NL2CA：基于无监督 CriticNL2LTL 框架，从自然语言自动形式化认知决策

Zihao Deng[1,2], **Yijia Li**[1,2],**Renrui Zhang**[1,2],**Peijun Ye**[1*]
[1]中国科学院自动化研究所 [2]中国科学院大学人工智能学院{
dengzihao2025, liyijia2023, zhangrenrui2024, peijun.ye}@ia.ac.cn

arXiv:2512.18189v1 [cs.AI] 20 Dec 2025

摘要

认知计算模型为刻画人类的思考与决策提供了一种形式化且可解释的方式，但其开发仍然十分耗费人力。本文提出 NL2CA，这是一种新方法，可根据对人类经验的自然语言描述，自动形式化认知决策规则。不同于大多数采用纯手动或人工引导交互式建模的相关工作，我们的方法完全自动化，无需任何人工干预。该方法首先使用经过微调的大语言模型（LLM）将文本转换为线性时序逻辑（LTL），随后通过无监督评论树对逻辑进行优化，最后将输出转换为可与符号认知框架兼容的可执行产生式规则。基于生成的规则，我们进一步构建认知智能体，并利用真实世界的行为数据，通过认知强化学习对其进行优化。我们在两个领域验证了该方法：（1）自然语言到 LTL 的转换，其中 CriticNL2LTL 模块在专家基准和大规模基准上均取得稳定表现，且无需人在环路中提供反馈；（2）认知驾驶仿真，其中基于人类访谈自动构建的智能体已成功学习了约 70 次试验中、不同关键场景下的多样化决策模式。实验结果表明，NL2CA 能够基于非结构化文本数据，实现可扩展、可解释且与人类相符的认知建模，为自动设计符号认知智能体提供了一种新范式。

引言

ACT-R（Anderson 和 Lebiere，2014年）和 Soar（Laird，2022）等认知架构是基于规则的模型，可模拟记忆、问题求解和学习等各种认知任务。由于能够出色地模拟人类认知过程且具有较高的可解释性，此类架构已广泛应用于人类行为建模。然而，要实例化此类架构，需要投入大量专家精力，而且其设计几乎完全依赖人类专家的先验知识。

大语言模型（大语言模型），例如 GPT-4（Achiam 等，2023年）和 Deepseek（Liu 等，2024年），则展现出了模仿人类推理和行为的良好能力，这得益于在经过人类反馈训练的语料上进行后训练（Kumar 等，2025）。尽管大语言模型仍因幻觉（Rawte 等，2023年）和规模扩展的收益递减（Villalobos 等，2024年）而受到批评，但它们对人类认知过程的了解仍可用于借助认知架构自动实例化决策模型（Kirk、Wray 和 Laird，2023年）（Niu 等，2024年）（Wray、Kirk 和 Laird，2025）。

认知架构的核心组成部分是陈述性记忆和程序性记忆，它们由一组产生式规则表示，每条规则都包含一个前置条件和一个效果。认知智能体按照感知–规划–行动循环运行，动态地将环境特征与这些规则进行匹配，以确定其行动。目前已有一些研究尝试使用大语言模型自动生成产生式规则（Zhu 和 Simmons，2024年）（Kirk 等，2024年），但此前几乎所有工作都是直接询问大语言模型，在未知情境下智能体应如何行动，从而生成产生式规则。这类方法在缺乏人类先验知识和数据的情况下构建认知模型，可能导致模型与真实人类行为的契合度较低。

在这项工作中，我们提出了一种构建认知智能体的新方法：**NL2CA**（自然语言到认知智能体），利用大语言模型将人类经验的自然语言描述形式化为产生式规则。如图1所示，大语言模型用于解读对人类审议的文本描述，并将其转换为结构化表示。因此，大语言模型不仅充当规则生成器，也充当知识提取器。生成的产生式规则既基于人类经验文档，也基于大语言模型，因此有望更加符合人类的认知。

实现这一愿景要求大语言模型既具备提供正确知识的能力，也具备根据人类经验文档将产生式规则形式化的能力。鉴于前一项能力已在以往工作中得到充分验证，我们的工作主要聚焦于提升大语言模型根据人类经验文档形式化产生式规则的能力。

