arXiv:2603.28426v1 [cs.CL] 30 Mar 2026

从自然语言到信号时序逻辑的结构歧义感知翻译

Kosei Fushimi、Kazunobu Serizawa、Junya Ikemoto、Kazumune Hashimoto

摘要——**信号时序逻辑（STL）广泛用于描述信息物理系统中的定时任务和安全关键任务，但非专家用户很难直接编写信号时序逻辑公式。自然语言（NL）提供了一种便捷的交互方式，但其固有的结构歧义使得将其逐一对应地翻译为信号时序逻辑并不可靠。本文提出一种**保留歧义的**方法，将自然语言任务描述转换为信号时序逻辑候选公式。其核心思想是在句法分析阶段保留多个合理的句法分析，而不是强行确定唯一解释。为此，我们开发了一个基于组合范畴语法（CCG）的三阶段处理流程：保留歧义的 n最佳句法分析、面向信号时序逻辑的模板式语义组合，以及结合分数聚合的规范化。所提出的方法会输出一组去重后的信号时序逻辑候选项及其可信度分数，从而明确表示对含糊指令的多种可能形式化解释。与现有仅输出一个结果的自然语言到逻辑翻译方法不同，所提出的方法旨在保留附着歧义和辖域歧义。针对代表性任务描述的案例研究表明，对于确有歧义的输入，该方法会生成多个信号时序逻辑候选项；而对于无歧义或规范等价的推导，则会归并为单个信号时序逻辑公式。**

一、引言

人们越来越期望自主机器人（如移动机器人、无人机和自动驾驶车辆）能够在满足时序和安全要求的同时执行复杂任务。形式化规约语言为高层任务描述与底层控制之间提供了严谨的接口。信号时序逻辑（STL）能够表达实值信号上的有界时间可达性、安全性、顺序要求和持续性要求，因此已成为连续状态系统中的标准选择 [1], [2]。此外，STL 还具有定量鲁棒性语义，可衡量一条轨迹满足某项规范的程度，因此使 STL 在运动规划、模型预测控制和控制器综合中颇具吸引力[3], [4], [5]。

尽管具有这些优势，对于非专业用户而言，直接编写信号时序逻辑公式仍然很困难。即使任务描述看似简单，也必须精准地选择原子命题、时序算子、时间区间和逻辑辖域。自然语言（NL）为任务规约提供了一个颇具吸引力的前端。近期关于机器人语言落地和形式化任务规约的研究，探索了将用户指令转换或落地为时序逻辑公式，以用于后续规划与执行 [6], [7], [8], [9]。关于机器人语言落地的更全面概述，请参见 [10]。然而，自然语言本质上具有歧义。特别是，像“避开 A 时”这样的修饰语或安全条件，既可能适用于单个子任务，也可能适用于整个任务，从而使同一条指令产生多种合理的形式化解释。如果忽略这种歧义，最终得到的时序逻辑公式可能无法表达预期的控制要求。

现有的自然语言到时序逻辑/信号时序逻辑方法主要将翻译视为单一最优预测问题。这一趋势体现在基于神经模型或大语言模型的学习型翻译器 [11], [12], [13], [14], 以及近期的信号时序逻辑翻译框架 [15]中。当预期的形式化表示唯一时，这种表述方式很有效；但对于诸如“在 10秒内到达 B 或到达 C，同时避开 A”这样的指令，由于修饰语的作用域并不唯一确定，这种方式便会产生问题。交互式方法在一定程度上解决了这一问题：Gavran 等人通过 *nl2spec*从自然语言和示范中综合线性时序逻辑规范；而另有方法利用大语言模型，通过可编辑的子翻译来检测并消解歧义 [16],[17]。在信号时序逻辑场景中，Mohammadinejad 等人考虑了基于语言和示范的交互式学习 [18]。然而，仍有一个挑战：当一个句子存在多种在结构上有效的解读时，如何获得一组规模小且易于解释的合理形式化表示。在这种情况下，理想的备选方案应当在差异上对应明确的语言选择，例如修饰语附着或作用域。否则，就难以区分输入句子本身的真正歧义与翻译过程引入的不确定性。

本文针对这一不足，提出了一种具备结构歧义感知能力的自然语言到信号时序逻辑的转换方法。其核心思想是在句法分析阶段保留歧义，而不是消除歧义。具体而言，我们将基于组合范畴语法（CCG）的语义解析（例如，参见 [19]）应用于自然语言到信号时序逻辑的转换：保留多个得分较高的解析树，利用任务专用词汇语义模板组合面向信号时序逻辑的中间语义，并将等价推导规范化为一个带分数的信号时序逻辑候选项集合。因此，所提方法的输出不是单个信号时序逻辑公式，而是一个候选集，用于明确表示对含歧义指令的多种合理形式化表示。

本文有两项主要技术贡献。首先，我们提出一种保留歧义的自然语言到信号时序逻辑翻译形式化方案，其输出是带分数的信号时序逻辑公式候选集，而非单个公式。其次，

作者就职于日本吹田市大阪大学研究生院工学研究科（Kazumune Hashimoto 为通讯作者；电子邮箱：hashimoto@eei.eng.osaka-u.ac.jp）。

我们开发了一套符号化流水线，通过结合 n最佳CCG解析、面向信号时序逻辑的模板化语义组合，以及结合分数聚合的规范化，实现这一形式化方案。具有代表性的案例研究表明，对于确有歧义的输入，所提出的方法会生成多个候选项；而对于无歧义输入或规范化后等价的推导，则会将其归并为单个信号时序逻辑公式。

本文其余部分安排如下。第二节回顾信号时序逻辑的预备知识。第三节介绍问题设定及引出本文研究动机的歧义模式。第四节介绍所提出的保留歧义的自然语言到信号时序逻辑翻译流程。第五节报告对代表性句子的数值评估结果。最后，第六节总结全文，并讨论其局限性和未来工作。

