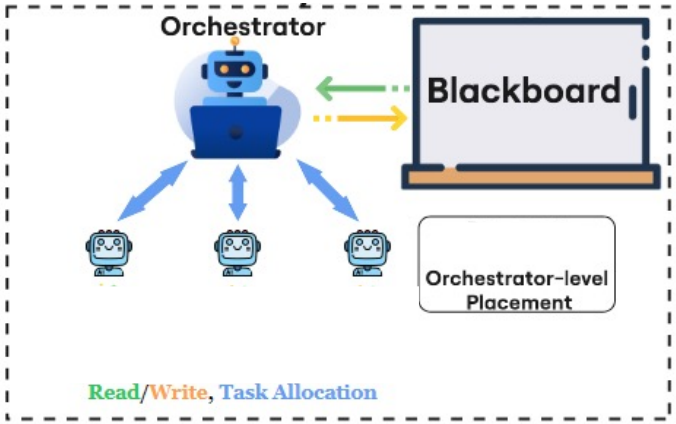


本地记忆是单个智能体内部维护的私有记忆模块，仅该智能体自身可以读取和写入

共享记忆

共享记忆是一个可被多个智能体共同访问的记忆资源，用于作为团队层面的公共知识库或全局记忆空间

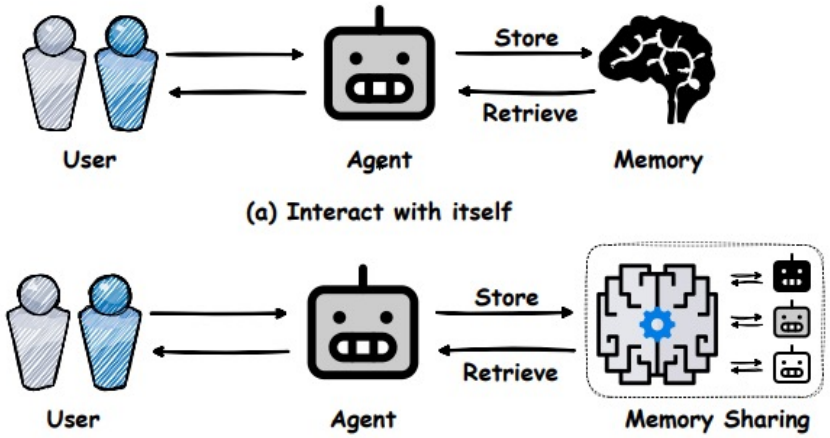
黑板



黑板是一种经典的协作架构，它将共享工作空间与控制/调度机制结合起来，用于支持多个智能体围绕同一公共状态进行迭代求解

1. 共享记忆—INMS

- 单个智能体在开放任务中受限于静态上下文和孤立记忆，缺少多智能体之间的动态经验共享



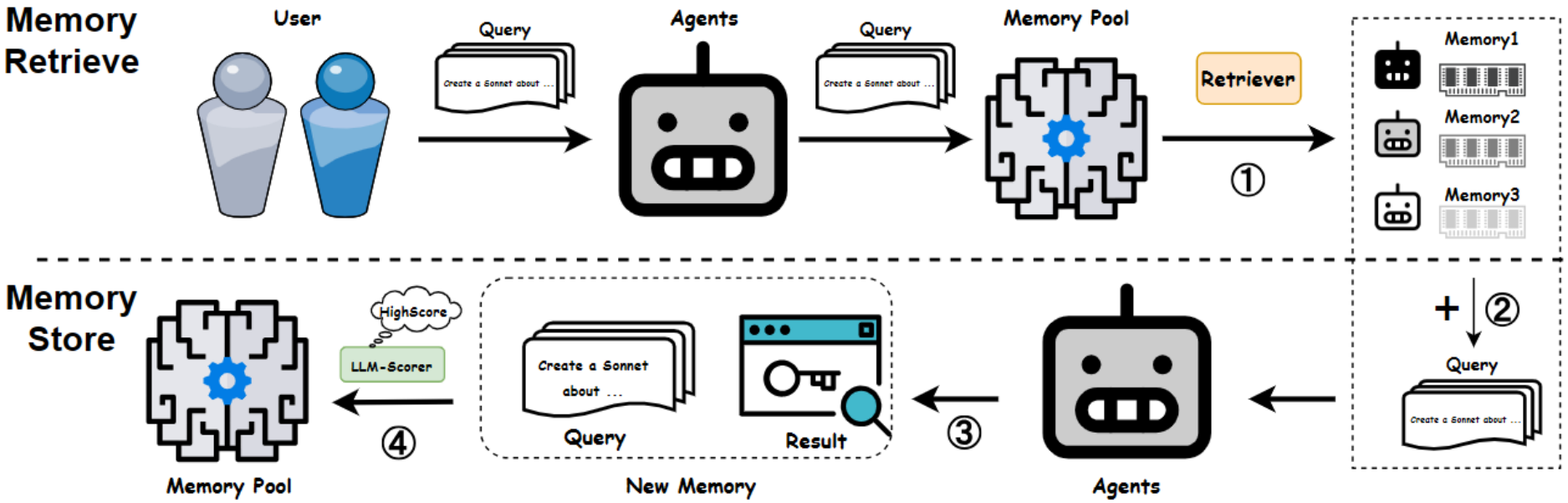
尝试通过**交互式记忆共享机制**，把分散的个体经验转化为可复用的群体知识

INMS-从个体记忆驱动，走向共享记忆驱动的多智能体协作

Memory sharing for large language model based agents. 2024

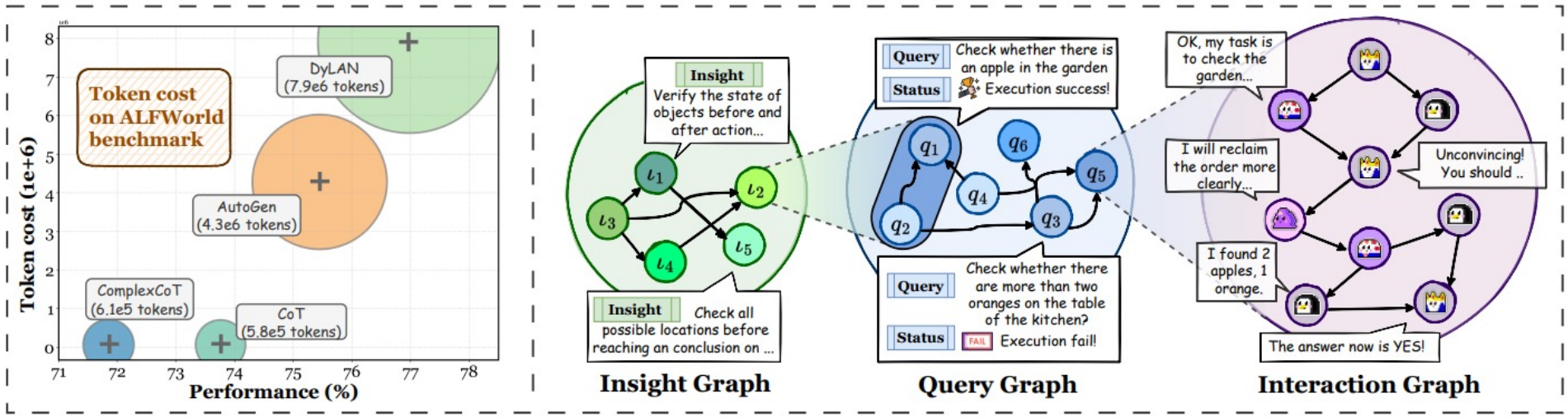
1. 共享记忆—INMS

- 从个体记忆驱动，走向共享记忆驱动的多智能体协作



2. 共享记忆—G-Memory

- 动机：传统记忆通常只是简单存储文本，而多智能体协作过程本身天然具有图结构



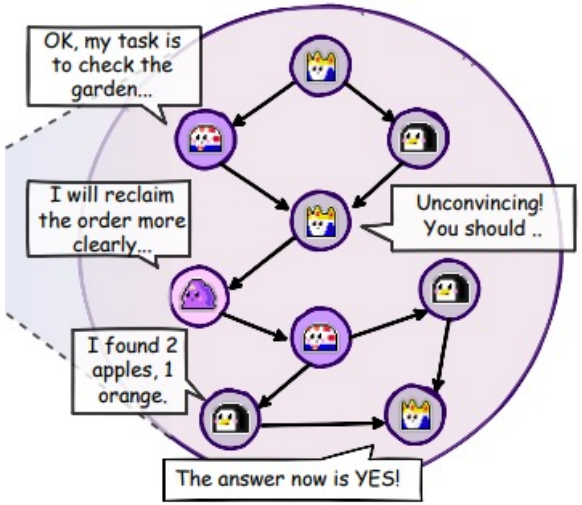
G-Memory将多智能体历史组织成**图结构形式**，从而支持跨任务

G-Memory: Tracing Hierarchical Memory for Multi-Agent Systems. NIPS 2025

2. 共享记忆—G-Memory

