一个事件终结性和圆满性的依赖类型演算

Pavel Kovalev

Carlo Angiuli

June 2025

Abstract

我们提出了一个依赖类型的跨语言框架，用于分析事件的终结性和完成性，并附有使用该框架建模英语句子的示例。我们的框架由两个部分组成。在名词域中，我们建模名词短语的有界性及其与子类型、限定量和形容词修饰的关系。在动词域中，我们定义一个依赖事件演算，将有界施事者的事件建模为终结事件，将达到其内在终点的终结事件建模为完成事件，同时考虑副词性修饰。在这两个领域中，我们特别关注相关的蕴涵关系。我们的框架被定义为依赖类型理论中的一个依意 Martin-LX_{MATH}XDQ 扩展，而且本文中的规则和示例已经在 Agda 证明助手中形式化。

自从自然语言正式语义学领域兴起以来，Church 的简单类型理论及其变体在自然语言形式分析中一直占据主导地位。然而，最近越来越多的人对使用依赖类型理论来实现这一目标表现出兴趣。与简单类型理论不同，依赖类型理论通过允许类型由项进行索引，在类型层面上具有更大的表达能力，从而可以分析或建模在简单类型理论环境中具有问题或不可能的某些自然语言现象。

在本文中，我们开发了一种基于依赖类型理论的框架，用于建模自然语言中涉及体完结性现象的片段，即事件具有内在终点的属性、体累积性，即一个体完结事件达到其终点的属性，以及它们的对应属性，如非体完结性和非体累积性。¹ 正如语言学文献中广泛讨论的那样 (?????????)，对于一大类动词而言，事件的体完结性取决于其受事的属性，受事是描述该事件的动词中最像患者的论元。

我们的框架本质上利用类型依赖，不仅用于在事件的终结性和其承受者的属性之间建立组合关系，而且还用于解释（非）完成性这一属性仅对终结事件有定义的事实，因为（非）完成性预设了一个固有终点的存在。

0.1 终结性和完成性

终止性是指事件具有固有终点的特性。虽然很难精确定义固有终点，但相对容易说明什么构成或不构成一个固有终点，以及达到该终点意味着什么。

下面的每个句子都描述了一个具有内在终点的事件。（在语言的使用上，我们此后将描述事件的句子与所描述的事件混为一谈。）终点在每个句子中要么作为动词的参数词汇上指定，例如 (1-a) – (1-d)，² 要么作为附加语，例如 (1-e)。例如，在 (1-a) 中，汤提供了一个内在的终点，我们说，如果汤被完全消费掉，这个终点就达到了。³ 换句话说，任何具有内在终点的事件都有一些与之自然相关的结果状态，如果结果状态发生，我们就说这个终点达到了。与 (1) 中的句子相关联的结果状态可以通过 (2) 中的句子来表达。

¹ 这些现象属于内时貌的范畴（也称为词汇体或情况体），与外时貌的范畴相对（也称为语法体或视点体）；参见例如？。我们将不涉及外时貌，也不涉及时态。

² 在？中，这样的论证被称为终止对象。在我们的设置中，更合适的名称应为圆满对象，其中“对象”是以非句法意义使用的。

³ 需要强调的是，汤仅仅提供了一个内在的终点，但说汤本身是一个内在的终点可能会有些违反直觉。

- (1)
 - a. John ate the soup.
 - b. John repaired my computer.
 - c. John wiped the table.
 - d. Three balloons popped.
 - e. John walked to the park.
- (2)
 - a. The soup doesn't exist anymore.
 - b. My computer is repaired.
 - c. The table is clean.
 - d. Three balloons are popped.
 - e. John is in the park.

相反，下面在 (3) 中的句子没有固有的终点。例如，句子 (3-a) 没有固有的终点，因为它可能无限期地继续下去；不像 (1-a)，吃汤的事件没有自然停止的点。⁴ 不像在 (1-a) 中的汤，(3-a) 中的汤不提供固有的终点，因此，与 (3-a) 相关的意图结果状态不存在。

- (3)
 - a. 约翰喝了汤。
 - b. 约翰修理了电脑。
 - c. 约翰擦了桌子。
 - d. 气球爆炸了。
 - e. 约翰在公园里散步。

(请注意，尽管 (3) 中的句子没有自然的（预期）结果状态的概念，但由于它们的动态性质，它们确实都涉及某种状态的变化。)

一个事件（或更一般地说，事件性；见备注 ??）是完全事件当且仅当它有一个内在的终点；否则，它是不完全事件。因此，(1) 中的事件是完全事件，而 (3) 中的事件是不完全事件。区分完全事件和不完全事件的主要特点是，完全事件具有达到或实现其内在终点的力量。相反，是否一个不完全事件达到终点的问题是不合理的，因为没有一个内在的终点可以在其上发生终结。

Remark 0.1.1. 根据 ?，我们广义地使用“可能性”这一术语，与 ? 中的“事态”术语类似。它包含了 Vendler 的四个动作类型：状态、活动、成就、完成(??)。我们使用“事件”一词仅指动态（或非静态）可能性，即活动、成就和完成。

一个事件达到其固有终点（即，达到顶点）如果该事件蕴含其相关的预期结果状态。在英语中，通过大多数具有指定固有终点的非进行时动词短语表达的事件会自动达到顶点。例如，(1) 中的事件（可能不包括 (1-c)，如下面讨论的）达到其固有终点，因此达到顶点。我们在 (4) 中对此进行说明：除去 (4-c) 外，每句的第一个分句在其终点上蕴含达到顶点，因此第二个分句引入了矛盾的信息。

- (4)
 - a. #约翰喝了汤，而我把它喝完了。
 - b. #约翰修好了我的电脑，但电脑还是不能用。
 - c. 约翰擦了桌子，但桌子还是脏的。
 - d. #三个气球破了，但其中一个仍然充满气。
 - e. #约翰走到了公园，但他从未到达那里。

对于 (4-c)，动词“wipe”可能在来回移动布的活动和真正尝试将桌面的表面清洁这两者之间存在歧义。⁵ 在前一种解释下，(4-c) 中的第一个从句是非终止性的（在这种情况下，没有

⁴有人可能会反对说它不能无限期地继续下去，因为它是过去时态，但如果将过去时态替换为将来时态，我们打算传达的对比仍然存在。

⁵感谢 Thomas Grano 在个人交流中指出这一点。

固有的终点，因此第二个从句没有可能的达到否定)，在后一种情况下，它是终止性但不达成终点的，这一点可以从第二个从句中看出。

(4-c)的第二种解释可以说是英语中少数几个具有目的但非完成的事件之一，但在某些语言中，非完成事件（也称为非完成成就）比英语中更为显著。我们仅限于一个来自普通话的例子，在 ?? 中给出；英文翻译在语义上在英语中是不可接受的，但原句在普通话中是可以接受的。

玛丽吃了一个苹果，还剩两口，被我吃了。

最后，我们注意到有一类动词，包括推、抚摸和驾驶，其词汇语义不涉及任何目标的接近，这与修理、爆破和融化等动词不同，后者的词汇语义确实涉及对某个目标的接近。像推、抚摸或驾驶这样的动词的论元永远不能引入固有的终点；只有附加语可以。这在 (4) 中有所说明。

约翰开车。 (非终点) 。 约翰开车去布卢明顿。 (终点)

总结来说，事件可以有终点的 (telic) 或无终点的 (atelic)。有终点的事件可能是达成终点的或未达成终点的，而无终点的事件则定义为不考虑达成终点的状态。在那些由词汇语义编码了某个目标指向的动词描述的事件中，动向性可能来自动词的一个论元或附加成分。相反，在没有目标指向的动词描述的事件中，在没有额外语言内容的情况下总是无终点的，虽然可以通过添加附加成分等额外语言内容使事件成为有终点的。(我们在本文中不考虑这种情况，将其留待未来研究。)

0.2 终结性与承受者之间的关系

根据角色与指称语法 (??)，我们使用术语“施事者”和“承受者”来标记及物动词的两个论元。施事者和承受者是语义上的宏观角色：施事者（或承受者）表示事件中最像代理者（或患者）的参与者。施事者的范围比代理者更广；它们不必是有意识的，并包括其他类似代理者的角色，如工具或力。我们的承受者概念也涵盖了各种传统的语义角色，如患者、主题、体验者、状态持有者和类似角色。例如，(1) 和 (3) 中的所有及物句的语法主语都是施事者，语法宾语是承受者；在 (2) 中的不及物句中的语法主语都是承受者。

如在第 0.1 节中所讨论的，对于那些词汇语义包含实现目标的意图的动词，一个固有的终点可能由动词的一个参数或一个附加语提供。将注意力限制在前一种情况和最多含有两个参数的动词，语言学家观察到动词域中的终结性特性与名词域中受事者的某一特性相关联 (?????????)。

?? 称后面的属性为界限性。根据 Jackendoff 的说法，一个有界限的名词短语是指一个被限定数量的某物或某种物质，例如，三个苹果、五升水、若干本书、一个番茄。没有界限的名词短语或不定名词短语的例子包括苹果、水、奶油冻。