II. 预备知识

A. 信号时序逻辑

信号时序逻辑（STL）是一种用于描述实值信号时序属性的形式语言。由于本文主要关注将自然语言控制描述翻译为 STL 时保留歧义，因此我们仅简要回顾下文将用到的 STL 语法和鲁棒性语义。有关 STL 的标准定义和符号说明，请参阅 [1], [2]。STL 公式φ 由原子命题、逻辑算子和时序算子递归定义如下：

$$\begin{aligned}\varphi ::= \top \mid \mu \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \\ \mid \mathbf{F}_I\varphi \mid \mathbf{G}_I\varphi \mid \varphi_1 \mathbf{U}_I \varphi_2,\end{aligned} \tag{1}$$

其中， $\top$ 表示布尔常量真， $\neg$、 $\wedge$和 $\vee$分别表示否定、合取和析取。原子命题 $\mu : \mathbb{R}^n \to \mathbb{B}$是关于当前信号值的布尔值谓词。例如，给定一个实值函数 $h : \mathbb{R}^n \to \mathbb{R}$, 定义

$$\mu(z) = \text{True} \iff h(z) > 0,\ z \in \mathbb{R}^n. \tag{2}$$

算子 $\mathbf{F}_I$、$\mathbf{G}_I$ 和 $\mathbf{U}_I$ 分别表示离散时间区间上的时序算子最终、始终和直到

$$I = [a, b] = \{\tau \in \mathbb{Z}_{\geq 0} \mid a \leq \tau \leq b\},\ a, b \in \mathbb{Z}_{\geq 0}. \tag{3}$$

用自然语言来说，$\mathbf{F}_I\varphi$ 表示“φ 在 I中的某个时刻成立”，$\mathbf{G}_I\varphi$ 表示“φ 在 I中的每个时刻都成立”，而$\varphi_1\mathbf{U}_I\varphi_2$ 表示“φ_2 在 I中的某个时刻成立，而 φ_1则持续成立，直到该时刻。”此外，最终 和始终算子可以用直到 算子表示为

$$\mathbf{F}_I\varphi := \top \mathbf{U}_I\, \varphi,\ \mathbf{G}_I\varphi := \neg\mathbf{F}_I\neg\varphi. \tag{4}$$

信号时序逻辑公式的满足程度可以通过其鲁棒性量化。鲁棒性为正表示满足，而为负则表示违反。对于轨迹 $x = x_{0:T} = (x_0,\ x_1,\ \ldots\ldots,\ x_T)$ 和时间 t, 鲁棒性 $\rho^{\varphi}(x,\ t)$ 递归定义如下：

$$\begin{aligned}
&\rho^{\mu}(x,t) = h(x_t),\\
&\rho^{\neg\mu}(x,t) = -h(x_t),\\
&\rho^{\varphi_1\wedge\varphi_2}(x,t) = \min(\rho^{\varphi_1}(x,t), \rho^{\varphi_2}(x,t)),\\
&\rho^{\varphi_1\vee\varphi_2}(x,t) = \max(\rho^{\varphi_1}(x,t), \rho^{\varphi_2}(x,t)),\\
&\rho^{\boldsymbol{F}_I\varphi}(x,t) = \max_{t_1\in t+I} \rho^{\varphi}(x,t_1),\\
&\rho^{\boldsymbol{G}_I\varphi}(x,t) = \min_{t_1\in t+I} \rho^{\varphi}(x,t_1),\\
&\rho^{\varphi_1\boldsymbol{U}_I\varphi_2}(x,t) = \max_{t_1\in t+I}\Big(\min(\rho^{\varphi_2}(x,t_1),\ \min_{t_2\in[t,t_1]} \rho^{\varphi_1}(x,t_2))\Big).
\end{aligned}$$

with $t + I := \{\, t + \tau \mid \tau \in I \,\}$.

III. 问题设定

A. 目标描述与结构歧义

将自然语言（NL）控制描述转换为信号时序逻辑很有吸引力，因为用户很容易提供自然语言描述，而直接编写信号时序逻辑公式则需要熟悉时序算子、逻辑作用域以及依赖于任务的原子命题。

自然语言天生具有歧义。在不同类型的歧义中，我们关注结构歧义，即由句法结构中的附着关系或作用域差异所导致的歧义。这类歧义在自然语言到信号时序逻辑的转换中特别重要，因为信号时序逻辑对作用域和组合结构高度敏感。

下文中，我们以导航/路径规划作为代表性应用领域。具体而言，我们关注由到达 和 避开等任务谓词、*10*秒内等时间修饰语，以及且、或和同时等连接词构成的自然语言控制描述。这一类别涵盖了许多导航/路径规划指令，同时使问题设定保持明确。尽管示例和数值实验均取自这一领域，但本文的形式化方法本身并不局限于此。请注意，我们关注的是保留结构上不同的解释，并将其转换为信号时序逻辑候选项。只要更改从词语到谓词和原子命题的映射，同一思路也可应用于其他能够用信号时序逻辑表达的控制描述。

作为一个启发性示例，请考虑以下用自然语言描述的任务：

*10*秒内，到达*B*或到达*C*，同时避开*A*。

对于此示例，设 ϕ_A、 ϕ_B和 ϕ_C 分别表示智能体位于区域 A、 B和 C中的原子命题。那么，根据短语同时避开*A*的作用域不同，该句至少有以下两种解释：

$$\varphi_1 = \mathbf{F}_{[0,10]}\phi_B \vee \big(\mathbf{F}_{[0,10]}\phi_C \wedge \mathbf{G}_{[0,10]}\neg\phi_A\big), \tag{5}$$

$$\varphi_2 = \mathbf{F}_{[0,10]}\big(\phi_B \vee \phi_C\big) \wedge \mathbf{G}_{[0,10]}\neg\phi_A. \tag{6}$$

公式 φ_1 对应一种局部作用域解释，其中规避条件仅与到达 C 相关；而 φ_2 对应一种全局作用域

解释，其中规避条件适用于整个析取任务。尽管这两种解释在语言上都合理，但它们对系统施加了不同的约束，因此可能导致不同的规划轨迹。这个示例说明了本文所探讨的核心问题：同一段自然语言描述可能对应多个不同但都合理的信号时序逻辑公式。