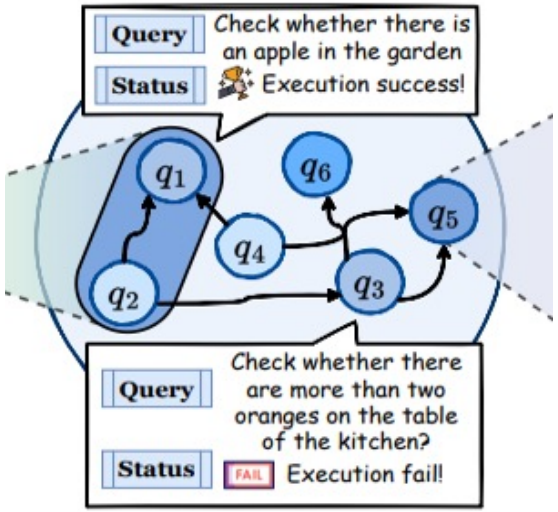
- 多智能体历史协作过程的三层图结构

交互图	任务图	经验图



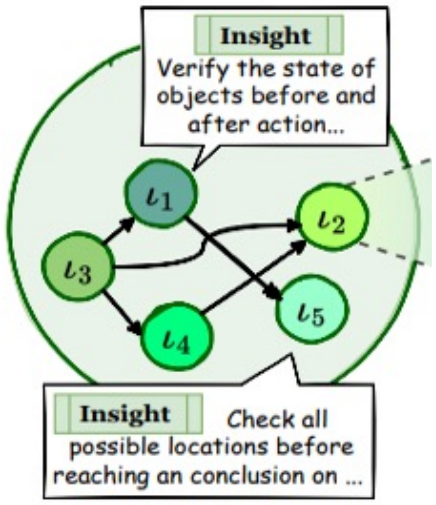
Interaction Graph

细粒度协作轨迹



Query Graph

历史任务记忆

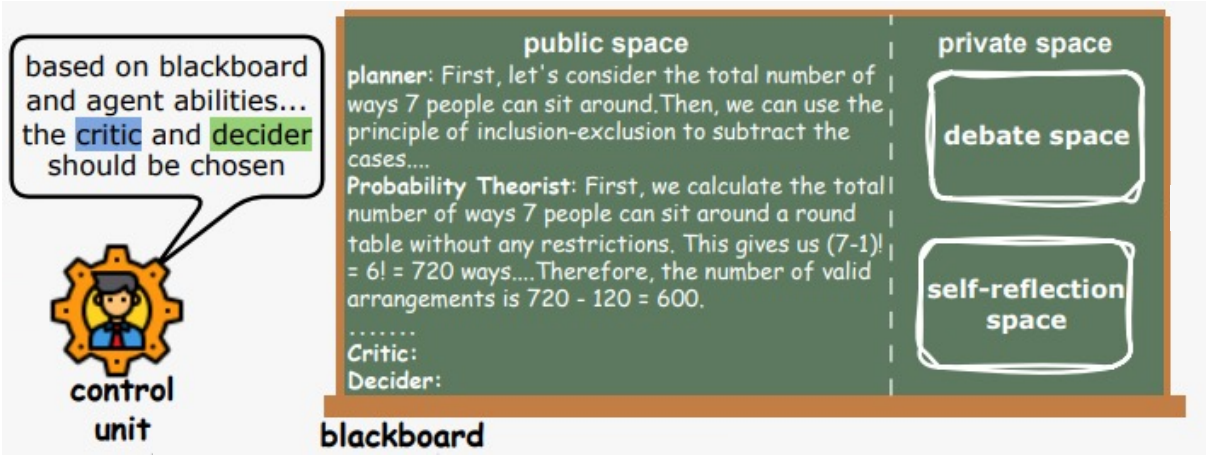


Insight Graph

高层经验总结

黑板—bMAS

- 动机：现有多智能体系统大多依赖固定或预设的协作结构，而一些“先搜索/训练再部署”的动态方法又存在额外训练开销大、搜索空间受限、难以覆盖复杂协作形态的问题

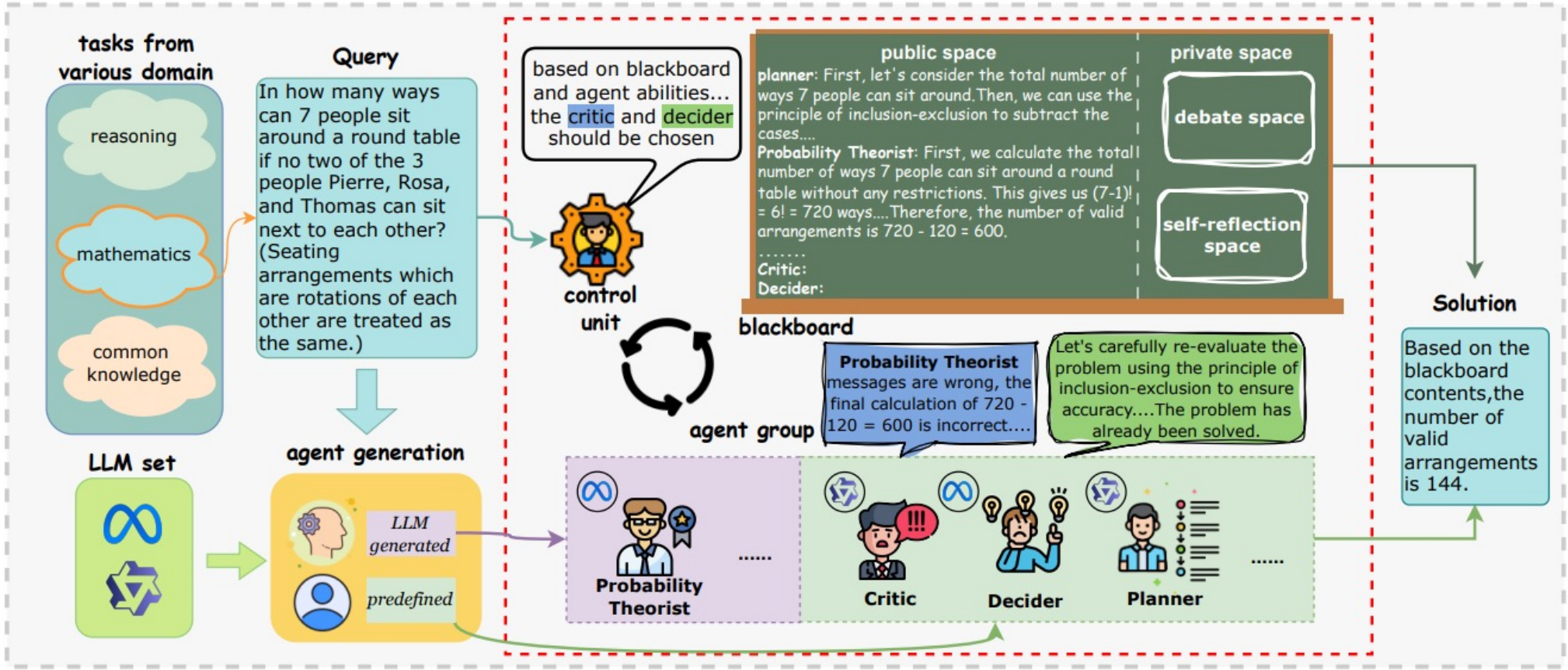


bMAS**基于黑板架构进行信息共享**，实现动态、多轮、多角色的协作

Exploring advanced llm multi-agent systems based on blackboard architecture. 2025

黑板—bMAS

- 利用黑板架构实现信息状态共享的动态、多轮、多角色协作



黑板—bMAS

❑ 智能体生成

$$\{(E_1, D_1), ..., (E_n, D_n)\} = AG(q, I)$$

输入是当前问题 q 和一条专家生成指令 I，输出是 n 个专家 prompt

❑ 黑板循环

- 智能体组　预定义智能体（决策者、规划者、评论家、冲突解决者和清理者）+ 生成的专家
- 黑板　黑板公共空间作为共享记忆，每个 LLM 智能体都可以从中读取和写入
- 控制单元　管理智能体之间的信息流和任务执行，确保根据黑板上的当前消息采取正确的行动。