为提高这种产生式规则提取的准确性和可靠性，我们首先提示大语言模型将自然语言描述中的逻辑信息形式化为线性时序逻辑（LTL），这是一种功能强大的形式

*通讯作者 版权所有 © 2026年，美国人工智能促进协会（www.aaai.org）。保留所有权利。

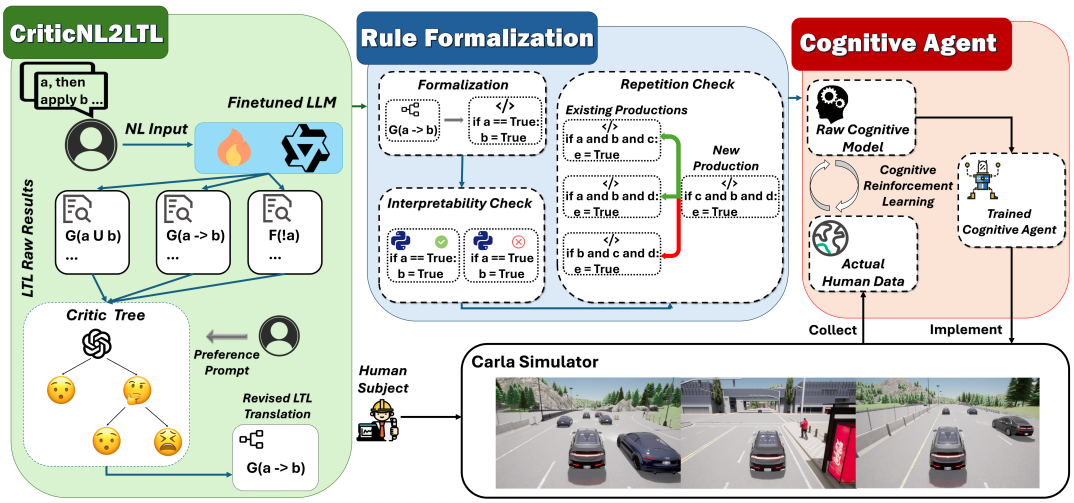


图1：NL2CA框架流程。

主义，可应用于机器人学（Wang 等，2025）、电子设计（Browne 等，1986）和自动驾驶（Maierhofer、Moosbrugger 和 Althoff，2022）等多个领域。为促进这一形式化过程，我们展示了一种新颖的框架，它结合了微调后的大语言模型与一种异构、无监督的思维树（ToT）式评论树结构，以提高 NL2LTL 翻译的准确性。这增强了所提框架的通用性，使其不仅适用于认知建模，也适用于其他采用线性时序逻辑的工程领域。

本文的主要贡献如下：1）提出了一种新颖的框架，可根据人类经验的自然语言描述，完全自动地将认知决策规则形式化；2）提出了一种更强大的自动化方法，用于处理从自然语言到线性时序逻辑形式化的转换；3）在各种驾驶模拟场景中检验了所提框架的有效性。

相关工作

认知架构设计自动化

认知架构（例如 ACT-R、Soar）已被证明能够有效地建模人类认知，但需要投入大量手动工作。为减轻这一负担，近期研究尝试利用大语言模型来实例化认知模型。STARS（Kirk 等，2024年）以及 Zhu 和 Simmons（2024年）直接向大语言模型发出查询，为特定环境状态生成产生式规则。Bajaj 等人（2023年）提出了一种知识提取机制，利用大语言模型填充符号记忆结构。

尽管这些方法能够加快认知系统的构建，但它们在实际的人类认知建模中的应用往往有限，因为这些方法要么基于任务学习，未以真实的人类决策轨迹为基础，要么仅用于知识提取。据我们所知，目前仍没有工作致力于自动化人类认知建模流程。我们的工作通过将对人类经验的自然语言描述形式化为线性时序逻辑，保留其可追溯性和类人性，并将其转换为可执行的认知建模产生式规则。

认知驾驶模型

Salvucci（2006）最早提出了结合认知架构来模仿驾驶员行为的构想。他们提出了利用特定认知架构对驾驶员行为进行建模的一般流程，并引入了适用于驾驶任务的线程化认知。

随后，研究者采用了一些方法，进一步拓展认知架构在构建虚拟驾驶员方面的应用。Ritter、Tehranchi 和 Oury（2019）阐释了 ACT-R 的运行机制，描述了驾驶员如何通过监控、决策和控制过程处理信息并生成行为。Oh、Yun 和 Myung（2024）提出了一种预测自动驾驶接管情境中心理工作负荷的方法。尽管已有诸多尝试，但由于缺乏学习能力，早期方法通常局限于单一场景，因此在自动驾驶中的应用仍然有限（Chater 等，2018）。

为解决这些问题，Qi、Hou 和 Ye（2024）提出了一种认知强化学习方法，通过人机交互使认知推理保持一致。Li 等人（2024）进一步改进了该方法，并提出了一种可解释的认知驾驶模型。据报道，该方法的表现优于标准的近端策略优化（PPO）算法，同时提供了高度可解释的模式。

然而，这些方法需要手动指定产生式规则，限制了其可扩展性。相比之下，我们的 NL2CA 框架通过直接从自然语言数据中形式化产生式规则，实现了整个流程的自动化，使其无需手动检查即可扩展到多个场景。

方法

NL2CA 框架包含三个阶段：1）使用微调后的大语言模型和无监督评论树，将自然语言文本形式化为逻辑表示；2）将

得到的时序逻辑转换为可执行的认知产生式规则；3）使用人类数据，通过认知强化学习（CRL）训练认知智能体。

在第一阶段，输入的自然语言片段被形式化为线性时序逻辑（LTL）公式。评论树模块会迭代地优化该公式，使其更贴合原句的含义。随后，优化后的 LTL 公式会使用预定义知识库中的变量进行实例化，并转换为与认知建模框架兼容的产生式规则。