B. 问题表述

给定输入的自然语言描述 d，我们的目标是生成一组候选信号时序逻辑规范

$$\mathcal{C}(d) = \{(\varphi_j, s_j)\}_{j=1}^{m}, \tag{7}$$

其中 φ_j 是一个信号时序逻辑规范，$s_j \in \mathbb{R}_{\geq 0}$ 是其合理性评分。直观而言，评分 s_j 表示翻译流程将 φ_j 作为 d 的一种解释时，所给予的支持程度。请注意，该评分用于对候选项进行排序，并不衡量控制性能或信号时序逻辑鲁棒性。在需要时，可以将这些评分归一化为概率

$$p_j = \frac{s_j}{\sum_{k=1}^{m} s_k}, \qquad \sum_{j=1}^{m} p_j = 1, \tag{8}$$

使得 p_j 表示 φ_j 在生成的候选项中的相对概率。

C. 实际益处

为什么生成多个候选项有用？这是因为语言中的歧义可能导致性质截然不同的控制行为。在示例中，使用 φ_1 的规划器可能会接受一条到达 B 的轨迹，即使该轨迹经过 A；而使用 φ_2 的规划器则必须拒绝任何经过 A 的轨迹。因此，如果系统过早地选定一个信号时序逻辑公式，就可能产生形式上符合该公式、但与用户的真实意图不一致的行为。

通过生成多个信号时序逻辑候选项，系统不仅可以呈现公式本身，还可以呈现由这些公式推导出的候选轨迹。这样，用户便能比较可能的行为，在执行前识别非预期的解释，并选定所需的规范。从这个意义上说，候选集充当了含有歧义的自然语言任务描述与形式化控制综合之间的可解释接口。

IV. 保留歧义的自然语言到信号时序逻辑转换

A. 概述

本文利用句法分析 [19],——这是一种自然语言处理中的标准技术——来捕捉结构歧义，并保留对输入句子的多种合理解释。所提出方法的概述如图 1 所示。给定一个自然语言描述 d，该方法会生成信号时序逻辑候选项

通过三阶段处理流程：首先，构建一组可信的解析树 $\mathcal{T}(d)$ 及其相关分数（步骤 1）；

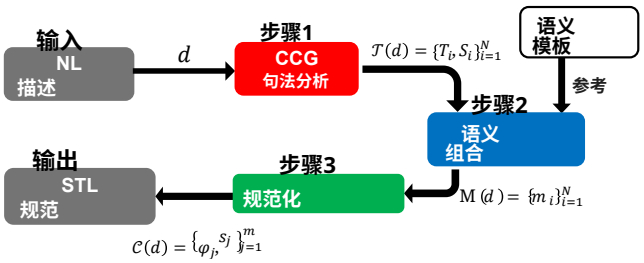


图1. 所提出方法的概述。

接着，通过语义组合推导出一组面向信号时序逻辑的含义表示 $\mathcal{M}(d)$（步骤 2）；最后，将它们转换为去重后的信号时序逻辑候选项集合 $\mathcal{C}(d)$，并附上可信度分数（步骤 3）。

尽管基本组成部分基于标准的组合范畴语法（CCG）语义解析，但所提出的方法并非直接应用现成的 CCG 解析器来获得单一的句法分析；换言之，我们调整了基于 CCG 的句法分析，以支持保留歧义的自然语言到信号时序逻辑（STL）转换。具体而言，如以下小节所述，所提出的处理流程（i）保留多个高分解析树，从而保留附着歧义和作用域歧义；（ii）使用任务专用的语义模板组合面向 STL 的意义表示；（iii）将语义等价的推导规范化，并汇总其分数，形成去重后的 STL 候选项集合。这样，在适当情况下，同一句子的不同句法解读可以传递为不同的 STL 规范；而当不同推导得出相同的 STL 公式时，则会抑制无效重复。下文将详细介绍这一过程的各个阶段。

B. 保留歧义的 *n-best CCG*解析

在步骤 1 中，概率 CCG 解析器对输入句子 d 进行解析，并保留多个得分较高的解析树，而不是只选定一种最佳分析。我们使用组合范畴语法（CCG） [19] 作为结构化解析后端。在本研究中，CCG 的作用并非提供详尽的语言学分析，而是明确表示附着关系和作用域的差异，以便将其传递至 STL 生成阶段。CCG 解析树记录了词语和短语如何组合成句子。当句子存在结构歧义时，解析器可以根据解析分数返回多个合理的解析树，而不是只返回一个。给定输入句子 d，我们得到一个 n-best 解析树集合

$$\mathcal{T}(d) = \{T_i, S_i\}_{i=1}^{N}, \tag{9}$$

其中每个 T_i 都代表对 d的一种合理句法解读，而 $S_i \in \mathbb{R}$ 表示与 T_i相关联的解析分数。在使用 depccg [20]的实现中，S_i 是底层局部因子化 CCG 模型分配的解析器内部得分；该模型结合词元级超标签分数和依存头分数，对候选解析进行排序。

例如，我们再来看下面这个句子：

在 *10*秒内，到达 *B* 或在避开 *A* 的同时到达 *C*。

在这个句子中，短语 *while avoidingA* 既可以附着于子表达式 *reach C* ，也可以附着于整个短语*reach B orreach C*。这两种附着方式对应不同的解析树。由于所提出的方法会保留多个解析树，而不是只选取排名最高的那棵，因此可以保留这两种解读，并将其传递到语义组合阶段。

C. 面向 *STL* 的基于模板的语义组合

在步骤 2 中，基于任务专用词汇语义模板 [19]，通过语义组合为每棵解析树构造面向 STL 的意义表示。给定每棵解析树 $T_i \in \mathcal{T}(d)$，我们通过语义组合构造面向 STL 的意义表示。在基于组合范畴语法的语义解析中，语义组合会为每个词汇项或短语分配一个词汇语义模板，并根据推导的组合结构组合这些模板。此处，语义模板是一条规则，用于将词汇项（及其组合范畴语法范畴）映射为面向 STL 的 λ 表达式。