控制单元动态调度整个协作流程，循环会一直持续，直到达到最大迭代次数，或者 决策者判断出正确答案

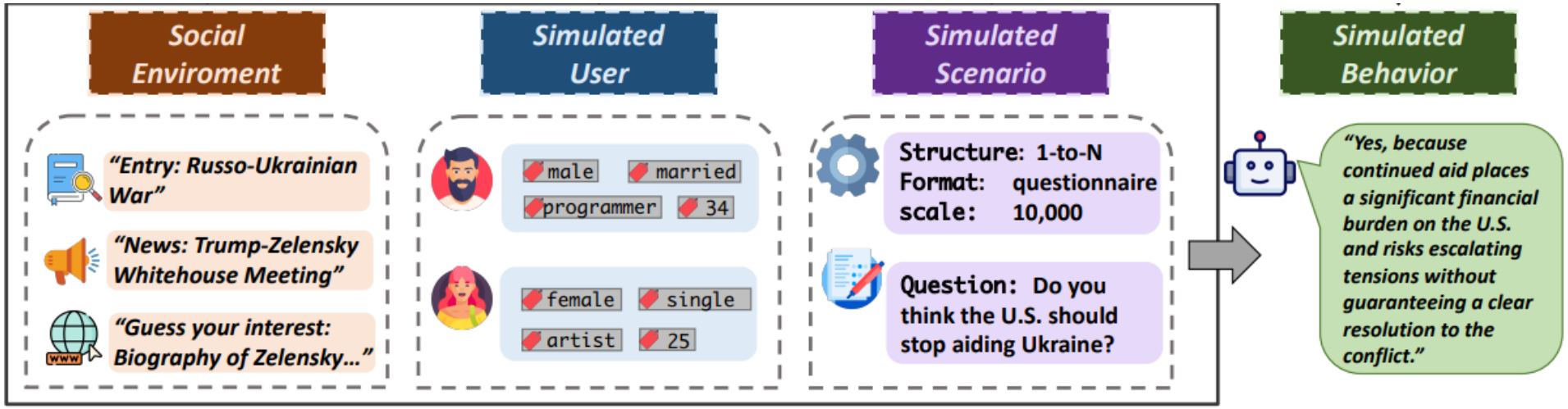
中国科学院大学
University of Chinese Academy of Sciences

目 录

什么是智能体社会仿真

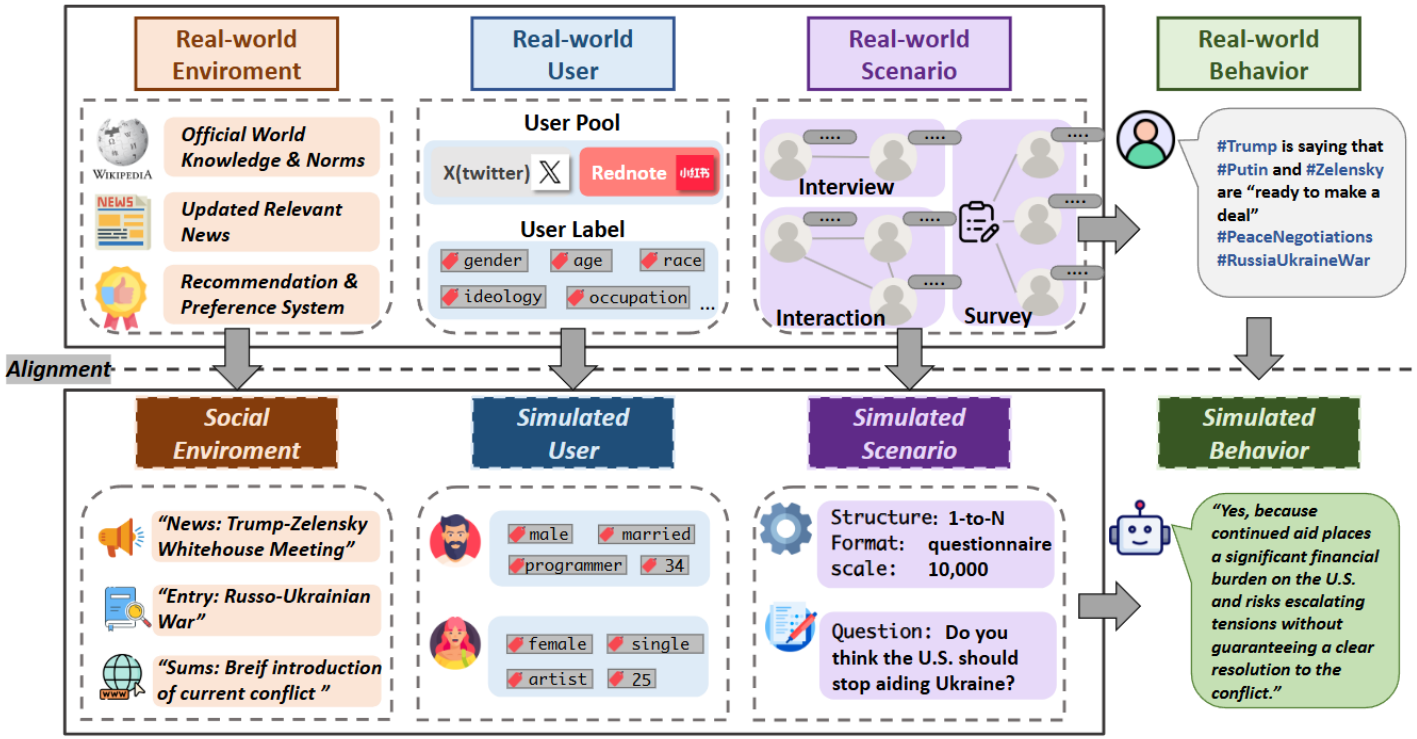
- 基于智能体的建模（Agent-Based Modeling, **ABM**）为核心方法，模拟社会系统中个体行为及其互动过程

核心思想：把社会看作由大量个体组成的系统，让每个个体依据局部规则去行动、决策和互动，再去观察整体层面会涌现出什么样的结果。



为什么需要智能体社会仿真

- 真实社会系统高度复杂，真实实验的成本往往很高，干预可能引发危险行为、伦理风险等问题



真实社会平台

社会仿真平台

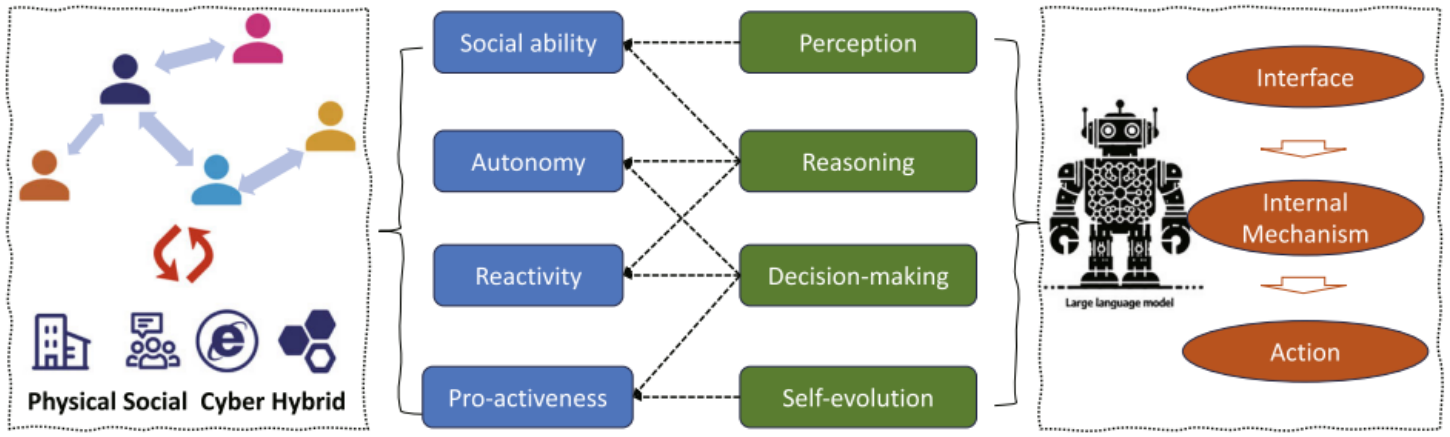
基于LLM的智能体社会仿真

❑ 传统的ABM局限：

- 依赖预设状态变量与固定行为规则；
- 难以刻画真实个体的复杂认知过程；
- 在物理、社会、网络与混合环境中表现有限。

❑ 基于LLM的ABM：

- 支持感知、推理、决策与自我演化；
- 生成行动的具体依据并自主参与社会交互；
- 体现社会能力、自主性、反应性与主动性。



Requirements for Agent-based Modeling and Simulation

Large Language Model Agents' Abilities

社会仿真正在
从**规则驱动**走向**认知驱动**！

研究意义

❑ 智能体社会仿真的研究意义

- **第一类：研究意义（理解社会现象机制）**

  通过社会仿真分析复杂、动态的社会过程，解释**意见极化**、**情绪传播**、**社会运动**等现象，揭示群体行为的演化机制。

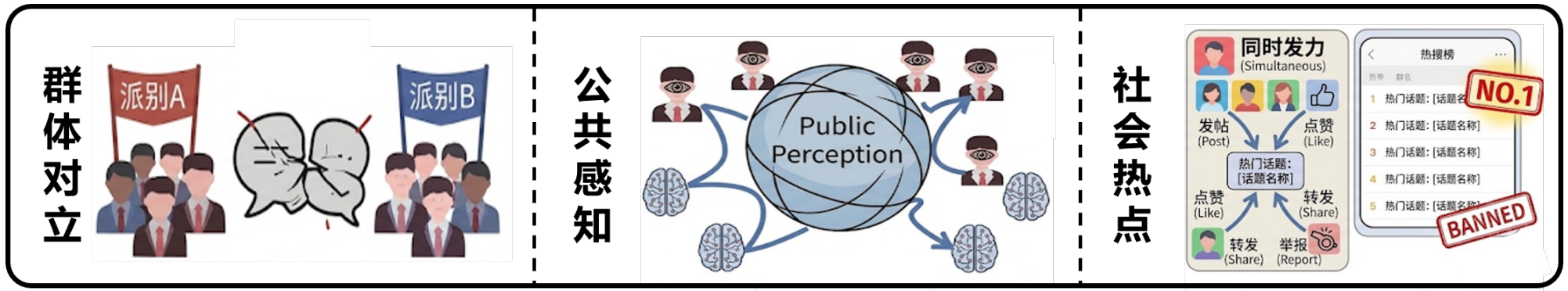
- **第二类：政策实验意义（推演评估）**

  通过仿真避免现实试错，为政策机制与治理方案提供预演空间，开展反事实推演，比较不同干预方式的效果及其潜在副作用。

研究场景

❑ 智能体社会仿真的研究场景

- **群体对立** → 关心不同派别之间的观点冲突如何形成并持续强化
- **公共感知** → 关心个体认知、群体互动与舆论环境如何共同塑造公众认知
- **社会热点** → 关心一个话题如何通过发帖、点赞、转发、举报等行为被迅速推高为热点
  又如何在平台规则干预下被限制、降温甚至下架