在最后阶段，基于生成的产生式规则构建认知符号驱动智能体。该智能体使用认知强化学习方法，在真实人类行为数据上进行训练，并根据其输出与人类行为的匹配程度调整规则的效用。图1展示了 NL2CA 流程。

CriticNL2LTL 框架

为实现准确且无监督的自然语言到 LTL 的形式化，我们引入了 CriticNL2LTL 模块，该模块由两个主要组件构成：用于生成初始译文的微调后的大语言模型，以及用于迭代优化的异构评论树，如算法1所示。

给定一个自然语言输入，微调后的大语言模型首先生成一个原始线性时序逻辑公式。随后，该公式会被传入评论树——一种自组织的修订结构，具有可由用户配置的参数，例如最大深度和评论者数量。评论树由一个修订大语言模型和多个评审大语言模型组成：前者负责修订当前公式，后者则评估公式在逻辑上的正确性，以及它与用户定义偏好的契合度（例如是否基于指定知识库、是否符合期望格式）。当某个评审者发现不匹配时，它会向修订者的上下文中添加一条反馈消息，指示修订者修改公式并拆分出一个新的子节点；该子节点存储修订后的线性时序逻辑公式以及修订者的上下文。此过程会重复进行，直到所有评审者都认可，且无法再创建新的子节点，或者达到最大深度。我们使用异构大语言模型构建评审者，以提供多样化的修订建议，因为据报道，标准的自我精炼方法未能提高自然语言到线性时序逻辑翻译的准确性（Fang 等，2025）。

算法 1：评论树

输入：自然语言文本 T，初始译文 A_{init}
参数：评论者数量 δ，最大深度 D
输出：最终译文 A_{final}

$(A_{\text{root}}, C_{\text{root}}) \leftarrow$ Revisor(T, A_{init})
初始化根节点 $v_0 = (A_{\text{root}},\ C_{\text{root}})$
初始化节点列表 $\mathcal{N} \leftarrow \{v_0\}$
对 $d = 0$到 D **执行**
  $\mathcal{N}_{\text{new}} \leftarrow \emptyset$
  遍历每个节点 $v = (A,\ C)$ 属于 $\mathcal{N}$ **执行**
    如果 v 已有子节点**则**
      继续
    结束如果
    $results \leftarrow \emptyset$
    对于 $k = 1$ 到 δ **执行**
      $r_k \leftarrow$ 评判器($A,\ T$)
      $results$.追加(r_k)
      如果所有 $results$ 表示批准 **则**
        $A_{\text{final}} \leftarrow A$
        返回 A_{final}
      结束 if
      $C' \leftarrow C \cup r_k$
      $(A',\ C'') \leftarrow$ Revisor.修订(C')
      将新节点 $v' = (A', C'')$ 添加到 $\mathcal{N}_{\text{new}}$
    结束 for
  结束循环
  $\mathcal{N} \leftarrow \mathcal{N}_{\text{new}}$
结束循环
$A_{\text{final}} \leftarrow A_{\text{root}}$
返回 A_{final}

逻辑到代码转换

完成逻辑形式化后，将 LTL 公式转化为与**pyactr**兼容的可执行产生式规则。**pyactr**是一个开源 Python 库，用于创建和运行认知模型。我们利用大语言模型将 LTL 公式中的命题与计算变量对应起来，并确保形式化过程正确无误。此步骤还包括后处理，以确保逻辑正确性、可解释性和非冗余性。

为了解决 Zhu 和 Simmons（2024年）报告的常见问题，例如函数接口不匹配、过度约束／泛化和规则重复，我们实现了以下机制：

- **函数接口不匹配**：生成的代码会立即使用 pyactr 进行解释执行。如果执行失败，错误消息和有问题的规则会通过错误引导的修正循环返回给大语言模型进行修订。
- **过度约束 / 过度泛化**：大多数逻辑不一致会被 CriticNL2LTL 模块过滤掉。然而，含糊或相互矛盾的输入文本仍可能产生有问题的规则。这些问题会在认知强化学习阶段处理，即根据实际人类行为对规则进行评估。
- **规则重复**：为避免冗余规则，我们根据每条规则的前置条件和效果为其命名。我们将规则名称转换为 BERT 嵌入，并使用余弦相似度来衡量相似程度。每条新生成的规则都会与相似度最高的前 5 条现有规则进行比较。如果新生成的规则在语义上与某条现有规则匹配，则将其丢弃。

这些措施确保在认知智能体建模中，仅保留可解释、彼此不同且可执行的产生式规则。

认知强化学习

从自然语言中提取的产生式规则仍可能因输入存在歧义而包含不一致或错误。为提高可靠性，我们采用认知强化学习，根据每条规则与人类决策行为的吻合程度来调整其效用。

策略	模型	$Dataset_1$		$Dataset_2$	
		Acc^***(%)**	$BLEU$	Acc**(%)**	$BLEU$
交互式	GPT-3.5	75.0	0.814	63.3	0.916
	GPT4	**94.4**	**0.911**	78.3	**0.978**
	Gemini	88.9	0.884	70.4	0.960
微调	Flan-T5	8.3	0.181	90.8	0.987
	Qwen3-0.6B	22.2	0.246	**97.5**	**0.993**
端到端	DeepSeek R1	63.9	0.546	–	–
	+自我精炼	63.9	0.490	–	–
无监督	**CriticNL2LTL（我们的方法）**	80.6	0.517	90.2	0.945