基本任务表达式映射为中间时序算子，例如 $\mathrm{F}(I, X)$ 和 $\mathrm{G}(I, X)$；而功能表达式映射为 λ 抽象，例如 $\lambda Q.\lambda P.(P \wedge Q)$、$\lambda Q.\lambda P.(P \vee Q)$、$\lambda Q.\lambda P.\,\mathrm{SEQ}(P, Q)$ 和 $\lambda P.\,P(I(a, b))$。此处，我们采用组合语义的标准 λ 演算记法：$\lambda x.\,f(x)$ 表示将参数 x 映射为 $f(x)$ 的函数，而 $\lambda Q.\lambda P.\,f(P, Q)$ 表示依次接受两个语义参数 Q 和 P 的高阶函数。在这种记法中，P 和 Q 表示子任务的中间语义表示，$\mathrm{SEQ}(P, Q)$ 表示顺序组合，而 $P(I(a, b))$ 表示将时间区间 $I(a, b)$ 应用于时序算子 P。此外，$P \wedge Q$ 和 $P \vee Q$ 分别表示逻辑合取和逻辑析取。这些算子是在步骤 2 中使用的中间语义构造，并将在步骤 3 中转换为最终的信号时序逻辑公式。

在我们的实现中，语义组合使用 ccg2lambda [19] 完成。ccg2lambda 最初旨在推导语言学上丰富的意义表示，而我们设计了针对自然语言到信号时序逻辑的转换的任务专用语义模板。将这些模板应用于每棵解析树，可得到

$$\mathcal{M}(d) = \{m_i\}_{i=1}^{N}, \tag{10}$$

其中，每个 m_i 都是沿推导 T_i组合词汇语义模板后得到的面向信号时序逻辑的意义表示。重要的是，m_i 是中间表示，不必已经是完整的良构的信号时序逻辑公式；最终转换、去重和分数聚合留待步骤 3进行。每个 m_i 还继承其源推导的支持分数，之后将在公式层级进行聚合。

表 I 列出了我们实现中使用的代表性语义模板。该表旨在举例说明，并非详尽无遗，重点展示了决定自然语言到信号时序逻辑的转换中任务结构、作用域和时间约束的模板模式。关键在于，模板本身是固定的；歧义源于它们沿解析树组合的方式。

表 I 面向信号时序逻辑的代表性语义模板。符号¬表示否定算子。

表达式	抽象模板
到达 *X*	$\lambda I.\mathrm{F}(I,\ X)$
避开 *X*	$\lambda I.\mathrm{G}(I,\ \neg X)$
or	$\lambda Q.\lambda P.(P \vee Q)$
然后	$\lambda Q.\lambda P.$顺序(P, Q)
当	$\lambda Q.\lambda P.(P \wedge Q)$
在 T 秒内	$\lambda P.P(I(0,\ T))$

因此，不同的解析树会导致诸如 ∧、∨ 和顺序等算子的作用域不同，从而产生不同的中间语义表示。

在这个示例中，一种解析方式将修饰语同时避开 *A*仅与到达 *C*关联，而另一种则将其与整个析取短语到达 *B* 或到达 *C*关联。因此，尽管两种情况下使用的都是表 I 中相同的抽象模板，但它们不同的组合结构会产生 $\mathcal{M}(d)$中的不同元素，随后这些元素会在步骤 3 中转换为不同的信号时序逻辑候选项。在 ccg2lambda 中，实际模板以基于 YAML 的规则实现；在这里，我们略去这些实现细节，聚焦于表 I 中总结的代表性抽象模板模式。

D. 转换为信号时序逻辑、规范化与分数聚合

在步骤 3 中，$\mathcal{M}(d)$ 中的中间意义表示会被转换为信号时序逻辑公式、规范化，并在公式层级上评分。具体而言，步骤 2 中引入的中间算子（如 F、G 和顺序）会被解析为标准的信号时序逻辑语法。由于 $\mathcal{M}(d)$ 中的元素是面向信号时序逻辑的中间表示，而非最终的信号时序逻辑公式，因此从信号时序逻辑语法的角度来看，某些 $m_i \in \mathcal{M}(d)$ 可能不完整或格式不正确；此外，不同的推导在转换后仍可能得到相同的信号时序逻辑公式。步骤 3 通过转换、规范化和分数聚合来解决这两个问题。

对于每个 m_i，我们首先尝试将其转换为完整的信号时序逻辑公式。无法转换为良构的信号时序逻辑公式的表示会被丢弃。随后，对剩余公式进行规范化，使推导过程不同、但依据我们的规范化规则等价的输出都映射到同一个范式。这样可避免因不同解析树或组合路径产生重复公式而人为扩大最终候选集。

最后，在公式层级汇总分数。由于每个 m_i 都会继承其来源推导的支持分数，因此，多个最终归并为同一规范信号时序逻辑公式的推导会共同计入一个候选分数。因此，式 (7) 中的最终集合仅包含唯一的

信号时序逻辑公式，以及反映其从保留歧义的翻译流程中获得的总支持度的合理性分数。这样，步骤 3 既保留了有意义的歧义，又抑制了无效重复。

V. 代表性句子的数值评估

A. 设置

为评估所提出的保留歧义的自然语言到信号时序逻辑翻译流程，我们考虑具有代表性的自然语言任务描述。具体而言，句子集涵盖简单的可达性、有时间界限的任务、析取、顺序组合，以及通过同时避开 A实现的修饰语附着。本节的目的不是提供大规模基准测试，而是验证所提出方法的三项特性：(i) 当输入包含真正的结构歧义时，该方法会生成多个信号时序逻辑候选项；(ii) 当输入在信号时序逻辑层面实际上不存在歧义时，该方法会生成单个规范STL候选项；(iii) 当不同推导归约为同一个规范信号时序逻辑公式时，该方法会合并这些输出。

对于CCG解析，我们使用 depccg [20]，并为每个输入句子保留排名最高的 $N = 40$ 棵解析树，从而得到集合 $\mathcal{T}(d)$，如式 (9) 所示。随后，将每棵解析树映射为面向信号时序逻辑的意义表示，并通过步骤 3 中所述的规范化和分数聚合过程进行处理。每个最终候选项 $\varphi \in \mathcal{C}(d)$都会获得一个聚合分数，该分数通过累加所有归约为 φ的推导所提供的支持度而得出；再按照式 (8) 对聚合分数进行归一化，得到 $p(\varphi \mid d)$。