中国科学院大学
University of Chinese Academy of Sciences

目 录

建模范式

❑ 三种建模方式：

规则驱动方式

采用显式规则进行建模，优势在于系统逻辑清晰、演化过程可控，以及仿真结果的高度可解释

学习驱动方式

通过学习驱动（多智能体强化学习）中的个体通过与环境的交互试错，动态迭代行为策略

生成驱动方式

关注如何基于语言、具体语境与角色设定，利用大模型生成符合情境的认知、表达和行为

1. 规则驱动

❑ 规则驱动建模方式：显式规则建模，强调清晰、可控、可解释

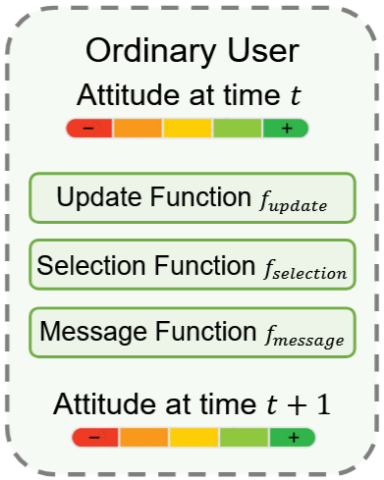
规则驱动建模方式

01 定义个体状态 *State Definition*

02 设定交互规则 *Rule Specification*

03 执行逐步更新 *Iterative Update*

04 观察交互结果 *Emergent Outcome*



$$s_{i,t} = \{a_{i,t}, N_{i,t}\}$$

$$m_{j,t} = f_{\text{message}}(a_{j,t})$$

$$J_{i,t} = f_{\text{selection}}(N_{i,t})$$

$$a_{i,t+1} = f_{\text{update}}(a_{i,t}, M_{i,t})$$

Unveiling the Truth and Facilitating Change: Towards Agent-based Large-scale Social Movement Simulation. ACL 2024

2.学习驱动

- 学习驱动建模方式：通过多智能体强化学习策略，强调适应性与动态优化

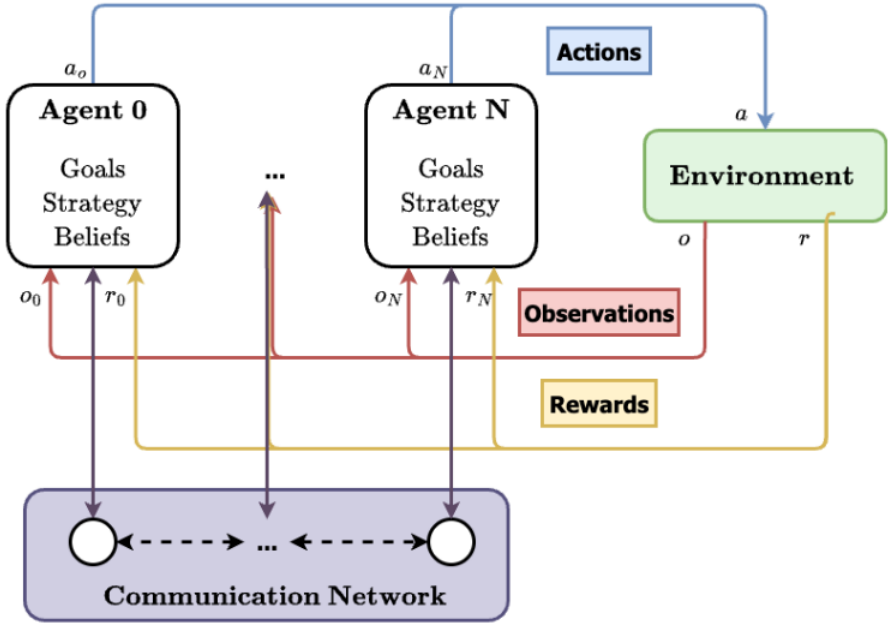
学习驱动建模过程

01 观察 *Observation*

02 行动 *Action*

03 奖励 *Reward*

04 策略更新 *Policy Update*



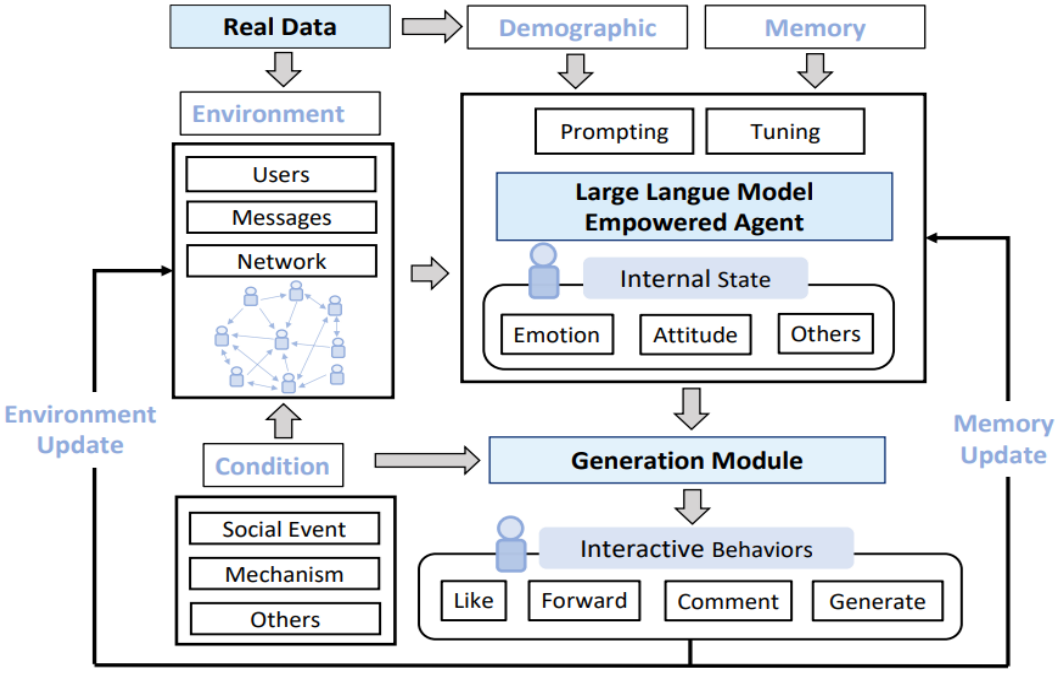
Multi-agent Reinforcement Learning: A Comprehensive Survey. 2024

3.生成驱动

- 生成驱动建模方法：利用大语言模型建模语言与社会认知，强调语义互动

核心优势：

- **生成式决策：**不依赖奖励反馈和反复试错，而是基于当前情境直接生成行为。
- **语境驱动：**结合语言信息、具体语境和角色设定生成行为。
- **过程可解释：**利用自然语言形式呈现理解、推理和响应过程。



S3: Social-network Simulation System with Large Language Model-Empowered Agents. 2023