表1：性能对比分析

如上所述，认知决策是指动态地将相关环境特征与产生式规则相匹配，并应用这些规则的效果。某些状态可能会触发多个产生式规则，从而产生许多推理分支。为减少歧义，将通过计算每条产生式规则 p_i 在步骤 t 的归一化效用值 u_i，来确定其激活概率：

$$p(q_t = p_i) = \frac{e^{u_i/\sigma}}{\sum_j e^{u_j/\sigma}} \tag{1}$$

其中， σ 是一个超参数。在认知智能体的训练过程中，我们主要关注其最终决策是否与人类在相同情境下的实际决策一致。因此，每当模型输出的结果与人类的实际决策一致时，其推理链就会获得奖励 R_+；否则，将被赋予负奖励 R_-。奖励将分配给先前推理路径上所有被激活的产生式规则：

$$r_i = R - \beta \Delta t \tag{2}$$

其中， Δt 表示产生式规则 p_i 被激活与收到奖励之间的时间间隔，而 β 表示衰减系数，反映了较早触发的产生式规则获得的奖励份额较少。随后，根据分配给每条产生式规则的奖励更新其效用：

$$u_i(n) = u_i(n-1) + \alpha \left[r_i(n) - u_i(n-1) \right] \tag{3}$$

其中， n 表示迭代次数， α 表示学习率。如果某条产生式规则获得的奖励超过其当前效用，其效用就会提高，从而增强它在类似条件下被激活的概率。反之，如果奖励较低，其效用就会降低。

线性时序逻辑与产生式规则

在本节中，我们将说明如何确保将线性时序逻辑形式化为产生式规则。

线性时序逻辑建立在经典命题逻辑之上，但引入时序算子后，也能够描述事件序列和时序属性。与 CTL（计算树逻辑）不同，线性时序逻辑只关注沿着各条执行路径的线性时间演化。线性时序逻辑的语法如下：

$$\varphi ::= true \mid a \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \cup \varphi_2 \tag{4}$$

其中， $true$ 表示始终满足的逻辑常量， a 表示代表系统基本陈述的原子命题， $\wedge$ 表示合取算子， $\neg$ 表示否定算子， $\bigcirc$ 是一个时序算子，表示该命题必须在下一状态或时刻为真； $\varphi_1 \cup \varphi_2$ 表示 φ_1 必须持续保持为真，直到未来某个时刻 φ_2 变为真。

另一方面，产生式规则通常采用“$IF < condition > THEN < action >$”格式来描述，表达一种简单的条件逻辑：条件满足时，就会触发相应的动作。因此，产生式规则可以看作线性时序逻辑的一个子集。线性时序逻辑公式“$G($ a $\rightarrow$ b $)$”描述了“只要 a 成立，b 也成立”的逻辑，可形式化为“IF a $THEN$b”，表达相同的逻辑。

但并非所有线性时序逻辑公式都能形式化为产生式规则。像“F”（最终）和“$\cup$”（直到）这样的运算符，在产生式规则中找不到对应的表达。这些时序算子的含义超出了产生式规则的表达能力。如果一段自然语言文本被形式化为包含这些时序算子的线性时序逻辑公式，就意味着文本涉及过于复杂的逻辑，因此不适合进一步形式化为产生式规则。因此，在我们的工作中，使用此类运算符的线性时序逻辑公式将被标记为**推理错误**，只有符合“$G($ $< propositions > \rightarrow < propositions >$”格式的线性时序逻辑公式才会被形式化为产生式规则。

将 LTL 引入为中间表示有以下几项优势：1）它能将对人类经验的非结构化自然语言描述转化为定义明确的逻辑公式，从而更容易提取并形式化相应的产生式规则。2）它能够显式表示无法用产生式规则表达的复杂逻辑，避免产生不恰当的结果。3）LTL 不依赖特定领域动作，因此可作为可复用、与任务无关的层，帮助在不同任务或场景之间实现泛化。

实验

为验证所提出的 NL2CA 框架，我们需要从两个方面考察其能力：

- CriticNL2LTL 框架能否从自然语言文本中提取准确的逻辑信息？（参见 **自然语言到线性时序逻辑实验** 部分）
- NL2CA 能否构建一个能够模仿人类行为的认知智能体？（参见 **认知驾驶仿真** 部分）

自然语言到线性时序逻辑实验

我们在两个数据集上开展自然语言到线性时序逻辑实验：$Dataset_1$，这是一个用于神经翻译方法的高难度开源基准（Cosler 等人，2023年）；以及 $Dataset_2$，这是一个规模更大的公开合成数据集（Xu、Feng 和 Miao，2024年），包含涵盖八种线性时序逻辑表达模式及多个领域的样本。