12个具有代表性的输入句子如下：**S1.** 在10秒内到达B。**S2.** 在避开A的同时，于10秒内到达B。**S3.** 在10秒内避开A并到达B。**S4.** 在10秒内到达B或C。**S5.**在10秒内到达B，或在15秒内到达C。**S6.** 在10秒内到达B，然后在15秒内到达C。**S7.** 在10秒内到达B，然后在15秒内到达C，接着在5秒内到达D。**S8.** 在10秒内到达B，或在避开A的同时到达C。**S9.**在10秒内到达B，或在避开A的同时于15秒内到达C。**S10.** 在10秒内到达B，然后在避开A的同时于15秒内到达C。**S11.** 在10秒内到达B，然后在避开A的同时于15秒内到达C，接着在5秒内到达D。**S12.** 在10秒内到达B，然后在15秒内到达C，或在避开A的同时于5秒内到达D。

其中，S8、S9、S10、S11、S12 分别对应本文讨论的两种动机性歧义模式，即修饰语辖域作用于析取式任务描述，以及修饰语辖域作用于顺序任务描述。

表 II 代表性句子的结果总结。

ID	Type	#参考文献	$\lvert\mathcal{C}(d)\rvert$	已恢复
S1	可达性（基础）	1	1	Yes
S2	合取 +修饰-阳离子（模式 1）	1	1	Yes
S3	合取 +修饰-阳离子（模式2）	1	1	Yes
S4	同时 + 析取-tion	1	1	Yes
S5	不同时间 +析-取	1	1	Yes
S6	序列	1	1	Yes
S7	长序列	1	1	Yes
S8	同时 + 析取式（辖域歧义）	2	2	Yes
S9	不同时间 +析取（辖域歧义）	2	2	Yes
S10	序列（辖域歧歧	2	2	Yes
S11	长序列（作用域歧义）	3	3	Yes
S12	序列 +析取（作用域歧义）	5	5	Yes

B. 结果总结

表 II 总结了针对 12 个代表性句子得到的输出。对于每个句子，“类型”列标明了该句子所针对的主要句法或语义现象，例如简单可达性、修饰语附着、析取、顺序组合或辖域歧义。我们还报告了人工识别的参考解释数量（#Ref.）、规范化后唯一的信号时序逻辑候选项数量（$\lvert\mathcal{C}(d)\rvert$），以及是否恢复了参考解释集。此摘要旨在说明，所提出的方法不仅能在需要时生成多个信号时序逻辑公式，还能在输入在信号时序逻辑层面实际上不存在歧义时，避免不必要的分支。

关键问题在于，所提出的处理流程是否能够有选择地传播歧义。具体而言，具有真实修饰语附着歧义或辖域歧义的句子应产生多个候选结果，而结构更简单的句子则应在步骤 3 后归并为单一的规范化信号时序逻辑公式。这样的结果将表明，所提出的方法并不局限于两个动机示例，而是能够应用于更广泛的代表性任务描述，同时通过规范化和分数聚合控制过度生成。因此，对于所有这些示例，均成功恢复了参考解释集合。

C. 代表性歧义案例的详细输出

表 III 报告了 S8、S9、S10、S11、S12 的完整信号时序逻辑公式和归一化概率。这 5 个案例展示了本文关注的歧义模式：修饰语附着于析取式任务描述，以及修饰语附着于顺序任务描述。

对于所有具有代表性的歧义示例，生成的

表 III 代表性歧义案例的生成信号时序逻辑公式及归一化概率。

信号时序逻辑公式 φ	$p(\varphi \mid d)$
S8: 在 *10*秒内到达 *B* 或 *C*，同时避开 *A*。	
$\mathbf{F}_{[0,10]}\phi_B \vee \left(\mathbf{G}_{[0,10]}\phi_C \wedge \mathbf{G}_{[0,10]}\neg\phi_A\right)$	0.6774565705
$\mathbf{F}_{[0,10]}\left(\phi_B \vee \phi_C\right) \wedge \mathbf{G}_{[0,10]}\neg\phi_A$	0.3225434295
S9: 在避开 *A* 的同时，于 *10*秒内到达 *B*，或于 *15*秒内到达 *C*。	
$\mathbf{F}_{[0,10]}\phi_B \vee \left(\mathbf{F}_{[0,15]}\phi_C \wedge \mathbf{G}_{[0,15]}\neg\phi_A\right)$	0.86272391
$\left(\mathbf{F}_{[0,10]}\phi_B \vee \mathbf{F}_{[0,15]}\phi_C\right) \wedge \mathbf{G}_{[0,15]}\neg\phi_A$	0.13727609
S10: 在避开 *A* 的同时，于 *10*秒内到达 *B*，然后于 *15*秒内到达 *C*。	
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C \wedge \mathbf{G}_{[0,15]}\neg\phi_A\right)$	0.9896166506
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C\right) \wedge \mathbf{G}_{[0,25]}\neg\phi_A$	0.0103833494
S11: 在*10*秒内到达 *B*，然后在*15*秒内到达 *C*，再在*5*秒内到达 *D*，同时避开 *A*。	
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\left(\phi_C \wedge \mathbf{F}_{[0,5]}\phi_D\right) \wedge \mathbf{G}_{[0,5]}\neg\phi_A\right)$	0.934112393
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\left(\phi_C \wedge \mathbf{F}_{[0,5]}\phi_D\right) \wedge \mathbf{G}_{[0,20]}\neg\phi_A\right)$	0.0599828964
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\left(\phi_C \wedge \mathbf{F}_{[0,5]}\phi_D\right)\right) \wedge \mathbf{G}_{[0,30]}\neg\phi_A$	0.0059047106
S12: 在*10*秒内到达 *B*，然后在*15*秒内到达 *C*，或者在*5*秒内到达 *D*，同时避开 *A*。	
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \left(\left(\mathbf{F}_{[0,15]}\phi_C \vee \mathbf{F}_{[0,5]}\phi_D\right) \wedge \mathbf{G}_{[0,15]}\neg\phi_A\right)\right)$	0.5004023346
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \left(\mathbf{F}_{[0,15]}\phi_C \vee \left(\mathbf{F}_{[0,5]}\phi_D \wedge \mathbf{G}_{[0,5]}\neg\phi_A\right)\right)\right)$	0.478979327
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \left(\mathbf{F}_{[0,15]}\phi_C \vee \mathbf{F}_{[0,5]}\phi_D\right)\right) \wedge \mathbf{G}_{[0,25]}\neg\phi_A$	0.015082217
$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C\right) \vee \left(\mathbf{F}_{[0,5]}\phi_D \wedge \mathbf{G}_{[0,5]}\neg\phi_A\right)$	0.0054311103
$\left(\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C\right) \vee \mathbf{F}_{[0,5]}\phi_D\right) \wedge \mathbf{G}_{[0,25]}\neg\phi_A$	0.000105011