中国科学院大学
University of Chinese Academy of Sciences

目 录

社会仿真框架

- 三个部分：个体层、交互层、环境层

个体层
用户画像 · 立场 · 情绪 · 目标 · 记忆 · 动作空间

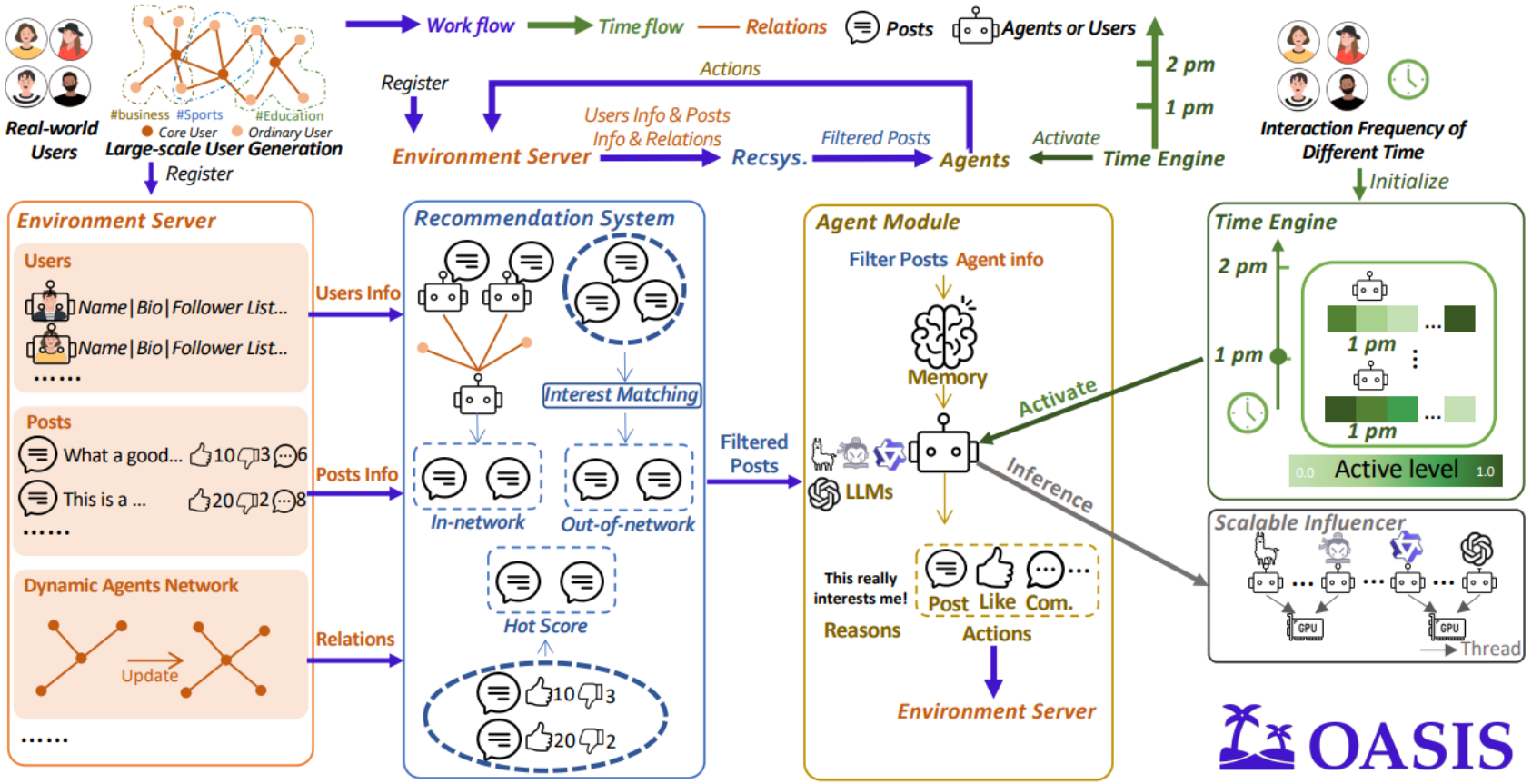
交互层
谁和谁相连 · 谁影响谁 · 谁能看到谁 · 关系强度与传播

环境层
时间线 · 推荐机制 · 热榜 · 平台规则 · 外部新闻事件

💡 **社会现象是用户行为、社交关系和平台机制共同作用的结果**

社会仿真框架：OASIS

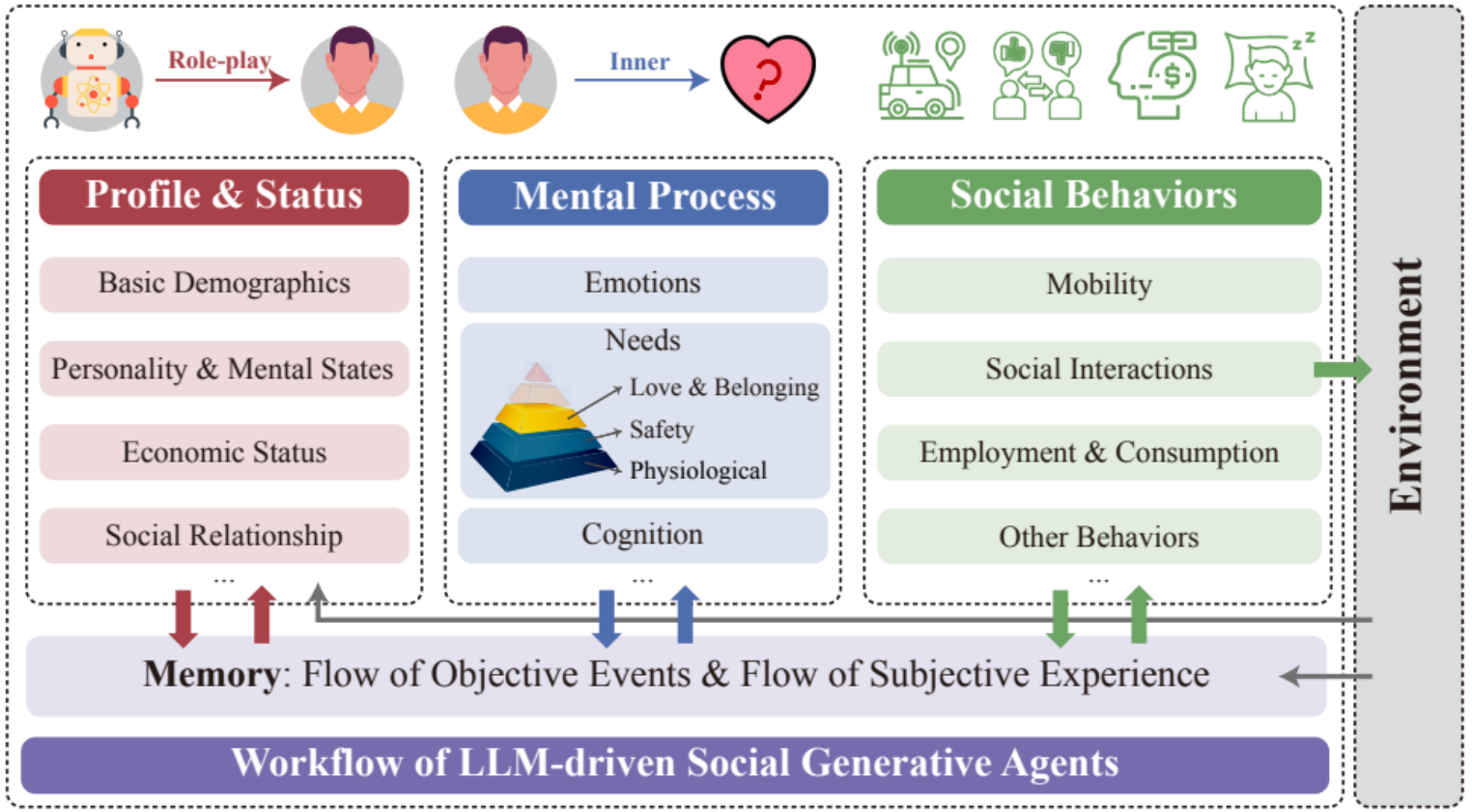
- OASIS仿真了一个百万级别社交媒体平台（多智能体系统）



Oasis: Open agent social interaction simulations with one million agents. 2024

社会仿真框架：AgentSociety

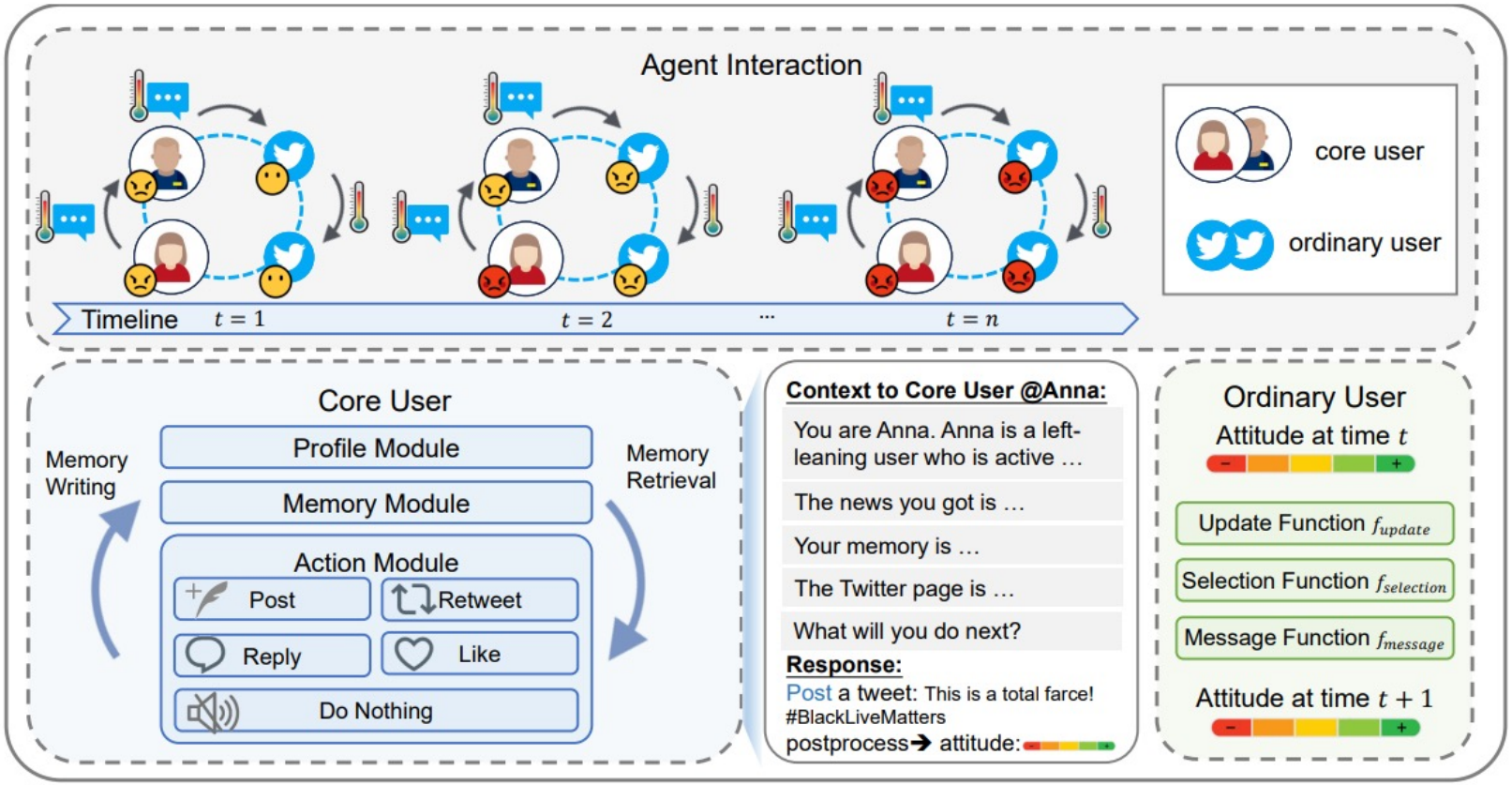
- AgentSociety仿真平台建模Agent用户的画像、记忆、动作和心理



Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society. 2025

社会仿真框架：HiSim

- HiSim仿真平台模拟仿真社会运动中公众在社交媒体上的响应



Unveiling the Truth and Facilitating Change: Towards Agent-based Large-scale Social Movement Simulation. ACL 2024

目 录

验证与评估

- 如何验证所构建智能体社会仿真框架的有效性？ 

微观层面　单个 Agent 像不像真人

- 说话是否自然流畅
- 立场是否前后一致
- 行为是否符合人设设定

宏观层面　整个系统像不像真实社会

- 群体态度是否合理演化
- 信息扩散符合真实平台规律
- 极化/从众现象能否稳定复现

单个 Agent 看起来很像人 ≠ 群体结果合理。评估社会仿真不能只看「单体对话像不像」，还要看「群体过程像不像社会」——这也是分层评估的出发点

验证与评估： UTFC

- 针对个体行为模拟，作者验证核心用户的**立场一致性、内容一致性和行为一致性**

  - **立场一致性：**评估生成内容的立场，将立场分类为**支持**、**中立**和**反对**三个类别
  - **内容一致性：**评估生成内容的类型，将类型分类为**行动号召**、**分享意见**、**引用第三方**、**提供证据**和**其他**五个类别
  - **行为一致性：**评估智能体是否采取了用户实际采取的行动（**发布/转发**）。

  ! 注意：由于在Twitter数据集中只能观察到发布和转发两种行为，因此将行动空间缩小为发布和转发。

Unveiling the Truth and Facilitating Change: Towards Agent-based Large-scale Social Movement Simulation. ACL 2024

验证与评估： UTFC

❑ 验证结论：

- **立场一致性：**由于价值观对齐，智能体很难生成一些不支持言论。（类别偏差）
- **内容一致性：**智能体生成的内容主要集中在**行动号召**和**分享意见**上，而**提供证据**表现较差，因为它们缺乏用户的线下体验。相似度反映了智能体能够复制真实用户反应的能力。
- **行为一致性：**智能体能够区分原创内容作者和转发者。

Dataset	Size	Stance	Content	Behavior
Metoo	2,214	2,166:33:15	89:78:67:1,792:188	422:1,792