我们可以通过“介词短语测试”检测动作终结性和有界性的关联：只有终结动作能够接在“介词短语”之后。实际上，“介词短语”规定了某件事情发生的时间跨度，而这种“发生”只有在存在固有终点时才可能发生；参见 ?, 223。另一方面，在英语中，只有包含有界承受体的句子才能与“介词短语”兼容，如 0.2 中的对比所示。这说明了一件事件的终结性与其承受体的有界性之间的关联。

约翰在十分钟内吃了三个苹果。 。 < 约翰在十分钟内吃了苹果。

Remark 0.2.1. 有一些已知的例外，特别是涉及量词的名词短语。例如，名词短语至少三个苹果在任何意义上都不是有限的，但根据 in -副词测试，它应该是有限的。另一方面，名词短语一些苹果似乎是有限的，但根据 in -副词测试，它的行为像一个无限的名词短语。

在英语中，有限和无限的名词短语在句法上有所标记：无限的名词短语是无冠词名词短语（即由形容词修饰但没有任何限定词的纯名词短语），而有限的名词短语是非无冠词名词短语。⁶ 然而，在无冠词语言中，纯名词短语可以具有特定的解读。

⁶正如在 Remark 0.2.1 中已经观察到的那样，像 some apples 这样的名词短语是这一概括的例外。

为了解释这种跨语言的变化，我们重新定义有界性概念，使其不涉及名词短语的句法结构。在我们的定义中，如果一个名词短语能够提供固有的终点，则被视为有界，否则为无界。也就是说，我们将有界性 (resp., 无界性) 视为终结性 (resp., 非终结性) 的名词对应关系，即如果一个事件的经历者是有界的，那么该事件就是终结的。(当然，这仅适用于词汇语义中包含实现目标的动词。) 在 ? 中，第一作者认为这一定义，以及检测有界和无界经历者的适当诊断，是对 ?'s (?) 有界性概念的适当概括，至少适用于某些缺乏冠词的语言，例如俄语，并且当限制在英语时，这两个有界性概念是一致的。

在本文中，我们将把注意力限制在那些词汇语义中包含达成目标的目的的及物和不及物动词 (即，有些人可能称之为“非活动”动词)，并且在动词论元而不是附加成分诱导完成性时的情况。这些动词的事件的完成性与其承受者的有限性相关联。

如上所述，我们将名词短语的有界性视为一个基本概念。在英语中，有界名词短语是那些非裸名词短语 (即包含限定词的名词短语)。对于那些裸名词短语既有有界又有无界解读的语言，如俄语，第一作者在 ? 中提出，在俄语中，裸名词短语的有界解读可以被解释为确定描述，或者被解释为具有一个未指定的数值量词。为了本文的目的，我们将自己限制在后一种解释上，把确定描述 (以及更一般的跨从句关系) 的建模留待未来的研究。

0.3 与先前关于完结性和达到顶点的研究的比较

尽管完结性和终止性在语言学文献中得到了充分研究，先前的研究几乎完全专注于解释这些现象的起源。相比之下，我们的关注点是与这些概念相关的蕴涵关系。可以说，蕴涵的概念是整个语义学中一个核心且最重要的概念；确实，它是将语义学与邻近的语用学领域区分开来的关键，后者的核心概念是会话含义。尽管如此，之前的工作并没有专注于正式建模在完结性语境中产生的蕴涵关系。我们也不知晓有研究使用依赖类型来建模完结性；我们所知的最接近的工作是 ?'s (?) 对 Vendler 的动作类划分的建模的简要介绍 (??)。

当前关于终止性和成就性的形式分析大多数可以大致分类为两类：一类是涉及事件和实体部分的整体论方法，另一类是涉及谓词分解的词汇语义学方法。前者可以追溯到 ?????，而后者最初在 ? 中提出，并在 ??? 中得到进一步发展，? 则提供了一个接近的替代方案。这两种方法在 ? 中得到了结合。

简单来说，另一个方法假设一个事件终结的前提是该事件的承受者被完全“耗尽”。例如，事件“约翰吃了三个苹果”在且仅在这三个苹果被完全吃掉时才结束。这种方法将终结性和顶点关联混为一谈；正如我们在第 0.1 节中讨论的那样，这种混淆一般适用于建模英语中的终结性，但——不像我们的方法——不适用于那些具有终结性但非顶点关联事件的语言。

另一方面，基于词汇分解的方法在描述事件的谓词的符号表示中编码事件的相关内在体貌属性。为了说明这一点，0.3 和 0.3 提供了在角色和指称语法 (? 99) 中某些类型的词汇表示的例子。

这艘船沉没了。 。 BECOME *sunken* (ship)

潜艇击沉了这艘船。 。 [*do* (submarine, É)] CAUSE [BECOME *sunken* (ship)]

我们对这种表示的主要批评是，它们通常没有附带形式语义 (无论是模型论、证明论或其他方式)。因此，尽管它们为句子提供了合适的符号表示，但由于缺乏语义，这使得讨论蕴涵 (以及因此进行自然语言推理) 成为不可能。

我们注意到，与之前描述的部分论元学方法类似，前述角色与参照语法中的例子将终结性与到达性相混淆。已有尝试统一这两种方法并增加一个单独的谓词来管理到达性，尤其是由 ? 所做的工作。尽管这些工作尝试定义语义，但在许多方面它们表现得不够正式，并且将句法和语义相混淆，这使得它们在自动化自然语言推理的背景下不适合形式化。

在本文中，我们旨在纠正这些问题。特别是，我们区分了终止性和完成性的概念，并进一步注意到，(非)完成性仅是为终止性事件定义的属性。我们对建模终止性的处理可以看作是对词汇分解方法的一种改进，在这种方法中，我们在 (依赖) 类型理论中工作，而不是创建一种单独的表达语言。

使用类型理论为我们提供了几个好处。首先，类型理论是一个足够丰富的框架，可以在不混淆两个概念的情况下建模终结性和达到性及其与论元结构之间的关系。其次，存在一个依赖类型理论模型的普遍概念（例如，由广义代数理论诱导的模型(??)），这使我们能够将框架配备语义，作为类型理论(?)的标准集合论语义的直接扩展。（有关这一点的进一步讨论，请参见第??节。）

最后，类型理论在现代证明助理中有许多计算机实现，例如 Agda(?)、Rocq/Coq(?)、Lean(?) 和 Lego/Plastic(??)。因此，我们在本文中开发的框架适合在证明助理中实现；实际上，在第3节中，我们描述了该文章中所有规则和示例的 Agda 实现。这样的实现可以进一步用于自动化自然语言推理，或者更广泛地用于自然语言推理(?)。先前已经在 Rocq/Coq(??)、Plastic(?) 和 Agda(?) 中实现了部分英语片段。

0.4 先前在语言建模中使用依赖类型

使用依赖类型进行自然语言分析至少可以追溯到 ? 和 ?，在那里他们被用于建模所谓驴子句中的照应关系(?)。虽然关于使用依赖类型进行自然语言分析的第一篇全面性工作大约在同一时间出现(?)，但随后的几年中相关研究相对较少(?????)。

在过去的十到十五年中，出现了几条将依赖类型应用于自然语言的工作路径。其中，可能最为成熟的框架是所谓的多类型理论语义学(MTT-semantics)(???)。另一个已发展的工作方向是依赖类型语义学(????)，以及具有记录的类型理论(?????)；还有许多其他独立的工作(?????????)。

除了上面提到的依赖类型论的一般优点之外，许多研究人员开始转向依赖类型论进行自然语言分析，因为与丘奇的简单类型论相比，它提供了对自然语言现象更为优雅的解释——或者在某些情况下，提供了任何解释——这些现象包括回指关系、共指、选择限制、事件量化问题和副词修饰等。(??)。

我们在本文中的框架在许多方面受到 MTT 语义学的启发。我们注意到，术语“MTT 语义学”代表“现代类型理论中的形式语义学”，原则上意在涵盖任何采用现代（依赖）类型理论的形式语义学。然而，在实践中，这个术语似乎指的是 ??? 开发的特定框架，其基础类型理论是依赖类型的统一理论(?)，我们正是以这种意义来使用该术语。

我们的工作与 MTT 语义学的主要共同点是使用共同名词作为类型的范式(?)，这可以追溯到早期的类型理论工作(???)，但与蒙太格语义学中将共同名词作为谓词的处理方法相对立。在 MTT 语义学中，将共同名词建模为其实例的类型，可以对涉及子类型的语言现象（如共谓）进行建模。在本文中，这使我们对名词和动词领域的处理更加统一。⁷

我们也采用了 MTT 语义学中事件类型依赖的一种变体(?)，尽管我们的方法有所不同，因为我们的依赖事件类型是事件的集合，这些事件自身就是类型。正如我们将在第2节讨论的那样，这一修改使我们能够在与形容词修饰并行的情况下应用副词修饰，同时还能够追踪事件的终止性和完成性状态。