我们使用 $Dataset_2$ 中的训练样本对 Qwen3-0.6B 模型进行微调，以生成初始线性时序逻辑译文；随后，使用基于 Deepseek 模型构建的评论树对其进行优化（评论者数量为 = 2，最大深度为 = 2）。修订器使用 DeepseekR1 构建。对于评论者，每个评论者均以 50% 的概率基于 DeepseekR1 构建，以 50% 的概率基于 DeepseekV3 构建。我们使用匹配准确率（ACC）和 $BLEU$ 分数评估性能。对于 $Dataset_1$，鉴于其规模较小且难度较高，我们通过人工语义验证来评估准确率（记作 ACC^*）。为确保通用性，我们仅使用包含最少领域知识的 $minimal$ 提示（详情见附录）。在实际应用中，可以将其替换为更具领域针对性的内容，进一步提升框架性能。

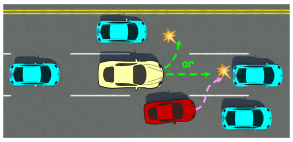
实验结果如表1所示。对于基于大语言模型的方法， $Dataset_2$ 上的性能是在从测试数据集中随机抽取的143个样本上进行测量的。

我们的方法在两个数据集上均表现出高度一致性，在 $Dataset_1$上优于纯微调方法和端到端方法，同时在 $Dataset_2$上保持了具有竞争力的结果。得益于人工监督的交互式方法在 $Dataset_1$ 上表现出色，但由于领域转移，在 $Dataset_2$ 上性能显著下降。值得注意的是，传统的自我精炼方法（Madaan 等人，2023年）未能提升端到端基线的性能，这凸显了我们所提出的评论树结构的有效性。

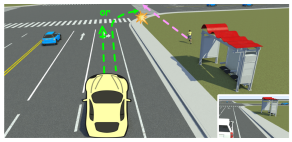
认知驾驶仿真

为验证NL2CA从人类经验文本中构建恰当认知模型的能力，我们开展了多项认知驾驶仿真实验。我们使用CARLAUE4构建了3个具有代表性的驾驶场景，包括高速公路、信号交叉口和变道干扰，如图2所示：

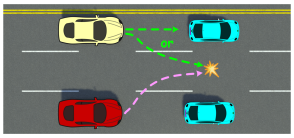
•高速公路：在一个三车道高速公路场景中，自车前方有一组车辆，占据了全部三条车道。在行驶过程中，位于自车右侧的一辆NPC车辆会尝试并入自车所在车道。



(a) 高速公路



(b) 信号交叉口



(c) 变道干扰-ence

图2：驾驶仿真场景示例。

参数	描述	值
α	学习率	2e-4
β	衰减系数	0.01
σ	效用系数	$\sqrt{2}$
u_0	生产的初始效用	0
R_+	正奖励	10
R_-	负奖励	0

表2：实验参数

• 信号交叉口：在一个信号交叉口场景中，自车需要右转通过交叉口。在此过程中，一名行人非玩家角色会尝试横穿道路，阻碍自车通行。

•变道干扰：在三车道高速公路场景中，自车行驶在最左侧车道，前方有一辆行驶缓慢的NPC车辆。如果自车试图超车，最右侧车道中的一辆NPC车辆将尝试向左变道。

这些场景经过精心设计，用于呈现城市和高速公路驾驶中的边缘情况。在这些情况下，决策对细微的环境变化和驾驶员意图高度敏感。通过聚焦于这些情况，我们旨在评估通过NL2CA构建的认知智能体是否能够捕捉一般性的人类推理模式。

我们还邀请了34名不同的驾驶员，并使用虚拟现实设备，在每个虚拟场景中总共进行了约70次试验。要求驾驶员从预先设定的起点驾驶至目的地，同时遵守交通规则并避免任何潜在碰撞。在驾驶过程中，我们记录他们的控制输入（例如油门、转向角等）以及环境状态信息，作为用于认知强化学习的实际人类数据。驾驶结束后，我们会访谈每位驾驶员，了解他们所做的驾驶决策及相应原因，从而收集一组用于逻辑信息提取的人类驾驶经验文本。

我们采用与自然语言到线性时序逻辑实验相同的参数，将自然语言文本中的产生式规则形式化。每段访谈记录首先输入 CriticNL2LTL，以获得中间的线性时序逻辑表示。所得线性时序逻辑

场景	模型	成功率 (%) ↑	RSR (%) ↑	ACR (%) ↑	AST (秒)	ALD (厘米)↓
高速公路	人类数据	53.73	–	83.98	27.79	27.86
	手动	25.00	**98.67**	**89.06**	26.29	25.78
	M_{lit}	**58.70**	87.62	82.02	26.41	**9.43**
	M_{sup}	12.00	93.07	77.58	22.29	10.70
信号化交叉路口	人类数据	84.13	–	99.56	16.08	31.93
	手动	75.00	**100.00**	92.41	10.19	22.99
	M_{lit}	**85.71**	99.46	**99.84**	7.25	**20.15**
	M_{sup}	76.60	99.62	99.77	8.83	22.26
变道干扰	人类数据	85.92	–	97.59	28.71	39.88
	手动	**95.92**	99.73	**97.81**	22.42	**25.54**
	M_{lit}	91.30	**100**	89.73	30.87	35.05
	M_{sup}	63.82	**100**	89.70	19.19	21.65