Roe	3,595	3,528:29:38	48:6:72:3,362:107	233:3,362
BLM	971	934:33:4	31:10:20:887:23	84:887

Datasets	Stance			Content			Behavior	
	Acc.	F1	MAE	Acc.	F1	Sim.	Acc.	F1
Metoo	0.9679	0.3400	0.2311	0.7010	0.1988	0.8064	0.7313	0.5212
Roe	0.9430	0.3361	0.2058	0.6423	0.1957	0.8090	0.6665	0.4691
BLM	0.8991	0.3735	0.1627	0.7353	0.2218	0.8406	0.7796	0.5759

验证与评估： UTFC

- 针对群体模拟，验证在群体活动时的态度分布、平均态度随时间的变化轨迹
  - 态度分布：Bias、Diversity
  - 平均态度的时间序列：DTW、Pearson

Method	Metoo				Roe				BLM			
	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑
BC	0.0124	0.0184	2.7760	0.4831	0.0265	0.0144	5.7662	-0.7755	0.0078	0.0036	5.2289	-0.4404
Hybrid w/ BC	0.0135	0.0108	1.8440	0.7043	0.0239	0.0121	2.4611	0.3607	0.0300	0.0069	3.9254	0.1248
HK	0.0093	0.0105	2.9171	0.0262	0.0258	0.0185	7.7254	-0.7532	0.0081	0.0101	4.1204	-0.3026
Hybrid w/ HK	0.0126	0.0037	1.9136	0.6517	0.0319	0.0157	3.6752	-0.0807	0.0578	0.0093	3.7288	-0.2433
RA	0.0062	0.0055	3.1063	-0.0687	0.0237	0.0120	2.9521	0.0811	0.0039	**0.0017**	3.0441	0.2666
Hybrid w/ RA	0.0117	**0.0008**	**1.7829**	**0.7238**	0.0221	0.0104	2.3326	0.4274	0.0376	0.0070	**2.2353**	**0.6050**
SJ	0.0064	0.0192	2.2994	0.2009	0.0209	0.0106	1.2739	0.6177	0.0411	0.0072	2.7778	0.4475
Hybrid w/ SJ	0.0098	0.0119	2.2789	0.6327	0.0203	**0.0095**	1.1896	0.6598	0.0076	0.0018	2.4564	0.5167
Lorenz	0.0131	0.0198	5.3049	-0.4657	0.0352	0.0172	1.1027	0.7329	0.0895	0.0094	2.8897	0.4387
Hybrid w/ Lorenz	**0.0035**	0.0116	2.9857	0.6103	**0.0093**	0.0147	**1.0148**	**0.7576**	**0.0023**	0.0079	2.5394	0.5055

验证与评估： UTFC

❑ 实验结论

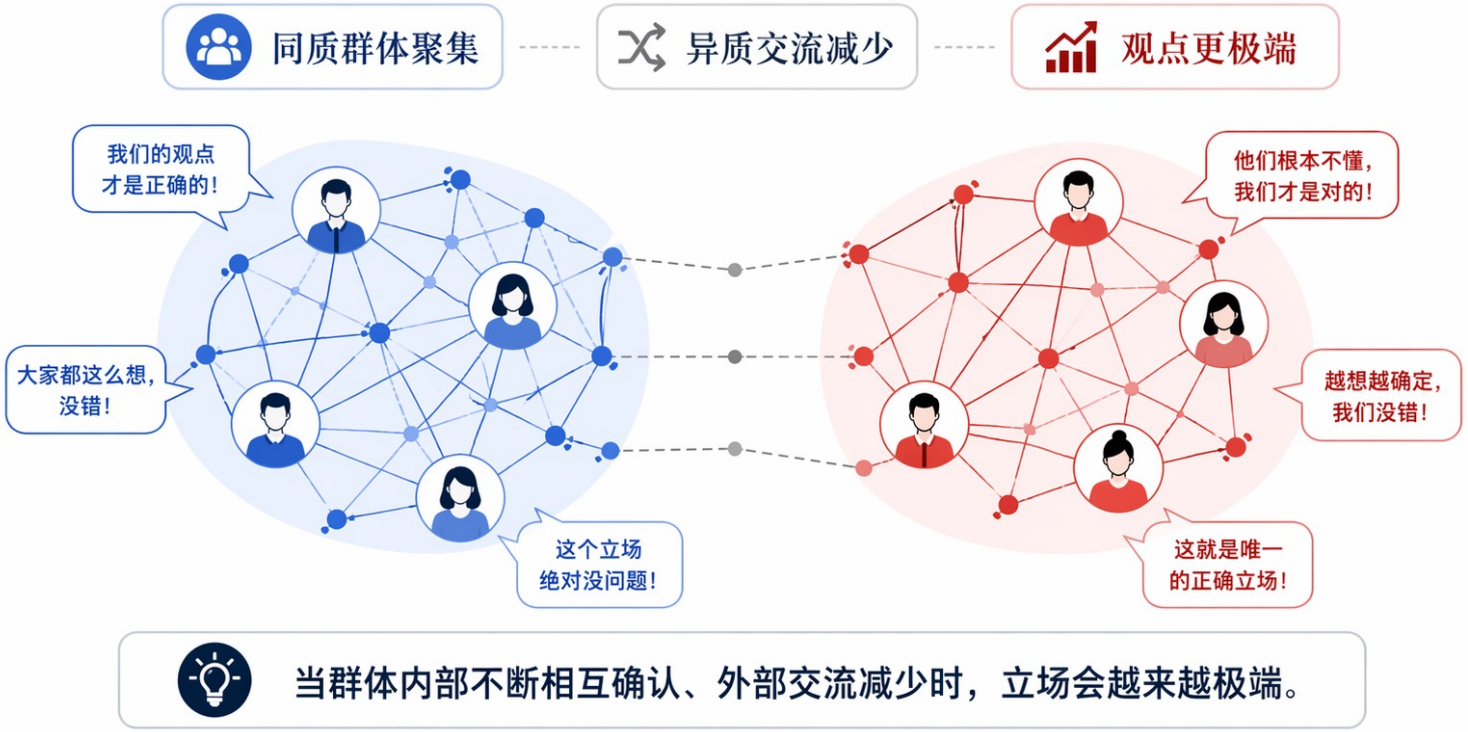
实验发现：1）发现混合模型(Hybird w/)在态度分布和时间序列指标上均表现出优于纯ABMs的性能。2）混合模型通常表现出更高的态度偏差。这是因为LLM智能体对立场中性内容模拟不足。