我们与 MTT 语义学最大的不同在于底层的类型论本身。MTT 语义学建立在非谓词性依赖类型统一理论(?)之上，并通过许多特性进行扩展，例如矫正子类型(???)、子类型宇宙(??)和点类型(?????)。这些特性在建模各种自然语言现象中发挥了关键作用；例如，引入点类型是为了处理共同述谓，子类型宇宙用于处理可分级形容词，弱和（可以通过非谓词性定义）用于解释驴子照应(?)。

相反地，我们在谓词性内涵的 Martin-LXMathXDQ 类型理论(?)中工作，这是一个从理论和实践角度都非常易于理解的系统。部分原因是我们并不试图模拟 MTT 语义所考虑的所有语言现象。在其他情况下，我们努力避免复杂的特性；例如，在第1.3节中，我们对子集包含的建模不是像在 MTT 语义中那样使用强制子类型，而是采用显式包含函数的方式。

⁷替代方法包括将普通名词视为集合类型(??)，谓词(????)，以及同时作为谓词和类型(?)。有关普通名词作为类型和谓词方法之间的比较，请参见 ?，更多讨论请参见 ?，Section 2.6。

在一个更标准的依赖类型理论内工作有多种好处。首先，它允许我们在 Agda 证明辅助工具内以最小的修改格式化我们的框架。其次，通过减少背景假设，我们的工作可以直接兼容更多的类型理论扩展，包括同伦类型理论(?)。最重要的是，Martin-LXMathxDQ 类型理论的集合论（以及范畴论，甚至同伦论）语义已得到充分研究(?)，这与含有强制类型转换的不可预测类型理论模型理论相比，后者则模糊得多。后者导致了一些论文试图澄清 MTT-语义除了是证明理论的，也同样是模型理论的(??)。

0.5 贡献

我们在 Martin-LXMathxDQ 下的依赖类型理论中开发了一个跨语言框架，用于分析事件成终性、完成性和相关的蕴含。据我们所知，这是依赖类型理论中成终性的首次分析。本文中我们专注于该框架在英语中的应用；第一作者的论文(?)中还包括了对俄语的深入讨论。特别是，我们的分析仔细地地区分了成终性和完成性现象，这两者（如在第 0.1 节中所讨论）通常在英语中一致，但在例如 俄语或普通话中则不然。

我们的分析基于一个被广泛观察到的观点，即对于一个大的动词类，事件的终止性与其承担者——最像接受者的参与者的有限性相关(????????)。因此，本论文的很大一部分致力于构建名词短语结构的详细本体论，特别是显然有限的名词短语的内部结构：数量词、计量词、形容词修饰、名词短语的实例以及它们的部分和之类。我们相信，我们框架的这一部分有独立的研究价值。

根据?，我们将事件视为由施动者和承受者索引的依赖类型族。与依赖类型理论的先前工作不同，我们认为事件产生其发生的集合，从而使我们能够轻松地建模副词修饰。由于事件的类型是由承受者的类型索引的，而承受者的类型又进一步由界限性索引，我们能够仅通过类型依赖检测到完成性。然后，我们将每个完成事件与其结果状态关联，并将累积性定义为完成事件在发生时获得其结果状态的属性。

除了对终止性和完成性进行建模之外，我们还旨在构建一个合理的最终事件本体论——既包括静态也包括动态的情况——并推进对事件结构的依赖类型的使用，这方面相对较少受到关注。除了?的依赖事件类型外，?在其 Vender 的动作类型类的建模概述中也进行了一些初步的工作，正如我们在第??节中讨论的那样。

最后，虽然我们的框架在许多方面受到 MTT 语义学(?)的启发，但我们局限于在一个标准的内涵型理论(?)内部工作，而不是使用扩展了各种复杂特性（如强制子类型）的依赖类型统一理论(?)。使用标准的类型理论不仅简化了我们的框架，还使我们能够在 Agda 证明辅助工具中形式化本文中的所有规则和示例，并为我们提供了一条清晰的途径，朝着一种集合论语义学方向，我们可以根据该语义学验证我们的蕴涵性。在这方面，我们的工作可以被视为对词汇（或谓词）分解方法的改进，因为它不仅提供了一种表示语言，而且还提供了一个相关的语义学。

大纲

本文的其余部分结构如下。在第 1 节中，我们快速总结了内涵 Martin-LXMathxDQ 类型理论，我们的环境类型理论。在第??和 2 节中，我们分别展示了框架的名词和动词片段，夹杂着例子。我们的名词框架特别强调跟踪名词短语的有界性，特别是显式有界名词短语的内部结构，我们认为这具有独立的研究兴趣。动词框架基于名词框架以定义和分析不同类型的事件性，包括静态事件性和完成、未完成、完成性和非完成性动态事件性，以及这些之间的蕴涵关系。在第 3 节中，我们讨论并简要展示了我们的 Agda 形式化处理规则和来自第??和 2 节的例子。最后，在第??节中，我们结束讨论了对我们框架的可能扩展以及如何与之前的工作进行比较。

致谢

本文扩展并重构了第一作者论文(?)的部分内容。我们感谢 Thomas Grano 和 Larry Moss 在准备该论文期间与第一作者进行的有益讨论，特别是在语言学方面的讨论。

1 类型理论背景

我们的框架建立在具有内涵递归的 Martin-LX_{MATH}XDQ 的依赖类型理论基础上(?)，这是一种类型理论的标准变体，大致相当于现代类型理论证明助手中可用特性的交集，例如 Agda(?)，Rocq/Coq(?) 和 Lean(?)。不熟悉类型理论的读者可以参考该理论或其变体的众多资料之一，例如??的书籍。在本节中，我们将快速回顾一些基础知识并确定符号。

依赖类型论由四个主要判断组成。第一个， $\Gamma \vdash A \text{ type}$ ，表明在上下文 Γ 下， A 是一个类型， $x_1 : A_1, x_2 : A_2, \dots, x_n : A_n$ 是变量/假设的列表。第二个， $\Gamma \vdash a : A$ ，表明在同样的上下文 Γ 下， a 是类型 A 的一个项。我们可以用两种方式理解判断 $\Gamma \vdash a : A$ ：要么 a 是命题 A 在假设列表 Γ 下的一个证明，要么 a 是一个以 Γ 为索引的 A 集合（以 Γ 为索引的）的元素族。（特别地，当 Γ 为空时， $\cdot \vdash a : A$ 可以被理解为命题 A 的证明，或者是集合 A 的一个元素。）最后，我们有判断等式 $\Gamma \vdash A = A' \text{ type}$ 和 $\Gamma \vdash a = a' : A$ ，表述两个类型（或项） A 和 A' （或 a 和 a' ）是相同的。这四个判断是由一系列自然演绎风格的规则归纳定义的，这些规则可以形式上被理解为共同指定了一个广义代数理论(?)的签名。

在本文中，我们几乎完全依赖于三种类型构造： Π -类型（依赖积）、 Σ -类型（依赖和）以及同伦命题的宇宙 **Prop**。⁸ Π -类型是普通函数类型的依赖泛化，并在逻辑上对应于全称量化。如果 $B(x)$ 是一个由 $x : A$ 索引的类型族，那么 Π -类型 $(x : A) \rightarrow B(x)$ 将由任意 $x : A$ 发送到 $B(x)$ 的元素 b 的函数 $\lambda x. b$ 组成。我们将 $f a : B(a)$ 用于将任意函数 $f : (x : A) \rightarrow B(x)$ 应用于一个参数 $a : A$ 。

$$\frac{\Gamma, x : A \vdash B(x) \text{ type}}{\Gamma \vdash (x : A) \rightarrow B(x) \text{ type}} \qquad \frac{\Gamma, x : A \vdash b : B(x)}{\Gamma \vdash \lambda x. b : (x : A) \rightarrow B(x)}$$

$$\frac{\Gamma \vdash f : (x : A) \rightarrow B(x) \quad \Gamma \vdash a : A}{\Gamma \vdash f a : B(a)}$$

另一方面， Σ 型是普通对偶类型的依赖泛化，并对应于一种存在量化形式。如果 $B(x)$ 是由 $x : A$ 索引的一组类型，那么 Σ 类型 $\sum_{x:A} B(x)$ 由一个项 $a : A$ 和一个项 $b : B(a)$ 的对组成。给定 $p : \sum_{x:A} B(x)$ ，我们写作 $\mathbf{fst}(p) : A$ 和 $\mathbf{snd}(p) : B(\mathbf{fst}(p))$ 来表示它的第一和第二投影。

$$\frac{\Gamma, x : A \vdash B(x) \text{ type}}{\Gamma \vdash \sum_{x:A} B(x) \text{ type}} \qquad \frac{\Gamma \vdash a : A \quad \Gamma \vdash b : B(a)}{\Gamma \vdash (a, b) : \sum_{x:A} B(x)}$$

$$\frac{\Gamma \vdash p : \sum_{x:A} B(x)}{\Gamma \vdash \mathbf{fst}(p) : A} \qquad \frac{\Gamma \vdash p : \sum_{x:A} B(x)}{\Gamma \vdash \mathbf{snd}(p) : B(\mathbf{fst}(p))}$$