表3：对比驾驶仿真分析

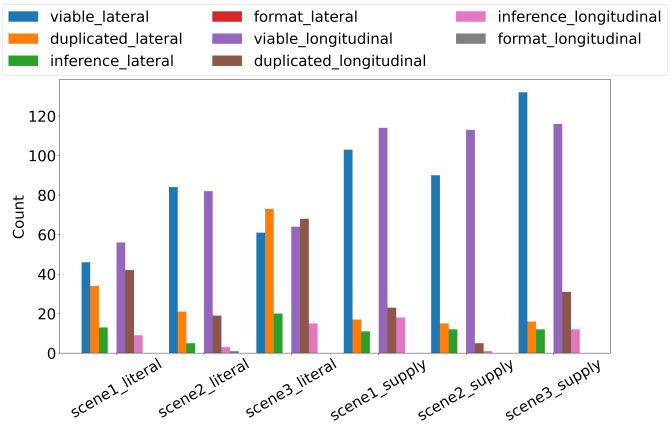


图3：逻辑提取与形式化结果。

公式随后输入另一个大语言模型，并转换为包含已落地变量的可执行产生式规则。

遵循标准的认知驾驶流程，我们将驾驶决策过程划分为纵向决策和横向决策。对于每个中间线性时序逻辑表示，将其形式化为两条产生式规则，分别描述纵向和横向决策过程。此外，由于并非每个文本片段都包含对这两种决策过程的完整描述，当大语言模型判断某个线性时序逻辑公式中的决策相关信息不完整或缺失时，我们也会提示其输出“$pass$”。

在提取逻辑信息并将其形式化为产生式规则后，我们还分析了本方法在三个驾驶场景的访谈记录上的结果。为明确说明结果，我们定义了以下四种形式化结果类型，并记录它们在不同场景数据集中的出现频率：

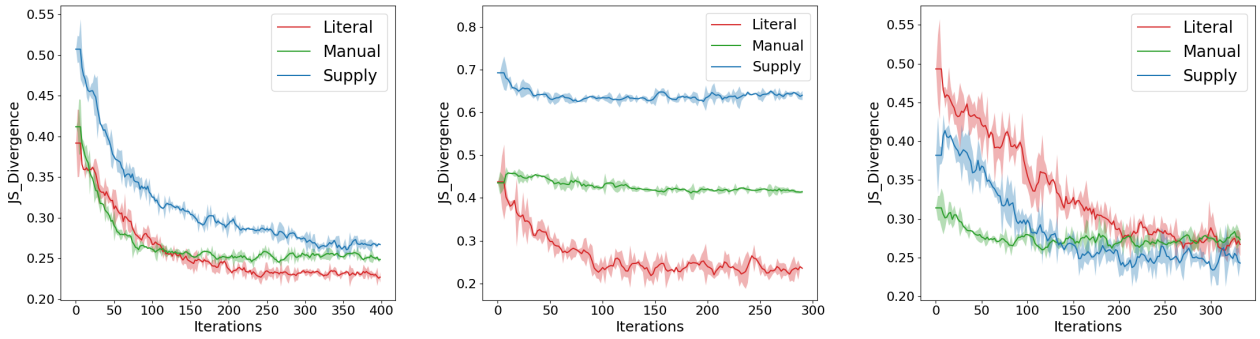
- **格式不匹配**：形式化产生式规则无法被推理引擎解析。
- **内容重复**：生成的产生式规则与已有的产生式规则内容相同。
- **推理错误**：生成的产生式规则包含未定义变量或运算符。
- **可行**：生成的产生式规则中未出现上述三种错误。

如图3所示，大多数形式化生产规则都是可行的。后缀“供给”和“字面”指的是两种不同的生成方法，稍后将对此进行说明。

使用生成的产生式规则，我们为3个场景构建认知驾驶智能体，并利用收集到的人类驾驶数据，通过认知强化学习流程对其进行训练。训练的关键在于更新驾驶模型中每条产生式规则的效用。为便于训练，模型的输入是经过精确标注的场景元素，而非图像和点云等原始感知数据；模型的输出则是方向盘、油门和刹车的控制信号。当模型预测的动作与人类的选择一致时，给予正奖励 R_+；否则，给予负奖励 R_-。更详细的参数设置见表2。

为了将我们的方法与当前直接询问大语言模型在未知情况下应如何反应的方法区分开来，我们还通过提示大语言模型主动推断访谈记录中可能缺失的信息并予以补充，收集了一组产生式规则。此举旨在探索进一步利用大语言模型的先验知识来发现隐藏逻辑，是否能够改进自动化认知建模。我们将这种方法记为 M_{sup}，而将直接依据访谈记录按字面形式化产生式规则的原始方法记为 M_{lito}。使用这两种方法构建的认知驾驶模型，均通过相同的认知强化学习流程，在实际人类数据集上进行训练。