Method	Metoo				Roe				BLM			
	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑	$\Delta_{Bias}\downarrow$	$\Delta_{Div.}\downarrow$	DTW↓	Corr.↑
BC	0.0124	0.0184	2.7760	0.4831	0.0265	0.0144	5.7662	-0.7755	0.0078	0.0036	5.2289	-0.4404
Hybrid w/ BC	0.0135	0.0108	1.8440	0.7043	0.0239	0.0121	2.4611	0.3607	0.0300	0.0069	3.9254	0.1248
HK	0.0093	0.0105	2.9171	0.0262	0.0258	0.0185	7.7254	-0.7532	0.0081	0.0101	4.1204	-0.3026
Hybrid w/ HK	0.0126	0.0037	1.9136	0.6517	0.0319	0.0157	3.6752	-0.0807	0.0578	0.0093	3.7288	-0.2433
RA	0.0062	0.0055	3.1063	-0.0687	0.0237	0.0120	2.9521	0.0811	0.0039	**0.0017**	3.0441	0.2666
Hybrid w/ RA	0.0117	**0.0008**	**1.7829**	**0.7238**	0.0221	0.0104	2.3326	0.4274	0.0376	0.0070	**2.2353**	**0.6050**
SJ	0.0064	0.0192	2.2994	0.2009	0.0209	0.0106	1.2739	0.6177	0.0411	0.0072	2.7778	0.4475
Hybrid w/ SJ	0.0098	0.0119	2.2789	0.6327	0.0203	**0.0095**	1.1896	0.6598	0.0076	0.0018	2.4564	0.5167
Lorenz	0.0131	0.0198	5.3049	-0.4657	0.0352	0.0172	1.1027	0.7329	0.0895	0.0094	2.8897	0.4387
Hybrid w/ Lorenz	**0.0035**	0.0116	2.9857	0.6103	**0.0093**	0.0147	**1.0148**	**0.7576**	**0.0023**	0.0079	2.5394	0.5055

典型社会现象的仿真

❑ 1. 群体极化

概念：群体成员在讨论某个议题之后，整体立场会朝着讨论前原本倾向的方向变得更极端。



公共议题可能存在多种中间立场

↓ 极化影响

讨论容易被推向二元对立

↓ 研究现象

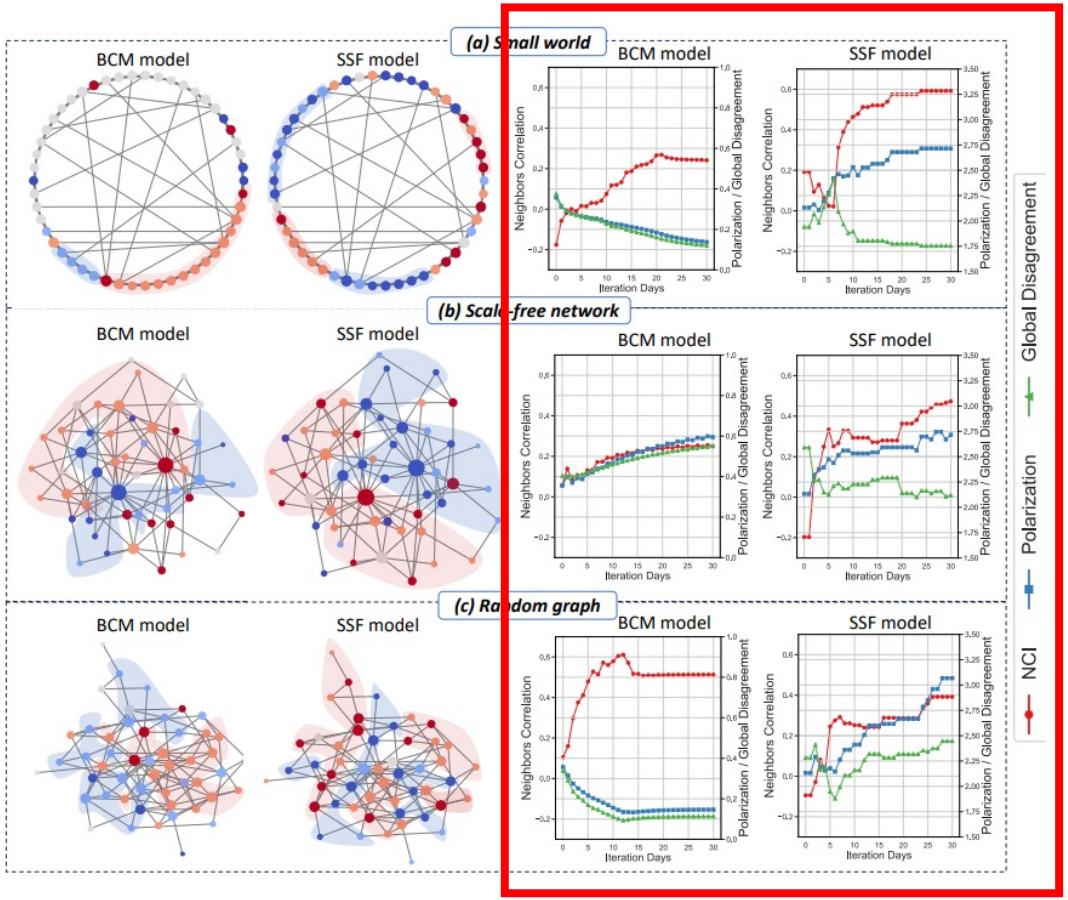
解释公共议题中的对立加剧问题

典型社会现象的仿真

❑ 1. 群体极化

在三种网络类型结构中设置50个智能体，围绕预设话题讨论30天，并以**极化指数**、**全局分歧度**和**邻居相关指数**衡量观点演化

网络类型	核心特征	现实对应
小世界网络	高聚类、短路径	紧密社群、熟人社交
无标度网络	幂律分布、存在枢纽节点	主流社交平台（存在大 V）
随机图	随机连接、无聚类 / 枢纽	无固定社交关系



Decoding Echo Chambers: LLM-Powered Simulations Revealing Polarization in Social Networks. COLING 2025

典型社会现象的仿真

❑ 2. 信息扩散

概念：信息在个体、群体或媒介节点之间，通过传播、转发、交流、模仿或推荐等机制，从少数节点逐步传播到更多节点的过程

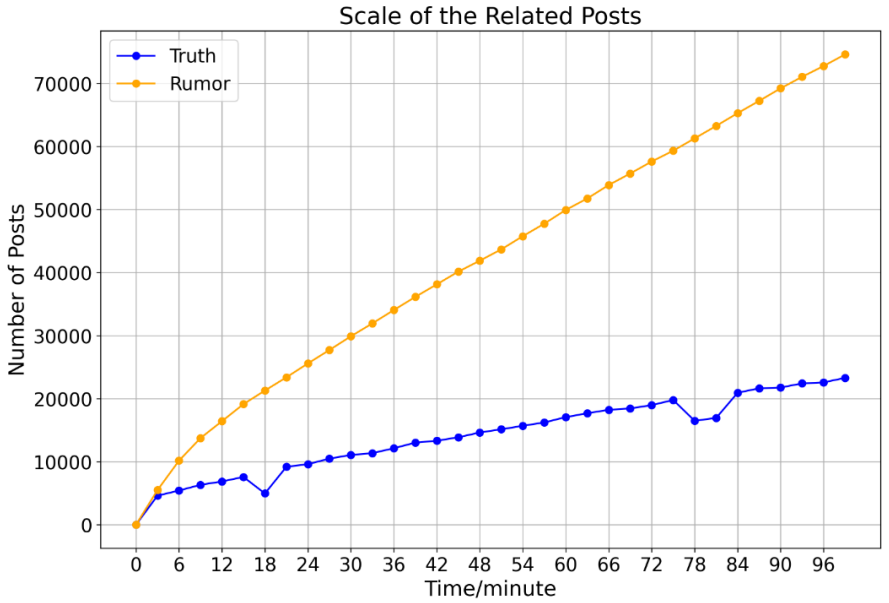
典型社会现象的仿真

❑ 2. 信息扩散

构建了一个包含 100 万用户的 Twitter 社交环境，其中包括 196 个核心用户，其余用户为普通用户。核心用户发布了 8 条消息，包括 4 对真假消息对，涉及科技、娱乐、教育和健康等领域