我们对于命题类型 **Prop** 的处理需要更多的讨论。我们通过使用“与证明无关”的命题系列（或至多只有一个项的类型）来建模形容词、副词和静态事件性；这确保了例如对于一个给定的猫是黑色的，最多只有一个证明。在类型理论文献中，有许多不同版本的 **Prop**，如同伦命题(?，Section 3.3)、严格命题(?) 和不定命题(?)。我们的要求非常宽松：我们需

⁸我们还假设一种类型宇宙 **Type**，内涵身份类型 $\mathbf{Id}_A(a, a')$ ，以及函数外延性。

要一种类型 $\mathbf{Prop}$ ，其元素 $P : \mathbf{Prop}$ 能够产生最多一个元素的类型 $\mathbf{Prf}(P)$ ，这基于内涵等同类型。

$$\frac{}{\Gamma \vdash \mathbf{Prop} \text{ type}} \quad \frac{\Gamma \vdash P : \mathbf{Prop}}{\Gamma \vdash \mathbf{Prf}(P) \text{ type}} \quad \frac{\Gamma \vdash P : \mathbf{Prop} \quad \Gamma \vdash p : \mathbf{Prf}(P) \quad \Gamma \vdash p'' : \mathbf{Prf}(P)}{\Gamma \vdash \mathit{irr}(p, p'') : \mathbf{Id}_{\mathbf{Prf}(P)}(p, p'')}$$

在我们的 Agda 形式化中，我们选择 $\mathbf{Prop}$ 作为同伦命题的类型

$$\mathbf{Prop} \triangleq \sum_{X:\mathbf{Type}} (x : X) \rightarrow (y : X) \rightarrow \mathbf{Id}_X(x, y)$$

，并在第 2.5 节假设函数外延性以展示当 $P : \mathbf{Prop}$ 时 $X \rightarrow \mathbf{Prf}(P)$ 是一个命题。然而，任何其他符合这些性质的 $\mathbf{Prop}$ 变体同样适合我们的目的。值得注意的是，与 $\mathbf{?}$ 不同的是，我们不要求我们的类型论中有任何形式的不可预测性。

为了模型化各种语言现象，我们的框架通过添加原始类型和术语构造器以及这些新原始内容上的判断等式来扩展内涵型类型论，比如给定边界性 $b : \mathbf{Bd}$ 的名词短语的依赖类型 $\mathbf{NP}^b$ 。这些扩展在第 ?? 节和第 2 节中进行了描述。在这些章节中，我们还通过一系列例子展示如何使用我们的框架来分析英语片段，这些例子通过额外的常数来暂时扩展框架以模拟英语单词，比如由常数 $\mathit{human} : \mathbf{NP}^U$ 表示的无界名词短语 human 。

在展开我们的关于动词领域中的限定性和终结性的研究之前，我们必须首先在名词领域打下基础。特别是，我们通过使用依赖类型来追踪名词短语的界限，以便在第 2 节中检测限定性，从而对 ??? 的类型范例进行细化。

在第 1.1 节中，我们介绍了两种类型，分别对应于有界和无界名词短语的集合。因为我们在将普通名词视为类型的范式中工作，我们将这些类型的每个元素（例如，人类）与其实例的集合关联（在这种情况下，是所有人类的类型）。在第 1.2 节中，我们定义了专有名词的类型，它们是由一个名词短语和该名词短语的特定实例（例如，John，一个特定的人）构成的对。

在第 1.3 节中，我们定义了名词短语之间的一种包含概念，使我们能够表达诸如“每个人都是人类”这样的关系。请注意，在我们的 Martin-LXMathXDQ 类型理论设置中，术语最多只有一种类型；我们并不通过 MTT 语义中的强制子类型(???)来表达包含关系，也不能像在集合论中那样引用子集关系。

在我们的框架中，诸如 human 的英语裸名词，无论是物质名词还是计数名词，都对应于无界名词短语。在第 1.4 节中，我们讨论了有界名词短语的起源，这些名词短语表达了某种限定数量，例如三个苹果。这些数量是根据度和单位来衡量的，遵循 $\mathbf{?}$ 和 $\mathbf{?}$ 。我们主要关注有界名词短语具有明显限定词的情况，如在英语中；然而，我们的框架也可以解释诸如俄语等无冠词语言（备注 ??），其中裸名词可以以有界的意义使用。在这一部分中，我们还解释了物质名词和计数名词之间的区别，并建模一种从原始个体获得复合个体的方法，为未来可能的分配现象分析奠定了基础。

我们认为，尽管第 1.4 节在我们的生成有限名词短语框架中的作用之外，它本身就具有独立的意义，因为在依赖类型的背景下，显式有限名词短语的内部结构并没有受到太多关注。据我们所知，我们的论述是在具有依赖类型的设置中首次对名词短语的内部结构进行或多或少完整的处理。

最后，在第 1.5 节，我们讨论形容词，特别关注由形容词修饰的名词短语的有限性。我们将研究限制在交叉形容词及其与我们框架其他部分的互动，具体来说是有有限性、第 1.3 节中的子类型关系、可数性质以及复合个体。

1.1 名义域中的宇宙

如第 0.4 节所讨论的，标准的 MTT-语义(?) 包括一个普通名词的宇宙 $\mathbf{CN}$ （可能被形容词修饰）。我们对应的是名词短语的宇宙 $\mathbf{NP}$ ，但我们关键在于将 $\mathbf{NP}$ 分为两个子宇宙：有限名

词短语的宇宙 $\mathbf{NP}^{\mathbf{B}}$ 和无限名词短语的宇宙 $\mathbf{NP}^{\mathbf{U}}$ 。

Remark 1.1.1. 注意，我们使用“名词短语”这个术语的方式与生成句法中的用法不同。具体来说，我们不认为专有名词是名词短语；相反，我们将它们分配为单独的类型 **Entity**，这一点将在第 1.2 节中介绍。

首先，我们引入了“有界性种类”的类型 **Bd**，它有两个元素：**B**（表示有界）和 **U**（表示无界）。这本质上是一个不同名字的布尔类型，尽管我们不会在 **Bd** 上进行任何案例分析，直到例子 2.5.2。

$$\frac{}{\Gamma \vdash \mathbf{Bd} \text{ type}} \qquad \frac{}{\Gamma \vdash \mathbf{B} : \mathbf{Bd}} \qquad \frac{}{\Gamma \vdash \mathbf{U} : \mathbf{Bd}}$$

对于任何 $b : \mathbf{Bd}$ ，我们引入一种类型 $\mathbf{NP}^b$ ，称为“具有界限性的 b 的名词短语”或 b -名词短语：

$$\frac{\Gamma \vdash b : \mathbf{Bd}}{\Gamma \vdash \mathbf{NP}^b \text{ type}}$$

一个“有界名词短语”是一个类型为 $\mathbf{NP}^{\mathbf{B}}$ 的术语，而一个“无界名词短语”是一个类型为 $\mathbf{NP}^{\mathbf{U}}$ 的术语。我们将每个 b -名词短语 $np : \mathbf{NP}^b$ 与一种“ np 实例”的类型关联，记作 $\mathbf{El}_{\mathbf{NP}}^b(np)$ ，用于收集 np 的“元素”。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}^b(np) \text{ type}}$$

Notation 1.1.2. 我们对框架中的术语和类型构造函数使用粗体字，sans-serif 用于引入术语常量以使用我们的框架来建模语言片段。我们使用斜体字来指代任意术语（如推理规则前提中的术语）。

所有名词短语的类型 $\mathbf{NP}$ 被定义为针对所有 $b : \mathbf{Bd}$ 上的索引和 (Σ 型)，以便类型 $\mathbf{NP}$ 的术语是一个有界性 $b : \mathbf{Bd}$ 和一些 b -名词短语 $np : \mathbf{NP}^b$ 的对 (b, np) 。名词短语 $n : \mathbf{NP}$ 的实例的类型 $\mathbf{El}_{\mathbf{NP}}(n)$ 可以根据上述规定的原始类型家族 $\mathbf{El}_{\mathbf{NP}}^b$ 来定义。

$$\frac{}{\Gamma \vdash \mathbf{NP} \triangleq \sum_{b:\mathbf{Bd}} \mathbf{NP}^b \text{ type}} \qquad \frac{\Gamma \vdash n : \mathbf{NP}}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}(n) \triangleq \mathbf{El}_{\mathbf{NP}}^{\text{fst}(n)}(\text{snd}(n)) \text{ type}}$$

同样地，未经扩展我们的框架，我们可以将任何 b -名词短语视为一个名词短语。我们说类型 $\mathbf{NP}^b$ 的术语可以“提升”到类型 $\mathbf{NP}$ 。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b}{\Gamma \vdash \mathbf{Lift}_{\mathbf{NP}}^b(np) \triangleq (b, np) : \mathbf{NP}}$$