优化认知驾驶模型后，我们根据从人类受试者在三个驾驶场景中收集的测试结果，评估各模型的性能。我们测量了关键驾驶性能指标，包括 RSR（推理成功率）、ACR（平均完成率）、AST（平均模拟时间）和 ALD（平均车道偏移量）；报告的指标为 50 次测试运行的平均结果。为获得更多对比结果，我们还进行了测试实验，采用



(a) 高速公路 (b) 信号交叉口 (c) 变道干扰

图 4：认知模型的 JS 散度曲线。

手动设计的可解释认知驾驶基线模型（Li 等，2024年），结果如表3所示。

实验结果如 M_{lit} 所示，其表现可与融入专家先验知识的手动构建基线相媲美，但 M_{sup} 相较于这两种方法并无优势。手动基线得益于专家的精心设计，推理完整性较高，因此在 RSR 指标上具有明显优势。在所有三个测试场景中，其平均 RSR 达到 99.47%，平均 ACR 最高，为 93.09%。相比之下， M_{lit}的平均 RSR 为 95.69%，平均 ACR 为 90.53%。然而，在成功率（SR）和 ALD 方面，手动基线与 M_{lit} 相比仍存在显著差距。具体而言， M_{lit} 取得了最高的平均 SR，达 78.57%，平均 ALD 为 $21.54cm$；而手动基线的相应得分分别为 65.30% 和 $24.77cm$。

为了全面评估认知模型与人类行为的对齐程度，我们进一步计算了各认知模型生成的决策分布 P与实际人类行为中观察到的相应分布 Q 之间的 Jensen-Shannon（JS）散度。JS 散度的定义如下:

$$JS(P\|Q) = \frac{1}{2} KL(P\|M) + \frac{1}{2} KL(Q\|M) \quad (5)$$

其中， KL表示 Kullback-Leibler（KL）散度，$M = (P + Q)/2$。具体而言，我们选取人类决策样本数最多的前 10 个环境输入。对于这些输入中的每一个，都运行认知决策模型 n 次，以生成相应的决策分布。n表示具有特定环境输入的样本数。在认知模型的训练过程中，我们持续跟踪相应的 JS 散度。结果如图 4 所示。

如结果所示， M_{lit} 与人类行为的对齐效果最佳，平均 JS 散度最低，同时也具备不错的学习能力。相比之下，在信号交叉口和变道干扰场景中，随着训练迭代次数增加，手动基线的 JS 散度变化有限。这表明其性能更多依赖专家的主观设计，而非从人类驾驶行为中学习。

M_{sup} 表现出一定的学习能力。然而，其近似人类决策分布的能力仍相对有限。这可能是因为其产生式规则的前置条件由大语言模型扩展，从而可能导致过度约束。这类约束会减少认知模型的决策分支数量（即针对给定环境输入的决策选项减少），限制模型拟合人类决策分布的能力。

讨论

我们提出了一种无需人工干预、从自然语言文本中自动构建认知决策规则的方法，这既有益于人工智能，也有益于认知计算。对于人工智能而言，该方法提供了一种将非结构化文本自动形式化为形式逻辑的可行途径，为不同于通过“黑箱”网络推理所有相关文本的另一种范式，从而产生可追溯且易于理解的决策，避免大语言模型因其概率性质而产生的幻觉（Wang 等，2023）（Zhang 等，2023）。对于认知计算而言，该方法开辟了一条基于人类自然语言、以低人工成本进行自动建模的路径，有助于研究人类在各种人机交互任务中的表现，例如协同驾驶或其他关键交互系统。

结论

本文提出了一个框架，用于从人类经验文档中自动形式化认知决策，并构建用于模仿人类行为的认知模型。我们展示了该框架如何从文本中形式化产生式规则，并将其应用于驾驶仿真。该工作为基于自然语言数据的自动认知建模提供了全新的见解。