候选集均包含人工识别的参考解释。尽管所得概率并不均匀，反映了底层解析器的偏好，但多个候选项的存在本身表明，该方法能够将结构诱导的歧义保留到形式化 STL 规范这一层面。更广泛地说，这一评估协议旨在验证同一处理流程能否应用于本文所考虑范围之外的典型句式模式，同时，步骤 3 中的规范化则可抑制不必要的重复。

D. 与基于 *ChatGPT* 的基线进行比较

作为比较，我们使用 ChatGPT 对五个结构歧义输入 S8–S12 进行了提示式通用大语言模型基线评估。本实验的目的并非对所有基于学习的自然语言到信号时序逻辑方法进行全面排名，而是考察经提示的大语言模型能否列举出针对这些示例所识别的、在结构上成立的信号时序逻辑解读。确切的提示词和评估协议见附录I。表 IV 总结了句子级结果，表 V 列出了 ChatGPT 的输出。仅当返回的公式与用于所提方法的人工识别参考解释之一相匹配时，才将其计为正确。在 ChatGPT 的输出中，其余错误情况分为两类：（E1）作用域／锚定不匹配，即规避条件被置于错误的时间层级，或其有效区间有误；以及（E2）结构上不受支持的额外解读，即输出与任何基于参考解析的解读都不对应。

ChatGPT 的输出恢复了 S8、S9 和 S11 的全部参考解读，但未能恢复 S10 和 S12 的全部参考解读。具体而言，S10 未恢复两个参考解读中的任何一个，而 S12 恢复了五个参考解读中的四个。总体而言，返回公式中有 14 个中的 11 个在

表 IV S8–S12 的 ChatGPT基线句子级摘要。“恢复”表示在返回公式中至少一次恢复了多少种参考解释。

ID	返回数	恢复数	正确数	E1	E2
S8	2	2/2	2	0	0
S9	2	2/2	2	0	0
S10	2	0/2	0	1	1
S11	3	3/3	3	0	0
S12	5	4/5	4	1	0
总计	14	11/14	11	2	1

规范化后与参考解释相匹配，涵盖了 S8–S12 中 14 个参考解读中的 11 个。主要失败模式是时间锚定错误。例如，在 S10 中，公式

$$\mathbf{F}_{[0,10]}\left(\phi_B \wedge \mathbf{F}_{[0,15]}(\phi_C \wedge \mathbf{G}_{[0,15]}\neg\phi_A)\right)$$

将$\mathbf{G}_{[0,15]}\neg\phi_A$ 放在内层最终性中，因此规避区间从到达 C 时开始，而不是从预期第二步子任务的开始时刻开始。在 S12 中，公式

$$\mathbf{G}_{[0,15]}\neg\phi_A \wedge \left(\mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C) \vee \mathbf{F}_{[0,5]}\phi_D\right)$$

也不匹配任何参考解释：对于顶层全局解读而言，它过短，因为该解读要求$\mathbf{G}_{[0,25]}\neg\phi_A$；而与后置析取范围解读不同，它被放在外层$\mathbf{F}^{[0,10]}$ 之外。S10 中剩下的未获支持案例将第一步规避条件与同样锚定错误的第二步解读结合起来，产生了一个不受任何参考解析支持的复合公式。

这些观察表明，尽管通过提示引导的大语言模型能够找回某些有效解读，但在固定的提示

协议下，它目前还无法可靠地枚举出所有且仅有结构上许可的信号时序逻辑候选项，也无法穷尽式地找回这些候选项。需要注意的是，大语言模型的输出可能会随提示设计而显著变化，因此应将基线结果理解为依赖于提示，而非 ChatGPT 固有能力的限制。尽管进一步的提示工程或许能改善某些情况，但这种敏感性本身也引发了人们对可复现性以及能否稳定覆盖结构上许可的解读的担忧。相比之下，所提出的符号化流水线仅根据显式表示的解析结构备选项生成候选项，并能为这些情况找回完整的参考集合。

六、当前局限与未来工作

我们提出了一种保留歧义的符号方法，用于将自然语言任务描述转换为信号时序逻辑公式。所提出的处理流程结合了 n 最优组合范畴语法句法分析、面向STL的模板式语义组合，以及带分数聚合的规范化，从而生成一组带评分的候选信号时序逻辑公式，而不是直接选定单一解释。通过这种方式，歧义句子中结构不同的解读可以分别转换为不同的形式化规范，而规范等价的推导则会合并为一个紧凑的候选集。

目前仍存在若干局限。首先，当前的表述假设输入句子提供了可映射为信号时序逻辑时间界限的显式时间信息。因此，对于不含显式时间表达式的任务描述，仍需作出额外假设，当前实现尚未以系统严谨的方式加以处理。其次，当前模板集仅涵盖有限种任务表达式和歧义模式，主要涉及可达性、规避、析取、顺序组合和修饰语作用域。如何将该方法扩展到更丰富的语言现象，例如隐式时间约束、依赖语境的表达或更多样的任务算子，仍有待解决。第三，由于候选集源自外部解析器生成的句法分析，解析器的解析错误或覆盖范围局限可能会直接影响生成的信号时序逻辑候选项的质量。最后，当前评估仅限于有代表性的案例研究，尚未开展大规模基准测试。

这些局限性指明了未来工作的若干方向。一个重要的下一步是处理省略了时间约束或仅隐含指定了时间约束的句子，例如通过引入默认假设、基于上下文的推理或交互式澄清。另一个方向是扩展模板库和所支持的歧义类型，以便能够组合式地转换更广泛的自然语言任务描述。此外，将所提出的候选项生成框架与交互式消歧、下游轨迹合成和基于信号时序逻辑的控制设计相结合也很有意义，这样便可在与用户及控制系统构成的闭环中选择、完善并验证候选公式。最后，构建更全面的歧义感知型自然语言到信号时序逻辑转换评估基准，将有助于在统一协议下比较符号方法与基于学习的方法。