请注意，根据 $\mathbf{NP}$ 作为 Σ 类型的定义， $\mathbf{Lift}_{\mathbf{NP}}^b$ 是类型良好的。此外，通过展开定义，很容易看出对于任何 $b : \mathbf{Bd}$ 和 $np : \mathbf{NP}^b$ ， $\mathbf{El}_{\mathbf{NP}}(\mathbf{Lift}_{\mathbf{NP}}^b(np)) = \mathbf{El}_{\mathbf{NP}}^b(np)$ 。

最后，我们要求 $\mathbf{NP}^b$ 在 $\mathbf{El}_{\mathbf{NP}}^b$ 上的 Σ -类型的谓词下是封闭的。也就是说，对于任何 b -名词短语 $np : \mathbf{NP}^b$ 和由 $\mathbf{El}_{\mathbf{NP}}^b(np)$ 索引的一族命题 P ，我们要求“满足 P 的 np 的子集”仍

然是一个 b -名词短语。这是一个名词短语被形容词修饰后仍然是名词短语这一条件的推广 (特别地, 它不限于形容词修饰)。还要注意, 这个操作保持了有界性。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma, p : \mathbf{El}_{\mathbf{NP}}^b(np) \vdash P : \mathbf{Prop}}{\Gamma \vdash \sum_{p:np}^{\mathbf{NP}} P : \mathbf{NP}^b}$$

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma, p : \mathbf{El}_{\mathbf{NP}}^b(np) \vdash P : \mathbf{Prop}}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}^b\left(\sum_{p:np}^{\mathbf{NP}} P\right) = \sum_{p:\mathbf{El}_{\mathbf{NP}}^b(np)} \mathbf{Prf}(P) \text{ type}}$$

在对英语的特定片段进行建模时, 我们将用单独的名词以及被交叉形容词修饰的单独名词来填充 $\mathbf{NP}^U$ (第 1.5 节)。因此, $\mathbf{NP}^U$ 在 ? 中扮演与 $\mathbf{CN}$ 相同的角色。

Example 1.1.3. 假设我们有兴趣分析包含词语 *human* 的英语片段。我们通过一个对应于无界名词短语 *human* 的术语

$$\frac{}{\Gamma \vdash \text{human} : \mathbf{NP}^U}$$

扩展我们的框架。类型 $\mathbf{El}_{\mathbf{NP}}^U(\text{human})$ 则是所有人类的类型。为了在我们的片段中进一步包含单词 *John*——一个特定人类的实例——我们将用第二个术语

$$\frac{}{\Gamma \vdash \text{john} : \mathbf{El}_{\mathbf{NP}}^U(\text{human})}$$

来扩展我们的框架。

1.2 实体

关于终结性, 像 *John* (例子 1.1.3) 这样的专有名称表现得与有限名词短语相同。然而, 我们并不主张 *john* 具有类型 $\mathbf{NP}^B$ 。首先, 我们的框架赋予 *john* 的类型为 $\mathbf{El}_{\mathbf{NP}}^U(\text{human})$, 它不能同时具有两种不同的类型。我们也不希望断言一个提升 $\iota : \mathbf{El}_{\mathbf{NP}}^U(\text{human}) \rightarrow \mathbf{NP}^B$, 声明 *human* 的每个实例都是一个 $\mathbf{NP}^B$, 因为这迫使我们考虑 $\mathbf{El}_{\mathbf{NP}}^B(\iota(\text{john}))$ 类型的“*john* 实例”, 这在语义上是有疑问的。

相反, 我们引入了一种新的专有名称或实体, 其元素是一个名词短语 $n : \mathbf{NP}$ 和一个 $p : \mathbf{El}_{\mathbf{NP}}(n)$ 的实例 n 的组合。事实上, 实体的类型已经可以被定义为一个 Σ 类型:

$$\frac{}{\Gamma \vdash \mathbf{Entity} \triangleq \sum_{n:\mathbf{NP}} \mathbf{El}_{\mathbf{NP}}(n) \text{ type}}$$

。扩展来自于 1.1 节中 $\mathbf{NP}$ 的定义, 类型为 $\mathbf{Entity}$ 的术语是其中 $b : \mathbf{Bd}$ 、 $np : \mathbf{NP}^b$ 和 $p : \mathbf{El}_{\mathbf{NP}}^b(np)$ 的元组 $((b, np), p)$ 。例如, $((U, \text{human}), \text{john}) : \mathbf{Entity}$ 是对应于名叫约翰的人的实体。

实体将在我们对形容词修饰 (第 1.5 节) 的处理和在第 2 节中对依赖事件类型的表述中发挥重要作用。

1.3 名义域中的子类型

接下来, 我们需要一种方法来断言 (例如) 每个人都可以被视为人类。我们定义了一类 $\mathbf{isA}(np, np')$ 类型, 捕获“每个 np 是 np' ”这一命题, 适用于具有任意 (特别是可能不同

的) 界限的 np, np'' 名词短语:

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma \vdash b'' : \mathbf{Bd} \quad \Gamma \vdash np'' : \mathbf{NP}^{b''}}{\Gamma \vdash \mathbf{isA}(np, np'') \text{ type}}$$

$\mathbf{isA}(np, np'')$ 的证明引入了从 np 实例的类型到 np'' 实例的类型的包含 $\mathbf{El}_{\mathbf{NP}}^b(np) \rightarrow \mathbf{El}_{\mathbf{NP}}^{b''}(np'')$ 。⁹ 我们将这种从证明到包含的映射记为 $\mathbf{El}_{\mathbf{isA}}$ ：¹⁰

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma \vdash b'' : \mathbf{Bd} \quad \Gamma \vdash np'' : \mathbf{NP}^{b''} \quad \Gamma \vdash \text{prf} : \mathbf{isA}(np, np'')}{\Gamma \vdash \mathbf{El}_{\mathbf{isA}}(\text{prf}) : \mathbf{El}_{\mathbf{NP}}^b(np) \rightarrow \mathbf{El}_{\mathbf{NP}}^{b''}(np'')}$$

Example 1.3.1. 假设我们对建模一个包含“人”，“女人”，“安”（作为“女人”的一个实例）以及“谈论”的英语片段感兴趣，后者是对（人类实例的）一个谓词。

$$\frac{}{\Gamma \vdash \text{human} : \mathbf{NP}^U} \quad \frac{}{\Gamma \vdash \text{woman} : \mathbf{NP}^U}$$

$$\frac{}{\Gamma \vdash \text{ann} : \mathbf{El}_{\mathbf{NP}}^U(\text{woman})} \quad \frac{}{\Gamma \vdash \text{talk} : \mathbf{El}^U(\text{human}) \rightarrow \mathbf{Prop}}$$

为了模拟每个女人都是人这一事实，我们进一步添加

$$\frac{}{\Gamma \vdash \text{womanIsHuman} : \mathbf{isA}(\text{woman}, \text{human})}$$

给我们一个包含 $\mathbf{El}_{\mathbf{isA}}(\text{womanIsHuman}) : \mathbf{El}_{\mathbf{NP}}^U(\text{woman}) \rightarrow \mathbf{El}_{\mathbf{NP}}^U(\text{human})$ 的关系，这使我们可以将 ann 看作 human ：

$$\text{annQuaHuman} \triangleq \mathbf{El}_{\mathbf{isA}}(\text{womanIsHuman}) \text{ ann} : \mathbf{El}_{\mathbf{NP}}^U(\text{human})$$

我们可以将命题“安说话”表述为 $\text{talk}(\text{annQuaHuman})$ 。请注意， $\text{talk}(\text{ann})$ 不是格式良好的，因为 talk 是对人类的谓词而不是对女人的。

最后，我们要求 $\mathbf{isA}$ 是自反和可传递的。¹¹

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b}{\Gamma \vdash \mathbf{isArefl}(np) : \mathbf{isA}(np, np)}$$

$$\frac{\Gamma \vdash b, b'', b''' : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma \vdash np'' : \mathbf{NP}^{b''} \quad \Gamma \vdash np''' : \mathbf{NP}^{b'''} \quad \Gamma \vdash \text{prf} : \mathbf{isA}(np, np'') \quad \Gamma \vdash \text{prf}'' : \mathbf{isA}(np'', np''')}{\Gamma \vdash \mathbf{isAtrans}(\text{prf}, \text{prf}'') : \mathbf{isA}(np, np''')}$$

⁹我们将 $\mathbf{isA}(np, np'')$ 表示为一种类型，而不是类型 $\mathbf{Prop}$ 的项，因为某些带有 $\mathbf{Prop}$ 的类型理论限制了从命题到非命题定义函数的能力（在这种情况下是 $\mathbf{El}_{\mathbf{NP}}^b(np) \rightarrow \mathbf{El}_{\mathbf{NP}}^{b''}(np'')$ ）。