作者统计了真假消息相关帖子的数量变化，以分析真假消息的传播和影响力差异。

实验结论：假消息的传播规律与人类社会中类似，表现出对假消息的强倾向性。



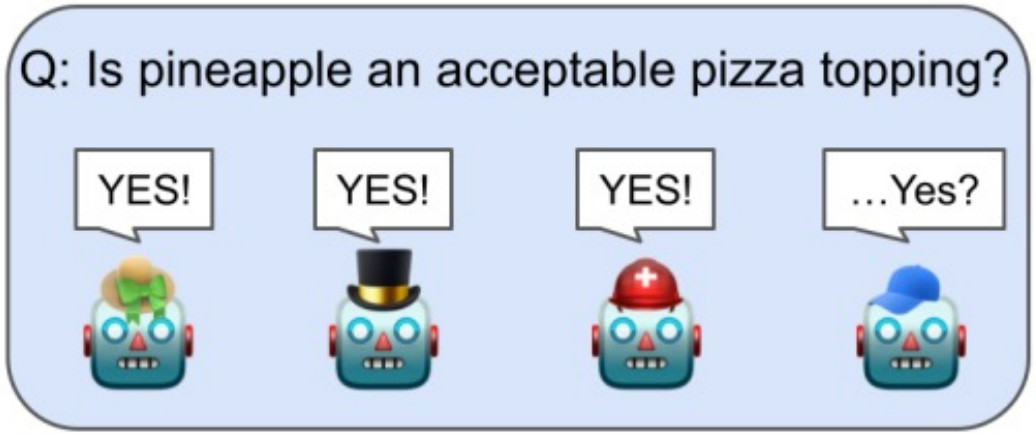
Oasis: Open agent social interaction simulations with one million agents. 2024

典型社会现象的仿真

❑ 3. 羊群效应

概念：个体在不确定情境中，基于信息依赖或社会从众压力，放弃或弱化独立判断，转而跟随他人尤其是多数人行为，从而形成群体性趋同的现象。

羊群效应削弱个体独立判断能力，会使个体在面对复杂信息时倾向于跟随多数人的选择，而不是基于事实、证据和理性分析作出判断，进而降低社会整体的信息辨别能力。



典型社会现象的仿真

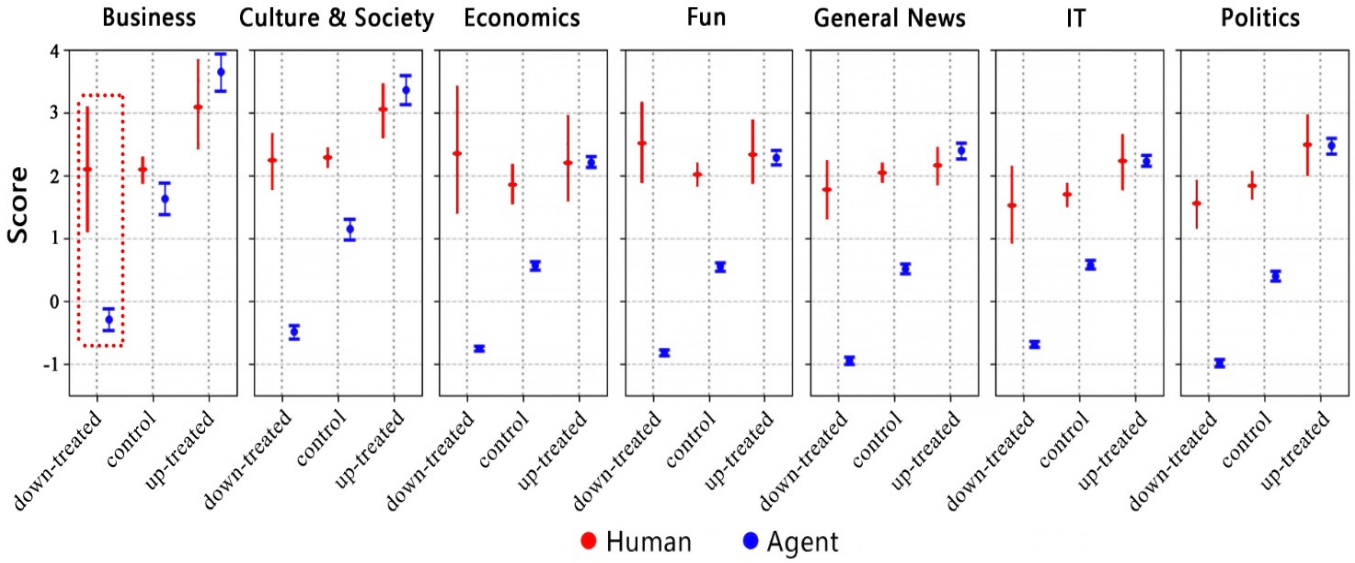
❑ 3. 羊群效应

作者从 Reddit 收集了 7 个主题的 116932 条真实评论，并用LLMs生成 3600 个用户的个人资料。
作者观察智能体交互后三组帖子的最终得分变化，用来评估初始得分对后续用户行为的影响。

实验结论

1）初始「赞」显著提高了帖子最终得分，而初始「踩」则对得分造成了抑制效果。

2）智能体表现出比人类更强的羊群效应。



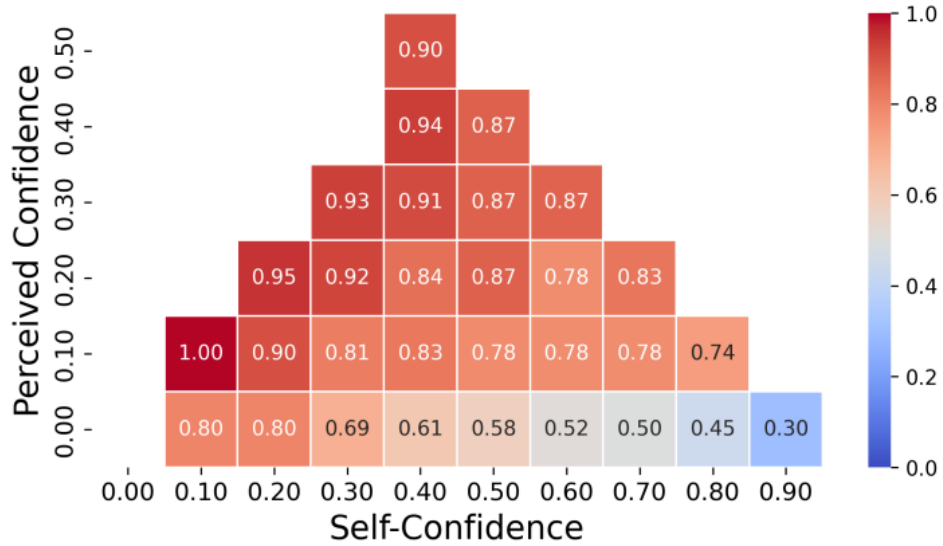
Oasis: Open agent social interaction simulations with one million agents. 2024

典型社会现象的仿真

❑ 3. 羊群效应

作者认为**信心可能会影响智能体表现出从众行为的倾向**，包括自我置信和感知置信。

首先，先让智能体 A 回答问题
其次，为智能体 B 设定一个答案。该答案可以与 A 相同，也可以是次优答案
最后，让 A 观察 B 的答案并重新作答，以判断其前后答案是否一致



翻转率：智能体在接收同伴意见后，其答案发生改变的比例。

Herd Behavior: Investigating Peer Influence in LLM-based Multi-Agent Systems. 2025

本节复习

- 多智能体系统
  - 协作拓扑：启发式拓扑、生成式拓扑
  - 记忆增强：本地记忆、共享记忆、黑板
- 智能体社会仿真
  - 建模范式：规则驱动、学习驱动、生成驱动
  - 仿真框架：个体层、交互层、环境层

参考文献

- Multi-agent collaboration mechanisms: A survey of llms. 2025.
- Autogen: Enabling next-gen LLM applications via multi-agent conversations. 2024.
- MetaGPT: Meta programming for a multi-agent collaborative framework. ICLR 2023.
- G-designer: Architecting multi-agent communication topologies via graph neural networks. 2024.
- OFA-MAS: One-for-All Multi-Agent System Topology Design based on Mixture-of-Experts Graph Generative Models. 2026.
- G-Memory: Tracing Hierarchical Memory for Multi-Agent Systems. NIPS 2025
- Oasis: Open agent social interaction simulations with one million agents. 2024

致 谢

- 胡玥、曹亚男、方芳：国科大《自然语言处理基础》
- 曹亚男、任昱冰：国科大《深度学习与自然语言处理概述》


THANKS
https://ictkc.github.io/teaching/2026spring-nlp