参考文献

[1] O. Maler 和 D. Nickovic，“连续信号时序属性的监测”，载于实时与容错系统形式化技术国际研讨会，第152–166页，施普林格，2004年。[2] A. Donz´e 和 O. Maler，“实值信号上的时序逻辑鲁棒满足性”，载于定时系统形式化建模与分析国际会议，第92–106页，施普林格，2010年。[3] V. Raman、A. Donz´e、M. Maasoumy、R. M. Murray、A. L. Sangiovanni-Vincentelli 和 S. A. Seshia，“基于信号时序逻辑规范的模型预测控制”，载于*2014*年*IEEE*第*53*届决策与控制年会（*CDC*），第81–87页，2014年。[4] Y. Lin、H. Li 和 M. Althoff，“信号时序逻辑谓词的模型预测鲁棒性”，*IEEE*机器人与自动化快报，第8卷，第12期，第8050–8057页，2023年。[5] K. Leung、N. Ar´echiga 和 M. Pavone，“通过信号时序逻辑规范进行反向传播：将逻辑结构融入基于梯度的方法”，《国际机器人研究杂志》，第42卷，第6期，第356–370页，2023年。[6] C. Lignos、V. Raman、C. Finucane、M. P. Marcus 和 H. Kress-Gazit，“基于自然语言的可证明正确反应式控制”，《自主机器人》，第38卷，第1期，第89–105页，2015年。[7] C. Wang、C. Ross、Y.-L. Kuo、B. Katz 和 A. Barbu，“面向具身机器人的自然语言到线性时序逻辑可执行语义解析器学习”，载于第*4*届机器人学习会议论文集，第155卷《机器学习研究论文集》，第1706–1718页，PMLR，2021年。[8] J. X. Liu、Z. Yang、I. Idrees、S. Liang、B. Schornstein、S. Tellex 和 A. Shah，“在未见环境中为时序任务落地复杂自然语言指令”，载于第*7*届机器人学习会议论文集（J. Tan、M. Toussaint 和 K. Darvish 编），第229卷《机器学习研究论文集》，第1084–1110页，PMLR，2023年11月。[9] J. Pan、G. Chou 和 D. Berenson，“用于机器人任务规范的自然语言到线性时序逻辑转换器的数据高效学习”，载于*2023*年*IEEE*机器人与自动化国际会议（*ICRA*），第11554–11561页，IEEE，2023年。[10] V. Cohen、J. X. Liu、R. Mooney、S. Tellex 和 D. Watkins，“机器人语言落地综述：符号与嵌入之间的权衡”，载于第三十三届人工智能国际联合会议论文集，*IJCAI-24*（K. Larson 编），第7999–8009页，人工智能国际联合会议组织，2024年8月。综述专题。[11] J. He、E. Bartocci、D. Nickovic、H. Isakovic 和 R. Grosu，“DeepSTL：从英语需求到信号时序逻辑”，载于第*44*届软件工程国际会议论文集，第610–622页，计算机协会，2022年。[12] Y. Chen、R. Gandhi、Y. Zhang 和 C. Fan，“NL2TL：使用大语言模型将自然语言转换为时序逻辑”，载于*2023*年自然语言处理实证方法会议论文集（H. Bouamor、J. Pino 和 K. Bali 编），（新加坡），第15880–15903页，计算语言学协会，2023年12月。[13] Y. Mao、T. Zhang、X. Cao、Z. Chen、X. Liang、B. Xu 和 H. Fang，“NL2STL：使用Llama2将逻辑自然语言转换为信号时序逻辑”，*IEEE*控制论与智能系统国际会议论文集，*CIS*，第2024期，第469–474页，2024年。第11届IEEE控制论与智能系统国际会议和第11届IEEE机器人、自动化与机电一体化国际会议（CIS-RAM 2024）。[14] Y. Fang、Z. Jin、J. An、H. Chen、X. Chen 和 N. Zhan，“利用多样化外部知识和大语言模型增强自然语言到信号时序逻辑的转换”，载于计算语言学协会论文集：*ACL 2025*精选论文（W. Che、J. Nabende、E. Shutova 和 M. T. Pilehvar 编），（奥地利维也纳），第10446–10458页，计算语言学协会，2025年7月。[15] S. Mohammadinejad、S. Paul、Y. Xia、V. Kudalkar、J. Thomason 和 J. V. Deshmukh，“从自然语言机器人任务描述到STL的系统化翻译”，载于弥合人工智能与现实之间的鸿沟：第二届国际会议，*AISoLA 2024*，希腊克里特岛，*2024*年*10*月*30*日至*11*月*3*日，论文集（B. Steffen 编），第15217卷《计算机科学讲义》，第259–276页，施普林格，2025年。[16] I. Gavran、E. Darulova 和 R. Majumdar，“从示例和自然语言交互式综合时序规范”，论文集

表 V ChatGPT 针对具有代表性的歧义案例生成的信号时序逻辑规范。原始输出已规范化为稿件中的记法 $(\phi_A, \phi_B, \phi_C, \phi_D)$。“正确”表示该输出经过规范化后与人工识别的某种参考解释相符。