¹⁰为简单起见，我们仅断言 $\mathbf{El}_{\mathbf{isA}}(\text{prf})$ 是一个函数，但还可以进一步断言它是一个单射函数。

¹¹此外，可以断言 $\mathbf{El}_{\mathbf{isA}}(\mathbf{isArefl}(np))$ 是 $\mathbf{El}_{\mathbf{NP}}^b(np)$ 上的恒等函数，并且 $\mathbf{El}_{\mathbf{isA}}(\mathbf{isAtrans}(\text{prf}, \text{prf}''))$ 是 $\mathbf{El}_{\mathbf{isA}}(\text{prf}'') \circ \mathbf{El}_{\mathbf{isA}}(\text{prf})$ ，但为了简化，我们省略了这些。

1.4 显式限定名词短语的内部结构

在第 1.1 节中，我们提到在对特定的英语片段建模时，类型 $\mathbf{NP}^U$ 将由不带修饰的名词短语填充。相反， $\mathbf{NP}^B$ 将由与名词短语对应的术语填充，这些短语表达某种限定的数量，¹² 包括诸如 3 个苹果、3 公斤的苹果、几个苹果、那些苹果、一个苹果、很少的苹果等名词短语。在介绍我们对这些名词短语的表示之前，我们必须讨论度和单位。

度量在语言学文献中被用来为可比较的形容词和名词以及度和比较结构建模。我们将只关注度量结构。我们对度和单位的处理将与 $?$ 和 $?$ 接近。然而，我们将视度量为完全原始的，正如在 $?$ 中所述；我们不认为度量是测量函数范围内的东西，正如在 $?$ 中所做的那样。具体来说，我们声明一种 **Degree** 类型的度量，它将包含诸如 height, weight, width, volume, quantity 等原始项（quantity 尤其将在瞬间起重要作用。）这是区分例如 3 个苹果和 3 公斤苹果所需要的类型，其中在前一种情况下，相关的度量是数量，而在后一种情况下，它是重量。

$$\frac{}{\Gamma \vdash \mathbf{Degree} \text{ type}}$$

$$\frac{}{\Gamma \vdash \text{quantity} : \mathbf{Degree}}$$

接下来，对于任何 $d : \mathbf{Degree}$ ，我们有一个对应于 d 的单位类型 $\mathbf{Units}(d)$ 。这使我们能够区分，例如，50 公斤的苹果和 50 克苹果。为了确保统一处理，我们采用自然单位的概念，记为 nu，如同在处理像五个苹果这样的名词短语时 $?$ 所提出的那样。单位的例子包括 meter : $\mathbf{Units}(\text{height})$ 、kilogram : $\mathbf{Units}(\text{weight})$ 、liter : $\mathbf{Units}(\text{volume})$ 、nu : $\mathbf{Units}(\text{quantity})$ 等。（nu 在框架中也将很快起到重要作用。）

$$\frac{\Gamma \vdash d : \mathbf{Degree}}{\Gamma \vdash \mathbf{Units}(d) \text{ type}}$$

$$\frac{}{\Gamma \vdash \text{nu} : \mathbf{Units}(\text{quantity})}$$

Remark 1.4.1. 请注意， $\mathbf{Units}$ 对 $\mathbf{Degree}$ 的依赖性允许我们表达每个单位仅与某些量级兼容；例如，不能用米来衡量体积，因此 meter 不是 $\mathbf{Units}(\text{volume})$ 类型的术语。相反，对我们来说， $\mathbf{Degree}$ 不依赖于被测量的名词，因此我们允许像三公斤书这样的短语。有人可能会认为 $\mathbf{Degree}$ （因此 $\mathbf{Units}$ ）应该依赖于 $\mathbf{NP}^U$ ，这样 weight 就不是 book : $\mathbf{NP}^U$ 的一个有效度量，但我们没有选择进行这样的建模。注意——尽管体积在任何情况下都不能用米来测量——诸如三公斤书这样的短语是可以理解的，例如如果一些书被细致地切碎并放入某个容器中。我们的单位概念也可以解释一些不太常规的单位，例如五杯水玻璃，以及所谓的名义分类词，这在英语中很少见（来自 $?$ 的一个例子是五头牛中的头），但在中文等语言中却很常见。然而，如果要更认真地考虑分类词，可能需要引入上述 $\mathbf{Units}$ 对 $\mathbf{NP}^U$ 的依赖性，因为数分类词通常依赖于它们所分类的名词。

最后，对于任何无界名词短语 $np : \mathbf{NP}^U$ 以及任何 $d : \mathbf{Degree}$ 、 $u : \mathbf{Units}(d)$ 和自然数 $m : \mathbf{Nat}$ ，我们引入一个界限名词短语 $\mathbf{AmountOf}(np, d, u, m) : \mathbf{NP}^B$ 来表示相应的 np 的数量。

$$\frac{\Gamma \vdash np : \mathbf{NP}^U \quad \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d) \quad \Gamma \vdash m : \mathbf{Nat}}{\Gamma \vdash \mathbf{AmountOf}(np, d, u, m) : \mathbf{NP}^B}$$

Example 1.4.2. 为了对包含名词短语“3 个成年人”和“3 公斤苹果”的英语片段进行建模，我们另外断言以下规则：使用上述框架规则，我们获得分别编码英语短语“3 个成年人”和“3 公斤苹果”的项。 $\mathbf{El}_{\mathbf{NP}^B}(\text{threeHumans})$ 类型的项是特定的 3 个成年人，例如约翰、玛丽和

¹²正如在 Section 0.2 中提到的，在缺乏冠词的语言中，单独的名词短语可能有确定的解读（与英语不同），因此除了不确定的解读外，它们还可以有确定的解读。我们将在 Section 1.4.2 中讨论无冠词语言的情况。

苏珊；同样地， $\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\text{threeKgApples})$ 类型的项是特定的 3 公斤苹果（虽然与人类不同，通常不会给特定的 3 公斤苹果实例命名）。我们不区分例如，一个苹果、这个苹果和一个苹果；前两个名词短语被建模为第三个。

1.4.1 可数名词

鉴于例子 1.1.3，我们现在有两种方法来表示特定的人类，例如约翰：作为 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ 类型的项和 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\text{human}, \text{quantity}, \text{nu}, 1))$ 类型的项。相同的方式适用于其他可数名词及修饰过的可数名词（统称为可数名词短语），但不适用于物质名词。实际上，我们将永远不会考虑物质名词的实例（例如， $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{water})$ 类型的项），因为（不像对人类而言）并不清楚应该如何看待水的一个实例。¹³

为了强制 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ 和 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\text{human}, \text{quantity}, \text{nu}, 1))$ 相同，我们首先定义一个谓词来表示无界名词短语是可数的：

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}}}{\Gamma \vdash \mathbf{isCount}(np) : \mathbf{Prop}}$$

然后，对于任何可数名词短语 np ，我们说一个 np 是 $\mathbf{AmountOf}(np, \text{quantity}, \text{nu}, 1)$ ，反之亦然。

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash prf : \mathbf{Prf}(\mathbf{isCount}(np))}{\Gamma \vdash \mathbf{NPisOneNP}(np, prf) : \mathbf{isA}(np, \mathbf{AmountOf}(np, \text{quantity}, \text{nu}, 1))}$$

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash prf : \mathbf{Prf}(\mathbf{isCount}(np))}{\Gamma \vdash \mathbf{OneNPisNP}(np, prf) : \mathbf{isA}(\mathbf{AmountOf}(np, \text{quantity}, \text{nu}, 1), np)}$$

计算名词短语也在 Σ 类型下封闭：如果 np 是一个计算名词短语，那么“满足 P 的 np 的子集”（对于任何谓词 P ）也是一个计算名词短语：

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash prf : \mathbf{isCount}(np) \quad \Gamma \vdash P : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(np) \rightarrow \mathbf{Prop}}{\Gamma \vdash \Sigma \mathbf{isCount}(np, prf, P) : \mathbf{Prf}(\mathbf{isCount}(\sum_{p:np}^{\mathbf{NP}} P(p)))}$$

在例子 ?? 中，我们说类型为 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\text{human}, \text{quantity}, \text{nu}, 3))$ 的项对应于特定的三个人类，但我们尚未引入从三个特定个体人类中推导出此类项的任何机制。为了解决这个问题，我们介绍：

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d) \quad \Gamma \vdash m, n : \mathbf{Nat} \quad \Gamma \vdash p : \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(np, d, u, m)) \quad \Gamma \vdash q : \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(np, d, u, n))}{\Gamma \vdash p \oplus q : \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(np, d, u, m + n))}$$

Example 1.4.3. 我们使用 $\oplus$ 和 $\mathbf{NPisOneNP}$ ，从两个 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ 类型的项中构造一个

¹³可以将水的读取强制为计数读取，例如“请给我一杯水”。在这种情况下，我们将一杯水表示为 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\text{water}, \text{quantity}, \text{nu}, 1))$ 类型的术语。