致谢

本工作部分得到国家自然科学基金资助，项目编号为 62476270和T2192933。

参考文献

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; 等. 2023年. GPT-4技术报告. *arXiv*预印本 *arXiv:2303.08774*。Anderson, J. R.; 和 Lebiere, C. J. 2014年. 思维的原子组成。心理学出版社。Bajaj, G.; Pearce, K.; Kennedy, S.; Larue, O.; Hough, A.; King, J.; Myers, C.; 和 Parthasarathy, S. 2023年. 为认知架构生成组块。载于*AAAI*研讨会系列论文集，第2卷，246–252。Browne；Clarke；Dill；和 Mishra. 1986年. 使用时序逻辑自动验证时序电路。*IEEE*计算机汇刊，100(12)：1035–1044。Chater, N.; Misyak, J.; Watson, D.; Griffiths, N.; 和 Mouz- akitis, A. 2018年. 探讨交通问题：认知科学能否助力自动驾驶汽车成为现实？认知科学趋势，22(2)：93–95。Cosler, M.; Hahn, C.; Mendoza, D.; Schmitt, F.; 和 Trippel, C. 2023年. nl2spec：利用大语言模型交互式地将非结构化自然语言转换为时序逻辑。载于计算机辅助验证国际会议，383–396。施普林格。Fang, Y.; Jin, Z.; An, J.; Chen, H.; Chen, X.; 和 Zhan, N. 2025. 利用多样化外部知识增强自然语言到信号时序逻辑的转换。*arXiv*预印本 *arXiv:2505.20658*。Kirk, J. R.; Wray, R. E.; 和 Laird, J. E. 2023年. 将语言模型用作认知智能体的知识来源。载于*AAAI*研讨会系列论文集，第2卷，286–294。Kirk, J. R.; Wray, R. E.; Lindes, P.; 和 Laird, J. E. 2024年. 通过智能体分析改进大语言模型的知识提取，以实现任务学习。载于*AAAI*人工智能会议论文集，第38卷，18390–18398。Kumar, K.; Ashraf, T.; Thawakar, O.; Anwer, R. M.; Cholakkal, H.; Shah, M.; Yang, M.-H.; Torr, P. H.; Khan, F. S.; 和 Khan, S. 2025. 大语言模型后训练：深入探究推理型大语言模型。*arXiv*预印本 *arXiv:2502.21321*。Laird, J. E. 2022. Soar简介。*arXiv*预印本 *arXiv:2205.03854*。Li, Y.; Qi, H.; Zhu, F.; Lv, Y.; 和 Ye, P. 2024年. 基于认知强化学习的可解释自动驾驶模型。载于*2024 IEEE*智能车辆研讨会（*IV*），515–520。IEEE。Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; 等. 2024年. DeepSeek-V3技术报告。*arXiv*预印本 *arXiv:2412.19437*。Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; 等. 2023年. 自我精炼：利用自我反馈进行迭代精炼。神经信息处理系统进展，36：46534–46594。Maierhofer, S.; Moosbrugger, P.; and Althoff, M. 2022. 以时序逻辑形式化交叉路口交通规则。载于*2022 IEEE*智能汽车研讨会 *(IV)*，1135–1144。IEEE。Niu, Q.; Liu, J.; Bi, Z.; Feng, P.; Peng, B.; Chen, K.; Li, M.; Yan, L. K.; Zhang, Y.; Yin, C. H.; 等. 2024年. 大语言模型与认知科学：相似之处、差异与挑战的全面综述。*arXiv*预印本 *arXiv:2409.02387*。Oh, H.; Yun, Y.; and Myung, R. 2024年. 自动驾驶中的驾驶员行为与接管安全心理工作负荷：ACT-R预测建模方法。《交通伤害预防》，25(3)：381–389。Qi, H.; Hou, E.; and Ye, P. 2024年. 认知强化学习：面向虚拟驾驶员的可解释决策。*IEEE*射频识别期刊，8：627–631。Rawte, V.; Chakraborty, S.; Pathak, A.; Sarkar, A.; Tonmoy, S. I.; Chadha, A.; Sheth, A.; and Das, A. 2023年. 大语言模型中幻觉现象令人担忧地出现——详尽的定义、量化与补救措施。计算语言学协会。Ritter, F. E.; Tehranchi, F.; and Oury, J. D. 2019. ACT-R：用于认知建模的认知架构。*Wiley*跨学科评论：认知科学，10(3)：e1488。Salvucci, D. D. 2006. 在认知架构中建模驾驶员行为。《人为因素》，48(2)：362–380。Villalobos, P.; Ho, A.; Sevilla, J.; Besiroglu, T.; Heim, L.; and Hobbhahn, M. 2024年. 观点：我们会耗尽数据吗？基于人类生成数据的大语言模型扩展限制。载于第四十一届国际机器学习会议。Wang, J.; Sundarsingh, D. S.; Deshmukh, J. V.; and Kantaros, Y. 2025. ConformalNL2LTL：将自然语言指令转换为具有保形正确性保证的时序逻辑公式。*arXiv*预印本 *arXiv:2504.21022*。Wang, W.; Xie, J.; Hu, C.; Zou, H.; Fan, J.; Tong, W.; Wen, Y.; Wu, S.; Deng, H.; Li, Z.; 等. 2023年. Drivemlm：通过行为规划状态对齐用于自动驾驶的多模态大型语言模型。*arXiv*预印本 *arXiv:2312.09245*。Wray, R. E.; Kirk, J. R.; and Laird, J. E. 2025. 将认知设计模式应用于通用大语言模型智能体。*arXiv*预印本 *arXiv:2505.07087*。Xu, Y.; Feng, J.; and Miao, W. 2024年. 从失败中学习：使用大语言模型将自然语言需求转换为线性时序逻辑。载于*2024*年 *IEEE* 第 *24* 届软件质量、可靠性与安全国际会议 *(QRS)*，204–215。IEEE。Zhang, H.; Song, H.; Li, S.; Zhou, M.; and Song, D. 2023年. 基于Transformer预训练语言模型的可控文本生成综述。*ACM*计算调查，56(3)：1–37。Zhu, F.; and Simmons, R. 2024年. 使用大语言模型引导认知智能体。载于*AAAI*人工智能会议论文集，第38卷，655–663。