信号时序逻辑公式 φ	评估	Note
S8: 在 *10*秒内到达 *B* 或 *C*，同时避开 *A*。		
$\mathbf{F}_{[0,10]}\phi_B \vee (\mathbf{G}_{[0,10]}\neg\phi_A \wedge \mathbf{F}_{[0,10]}\phi_C)$	**正确**	与局部作用域的参照项匹配-ence.
$\mathbf{G}_{[0,10]}\neg\phi_A \wedge \mathbf{F}^{[0,10]}(\phi_B \vee \phi_C)$	**正确**	等同于全局作用域规范化后的参照。
S9: 在避开 *A* 的同时，于 *10*秒内到达 *B*，或于 *15*秒内到达 *C*。		
$\mathbf{F}_{[0,10]}\phi_B \vee (\mathbf{G}_{[0,15]}\neg\phi_A \wedge \mathbf{F}_{[0,15]}\phi_C)$	**正确**	与局部作用域引用相匹配-ence.
$\mathbf{G}_{[0,15]}\neg\phi_A \wedge (\mathbf{F}_{[0,10]}\phi_B \vee \mathbf{F}_{[0,15]}\phi_C)$	**正确**	符合全局作用域引用——ence.
S10: 在 *10*秒内到达 *B*，然后在避开 *A* 的同时于 *15*秒内到达 *C*。		
$\mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}^{[0,15]}(\phi_C \wedge \mathbf{G}_{[0,15]}\neg\phi_A))$	**E1**	规避区间是一个——锚定在 *C*-到达 in-常数，而非在第二步子任务的开头。
$(\mathbf{G}_{[0,10]}\neg\phi_A \wedge \mathbf{F}_{[0,10]}\phi_B) \wedge \mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}^{[0,15]}(\phi_C \wedge \mathbf{G}_{[0,15]}\neg\phi_A))$	**E2**	不支持的复合解读；既不符合局部作用域也不符合全局作用域参照。
S11: 在 *10*秒内到达 *B*，然后在 *15*秒内到达 *C*，接着在 *5*秒内到达 *D*，同时避开 *A*。		
$\mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}_{[0,15]}(\phi_C \wedge (\mathbf{G}_{[0,5]}\neg\phi_A \wedge \mathbf{F}_{[0,5]}\phi_D)))$	**正确**	匹配局部-*D*-作用域引用-ence.
$\mathbf{F}_{[0,10]}(\phi_B \wedge (\mathbf{G}_{[0,20]}\neg\phi_A \wedge \mathbf{F}^{[0,15]}(\phi_C \wedge \mathbf{F}_{[0,5]}\phi_D)))$	**正确**	匹配 the $C \rightarrow D$ 子序列范围参考。
$\mathbf{G}_{[0,30]}\neg\phi_A \wedge \mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}_{[0,15]}(\phi_C \wedge \mathbf{F}_{[0,5]}\phi_D))$	**正确**	符合全局作用域的引用-ence.
S12: 在*10*秒内到达*B*，然后在*15*秒内到达*C*，或者在*5*秒内到达*D*，同时避开*A*。		
$\mathbf{F}_{[0,10]}(\phi_B \wedge (\mathbf{F}_{[0,15]}\phi_C \vee (\mathbf{G}_{[0,5]}\neg\phi_A \wedge \mathbf{F}_{[0,5]}\phi_D)))$	**正确**	匹配后置的B 局部-D 引用-ence .
$\mathbf{F}_{[0,10]}(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C) \vee (\mathbf{G}_{[0,5]}\neg\phi_A \wedge \mathbf{F}_{[0,5]}\phi_D)$	**正确**	与顶层局部-D 引用-相关 .
$\mathbf{F}_{[0,10]}(\phi_B \wedge (\mathbf{G}_{[0,15]}\neg\phi_A \wedge (\mathbf{F}_{[0,15]}\phi_C \vee \mathbf{F}_{[0,5]}\phi_D)))$	**正确**	与后置B 析取-宽引用相匹配。
$\mathbf{G}_{[0,25]}\neg\phi_A \wedge \mathbf{F}_{[0,10]}(\phi_B \wedge (\mathbf{F}_{[0,15]}\phi_C \vee \mathbf{F}_{[0,5]}\phi_D))$	**正确**	匹配后续的B 全局作用域参照。
$\mathbf{G}_{[0,15]}\neg\phi_A \wedge (\mathbf{F}^{[0,10]}(\phi_B \wedge \mathbf{F}_{[0,15]}\phi_C) \vee \mathbf{F}_{[0,5]}\phi_D)$	**E1**	与两种参考都不匹配：它太短，不足以构成顶层全局解读，而且位置也不正确析取后的层级，宽泛解读。

评估标签。正确：规范化后与人工识别的参考解释相匹配。**E1**：作用域／锚定不匹配，即规避区间被置于错误的时间层级，或有效区间有误。**E2**：结构上不受支持的额外解读，即输出与任何基于参考解析的解读均不对应。

*ACM*编程语言论文集，第4卷，OOPSLA期，第201:1–201:26页，2020年。[17] M. Cosler、C. Hahn、D. Mendoza、F. Schmitt 和 C. Trippel，“nl2spec：使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，第13965卷《计算机科学讲义》，第383–396页，施普林格，2023年。[18] S. Mohammadinejad、J. Thomason 和 J. V. Deshmukh，“使用信号时序逻辑从自然语言和示范中进行交互式学习”，*CoRR*，第abs/2207.00627卷，2022年。[19] P. Mart´ınez-G´omez、K. Mineshima、Y. Miyao 和 D. Bekki，“ccg2lambda：一种组合语义系统”，载于*ACL-2016*系统演示论文集（S. Pradhan 和 M. Apidianaki 编），（德国柏林），第85–90页，计算语言学协会，2016年8月。[20] M. Yoshikawa、H. Noji 和 Y. Matsumoto，“使用超标签和依存分解模型的A* CCG解析”，载于计算语言学协会第*55*届年会论文集（第*1*卷：长篇论文），第277–287页，计算语言学协会，2017年。

附录I
ChatGPT基线的提示词与评估协议
ChatGPT基线

我们使用了 ChatGPT（GPT-5.4 Thinking，访问日期为*2026*年*3*月*29*日）。每个输入句子都在一个新的聊天会话中进行评估。原子命题定义为 ϕ_A、ϕ_B、 ϕ_C 和 ϕ_D，分别表示智能体位于区域A、区域B、区域C和区域D。S8–S12 使用了相同的提示，仅替换了最后一句。

用于每个输入句子的提示词。

> 给定一条导航指令，以及区域名称到原子命题的映射。仅使用算子$\mathbf{F}^{[a,b]}$、$\mathbf{G}^{[a,b]}$、 $\wedge$、 $\vee$和括号。化简后不要输出重复项。原子命题： ϕ_A：智能体位于区域A；ϕ_B：智能体位于区域B； ϕ_C：智能体位于区域C； ϕ_D：智能体位于区域D。句子：S8/S9/S10/S11/S12 仅返回带编号的信号时序逻辑公式列表。

在评估中，经过规范化后，将ChatGPT原始输出与人工确定的参考解释进行比较。如果某个原始输出与某个参考解释匹配，则标记为正确；否则，将其归类为第五节中定义的一种或多种错误类型。