$\mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\mathbf{human}, \mathbf{quantity}, \mathbf{nu}, 2))$ 类型的项。假设我们有：

$$\frac{}{\Gamma \vdash \mathbf{human} : \mathbf{NP}^{\mathbf{U}}} \quad \frac{}{\Gamma \vdash \mathbf{prf} : \mathbf{Prf}(\mathbf{isCount}(\mathbf{human}))}$$

$$\frac{}{\Gamma \vdash \mathbf{john} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\mathbf{human})} \quad \frac{}{\Gamma \vdash \mathbf{mary} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\mathbf{human})}$$

注意，我们不能直接将 $\oplus$ 应用于 $\mathbf{john}$ 和 $\mathbf{mary}$ ，因为它们没有合适的类型作为 $\oplus$ 的参数。然而，注意 $\mathbf{NPisOneNP}$ （和 $\mathbf{prf} : \mathbf{Prf}(\mathbf{isCount}(\mathbf{human}))$ ）会从 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\mathbf{human})$ 引导到合适的类型。写作 $\mathbf{oneHuman} \triangleq \mathbf{AmountOf}(\mathbf{human}, \mathbf{quantity}, \mathbf{nu}, 1)$ ，

$$\frac{\frac{}{\Gamma \vdash \mathbf{human} : \mathbf{NP}^{\mathbf{U}}} \quad \frac{}{\Gamma \vdash \mathbf{prf} : \mathbf{Prf}(\mathbf{isCount}(\mathbf{human}))}}{\Gamma \vdash \mathbf{NPisOneNP}(\mathbf{human}, \mathbf{prf}) : \mathbf{isA}(\mathbf{human}, \mathbf{oneHuman})}}{\Gamma \vdash f \triangleq \mathbf{El}_{\mathbf{isA}}(\mathbf{NPisOneNP}(\mathbf{human}, \mathbf{prf})) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\mathbf{human}) \rightarrow \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{oneHuman})}$$

因此 $f(\mathbf{john}), f(\mathbf{mary}) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{oneHuman})$ ，所以我们有

$$f(\mathbf{john}) \oplus f(\mathbf{mary}) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\mathbf{human}, \mathbf{quantity}, \mathbf{nu}, 2))$$

代表约翰和玛丽。

上述论点可以推广到所有可数名词。至于不可数名词，因为我们不考虑不可数名词的实例（例如水），所以我们无法对这些实例进行求和；然而，我们可以考虑三升水的实例，并且我们可以将三升水和四升水的实例相加，从而得到七升水的实例。

1.4.2 模糊名词短语

到目前为止，我们已经在 $\mathbf{NP}^{\mathbf{B}}$ 中填充了名词短语，这些短语表达了一些 $\mathbf{NP}^{\mathbf{U}}$ 的限定数量，其中 $\mathbf{NP}^{\mathbf{U}}$ 的元素是裸名词短语。然而，正如在第 0.2 节中指出的那样，没有冠词的语言中的裸名词短语可能具有确定的解释，因此可能具有有界的解释（除了无界的解释之外），尽管它们没有明确表达一个限定数量。例如，俄语中苹果的有界理解可以指苹果，或者象征某些 n 的 n 个苹果，如 ? 中讨论的那样。建模前一种解释需要跨句关系，这是本文的研究范围之外，但对于后一种解释的建模是直接的。也就是说，我们引入了一种新的有界名词短语的形式：

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d)}{\Gamma \vdash \mathbf{several}(np, d, u) : \mathbf{NP}^{\mathbf{B}}}$$

。例如， $\mathbf{several}(\mathbf{apple}, \mathbf{quantity}, \mathbf{nu}) : \mathbf{NP}^{\mathbf{B}}$ 代表英语短语几个苹果。

为了表示“几个苹果”意味着有 n 个苹果，我们添加一个方程，指定前者的实例类型是后者的实例类型，其中一些 n 表示为 Σ 类型：

$$\frac{\Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d)}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{several}(np, d, u)) = \sum_{n:\mathbf{Nat}} \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(np, d, u, n)) \text{ type}}$$

Remark 1.4.4. 虽然对确定性的建模不在本文的范围内，我们可以使用 $\mathbf{several}$ 来近似英语中确定性名词短语的意义。例如，我们可以将“the water”（以及在无冠词的语言中与“water”对应的裸名词短语的有界解读）建模为 n 升水，其中 n 是某个值。这是一种过于简化的处

理，但如果只对终结性感兴趣，这种简化是可以接受的，因为“the water”和某个 n 的 n 升水都产生具有相同终结性状态的事件。然而，这引出一个重要问题：为什么我们选择体积和升作为上下文相关的程度和单位？一个类似的问题是——在处理无冠词语言中与“water”对应的裸名词短语时，我们如何知道是使用该词的有界还是无界表示？

我们的答案是，在形式语义中，所有的歧义必须在对自然语言的一部分进行建模之前得到解决。因此，在甚至建模类似“John drank water”这样一句无冠词语言中的句子之前，我们必须决定（例如，基于先前的对话）参与者是有界还是无界解读，在后者情况下，相关的程度和单位是什么。如果我们将“water”明确为无界解读，那么我们将其表示为 $\text{water} : \text{NP}^U$ ；如果我们反而将其明确为有界解读，以体积和升作为其程度和单位，分别表示为 $\text{several}(\text{water}, \text{volume}, \text{liter}) : \text{NP}^B$ 。

1.5 形容词修饰

在本节中，我们通过交集形容词(???)来优化 MTT-语义方法修改，特别关注形容词修饰名词短语的界限性。我们不会考虑其他种类的形容词，这些将留待未来的工作中进行研究。¹⁴

我们介绍了一种类型为 **IntAdj** 的交叉形容词，断言每一个 $\text{adj} : \text{IntAdj}$ 都引起对实体的谓词，即一个函数 **Entity** $\rightarrow$ **Prop**：

$$\frac{}{\Gamma \vdash \text{IntAdj type}} \qquad \frac{\Gamma \vdash \text{adj} : \text{IntAdj}}{\Gamma \vdash \text{El}_{\text{IA}}(\text{adj}) : \text{Entity} \rightarrow \text{Prop}}$$

形容词修饰保持有界性。也就是说，像英语中无冠词复数名词和物质名词（例如，cats）这样的无界名词短语，在被形容词修饰时（例如，black cats），仍然保持无界性；而表达某个限定数量的有界名词短语（例如，two cats），在被形容词修饰时（例如，two black cats），则仍然保持有界性。这是因为在体貌上，cats 和 black cats 都产生不成就事件，而 two cats 和 two black cats 都产生成就事件。

回忆 1.1 节，对于任何谓词 P 关于 b -名词短语 np 的实例，满足 P 的 np 的子集， $\sum_{p:np}^{\text{NP}} P$ ，再次形成一个 b -名词短语。特别地，对于任何 $b : \text{Bd}$ ， $np : \text{NP}^b$ 和交叉形容词 $\text{adj} : \text{IntAdj}$ ，我们有另一个 b -名词短语

$$np^{\cdot} \triangleq \sum_{p:np}^{\text{NP}} (\text{El}_{\text{IA}}(\text{adj}) ((b, np), p)) : \text{NP}^b$$

，其实例是 np 的实例 p 的对，伴随着一个证明，即作为 **Entity** 的 p 满足谓词 $\text{El}_{\text{IA}}(\text{adj})$ ：

$$\text{El}_{\text{NP}}^b(np^{\cdot}) = \sum_{p:\text{El}_{\text{NP}}^b(np)} \text{Prf}(\text{El}_{\text{IA}}(\text{adj}) ((b, np), p)) \text{ type}$$

Example 1.5.1. 让我们来说明如何表示（无限制的）名词短语“黑猫”。首先我们声明：

展开 El_{IA} 和 **Entity** 的定义， $\text{El}_{\text{IA}}(\text{black})$ 是一个关于 **NP** 和该 **NP** 的实例的对子谓词：

另一方面，展开 **NP** 和 El_{NP} 的定义：

通过组合， $\lambda p.\text{El}_{\text{IA}}(\text{black})((U, \text{cat}), p)$ 因此是一个关于猫的谓词：

因此，我们将黑猫建模为名词短语——在我们的框架下被自动强制为无界——对应于黑色的猫的子集：

blackCat 的一个实例是一个类型为 $\text{El}_{\text{NP}}^U(\text{blackCat})$ 的术语，它计算为：

这种类型的术语是猫 (p, prf) 和一个证据的对子 $p : \text{El}_{\text{NP}}^U(\text{cat})$ ：

证明 $\text{prf} : \text{Prf}(\text{El}_{\text{IA}}(\text{black})((U, \text{cat}), p))$ 是黑色的。例如，如果我们进一步声明：

那么 $(\text{tom}, \text{tomIsBlack}) : \text{El}_{\text{NP}}^U(\text{blackCat})$ 是黑猫的一个实例。

¹⁴关于如何在 MTT 语义中建模次属性、剥夺性和非承诺性形容词的若干建议在 ??? 中给出。

有几个交叉形容词的性质是我们希望确保被我们的框架准确建模的。我们将在示例 ?? 的背景下说明这些性质：

- (i) 一只黑猫是黑色的；
- (ii) 两只黑猫相同的条件是它们作为猫是相同的；
- (iii) 黑猫是一只猫；
- (iv) 如果每只猫都是动物，那么黑猫就是黑色的动物；
- (v) 两只黑猫就是两只黑猫；并且
- (vi) 两只黑猫是两只猫的一个黑色例子。

性质 (i) 根据我们将黑猫表示为一对由猫和相应实体是黑色的证明的方式的定义成立。性质 (ii) 来源于 $\mathbf{El}_{\mathbf{IA}}(adj)$ 是 $\mathbf{Prop}$ 值的事实：在类型理论中，当且仅当每个投影相等时，两对是相等的，并且任何类型为 $\mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{black}) e)$ 的两个项都是相等的（其中 $e : \mathbf{Entity}$ ）。

对于性质 (iii)，我们断言任何由交叉形容词修饰的名词短语实例 “ $\mathbf{isA}$ ” 都是基础名词短语的实例，如第 1.3 节中所述。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma \vdash adj : \mathbf{IntAdj}}{\Gamma \vdash \mathbf{IANPisNP}(np, adj) : \mathbf{isA} \left(\sum_{p:np}^{\mathbf{NP}} \mathbf{El}_{\mathbf{IA}}(adj) ((b, np), p), np \right)}$$

对于性质 (iv)，我们断言每当 $\mathbf{isA}(np, np^{\ddot{}})$ 成立并且我们有某个满足交叉形容词 $adj : \mathbf{IntAdj}$ 的实例 $p : \mathbf{El}_{\mathbf{NP}}^b(np)$ 时，若将其视为 $np^{\ddot{}}$ 的实例， p 也满足 adj ：

$$\frac{\Gamma \vdash b, b^{\ddot{}} : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma \vdash np^{\ddot{}} : \mathbf{NP}^{b^{\ddot{}}} \quad \Gamma \vdash \mathbf{isA} : \mathbf{isA}(np, np^{\ddot{}}) \quad \Gamma \vdash adj : \mathbf{IntAdj} \quad \Gamma \vdash p : \mathbf{El}_{\mathbf{NP}}^b(np) \quad \Gamma \vdash \mathit{prf} : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(adj) ((b, np), p))}{\Gamma \vdash \mathbf{IARespectsIsA}(\mathbf{isA}, adj, p, \mathit{prf}) : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(adj) ((b^{\ddot{}}, np^{\ddot{}}), \mathbf{El}_{\mathbf{isA}}(\mathbf{isA}) p))}$$

Remark 1.5.2. 上述规则是我们在第 1.3 节引入 $\mathbf{isA}$ 的原因，而不是简单地通过存在一个函数 $\mathbf{El}_{\mathbf{NP}}^b(np) \rightarrow \mathbf{El}_{\mathbf{NP}}^{b^{\ddot{}}}(np^{\ddot{}})$ 来建模每个 np 是一个 $np^{\ddot{}}$ 。因为在名词短语实例的类型之间有太多可定义的函数，所以说交集形容词遵循任意函数是过于笼统的。在类型理论中，我们也不能直接说 $\mathbf{El}_{\mathbf{NP}}^b(np)$ 是 $\mathbf{El}_{\mathbf{NP}}^{b^{\ddot{}}}(np^{\ddot{}})$ 的“子集”。例如，给定 $\text{john} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ ，我们可以定义一个常量函数 $\lambda a. \text{john} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat}) \rightarrow \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ 。如果我们将 $\mathbf{isA}(np, np^{\ddot{}})$ 定义为类型 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat}) \rightarrow \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{human})$ ，那么上述 $\mathbf{IARespectsIsA}$ 的规则将意味着一只黑猫是一个黑色的人类。

Example 1.5.3. 为了说明性质 (iii) 和 (iv)，我们从条纹（黑猫）的实例推导出到条纹猫的实例的一个函数。我们在我们的框架中加入：

$$\frac{}{\Gamma \vdash \text{cat} : \mathbf{NP}^{\mathbf{U}}} \quad \frac{}{\Gamma \vdash \text{black} : \mathbf{IntAdj}} \quad \frac{}{\Gamma \vdash \text{striped} : \mathbf{IntAdj}}$$

根据示例 ??，我们将黑猫、条纹黑猫和条纹猫建模如下：

$$\begin{aligned} \text{blackCat} &\triangleq \sum_{p:\text{cat}}^{\mathbf{NP}} (\mathbf{El}_{\mathbf{IA}}(\text{black}) ((\mathbf{U}, \text{cat}), p)) : \mathbf{NP}^{\mathbf{U}} \\ \text{stripedCat} &\triangleq \sum_{p:\text{cat}}^{\mathbf{NP}} (\mathbf{El}_{\mathbf{IA}}(\text{striped}) ((\mathbf{U}, \text{cat}), p)) : \mathbf{NP}^{\mathbf{U}} \\ \text{stripedBlackCat} &\triangleq \sum_{p:\text{blackCat}}^{\mathbf{NP}} (\mathbf{El}_{\mathbf{IA}}(\text{striped}) ((\mathbf{U}, \text{blackCat}), p)) : \mathbf{NP}^{\mathbf{U}} \end{aligned}$$

根据 **IANPIsNP** 规则，我们得出每只黑猫都是一只猫：

$$\text{bclsC} \triangleq \mathbf{IANPIsNP}(\text{cat}, \text{black}) : \mathbf{isA}(\text{blackCat}, \text{cat})$$

将 **IARespectsIsA** 规则应用于“每只黑猫都是一只猫”以及形容词“条纹”，对于任何 $p : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{blackCat})$ 和证明 $\text{prf} : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{striped})((\mathbf{U}, \text{blackCat}), p))$ 表明 p 是作为黑猫的条纹，我们同样有一个证明表明 $\mathbf{El}_{\mathbf{isA}}(\text{bclsC}) p : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat})$ 是作为猫的条纹。

$$\mathbf{IARespectsIsA}(\text{bclsC}, \text{striped}, p, \text{prf}) : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{striped})((\mathbf{U}, \text{cat}), \mathbf{El}_{\mathbf{isA}}(\text{bclsC}) p))$$

为了构建一个函数 $\mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{stripedBlackCat}) \rightarrow \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{stripedCat})$ ，我们必须以一对某个 $p : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{blackCat})$ 和一个证明 $\text{prf} : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{striped})((\mathbf{U}, \text{blackCat}), p))$ (表明 p 是作为黑猫的条纹) 作为输入，并输出一对某个 $p'' : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat})$ 和一个证明 $\text{prf}'' : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{striped})((\mathbf{U}, \text{cat}), p''))$ (表明 p'' 是作为猫的条纹)。我们如下述方式取 $p'' \triangleq \mathbf{El}_{\mathbf{isA}}(\text{bclsC}) p$ ，并取 $\text{prf}'' \triangleq \mathbf{IARespectsIsA}(\text{bclsC}, \text{striped}, p, \text{prf})$ 。

性质 (v) — 两只黑猫就是两只黑猫 — 源自于第 1.4 节中已经规定的几个可数名词短语的性质，如下面的例子所示。

Example 1.5.4. 为了说明性质 (v)，我们展示如何从“黑猫”的两个实例推导出“两只黑猫”的一个实例。让我们继续例子 ??，其中我们将“黑猫”定义如下：

$$\text{blackCat} \triangleq \sum_{p:\text{cat}}^{\mathbf{NP}} (\mathbf{El}_{\mathbf{IA}}(\text{black})((\mathbf{U}, \text{cat}), p)) : \mathbf{NP}^{\mathbf{U}}$$

并表明 $(\text{tom}, \text{tomIsBlack}) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{blackCat})$ 是一只黑猫，其中 $\text{tom} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat})$ 和 $\text{tomIsBlack} : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{black})((\mathbf{U}, \text{cat}), \text{tom}))$ 。如果我们进一步假设

$$\frac{}{\Gamma \vdash \text{felix} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{cat})} \quad \frac{}{\Gamma \vdash \text{felixIsBlack} : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{black})((\mathbf{U}, \text{cat}), \text{felix}))}$$

那么我们有第二个“黑猫”的实例 $(\text{felix}, \text{felixIsBlack}) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{blackCat})$ 。如果我们通过添加以下规则将“猫”建模为一个计数名词短语：

$$\frac{}{\text{catIsCount} : \mathbf{Prf}(\mathbf{isCount}(\text{cat}))}$$

那么根据计数名词短语在 Σ 型下的闭包性质 ($\Sigma\mathbf{IsCount}$ 源自章节 1.4)，可以得出“黑猫”也是一个计数名词短语：

$$\text{prf} \triangleq \Sigma\mathbf{IsCount}(\text{cat}, \text{catIsCount}, \lambda p. \mathbf{El}_{\mathbf{IA}}(\text{black})((\mathbf{U}, \text{cat}), p)) : \mathbf{Prf}(\mathbf{isCount}(\text{blackCat}))$$

然后，根据任意计数名词短语 np ($\mathbf{NPisOneNP}$ 来自章节 1.4) 的一个 np 是一个 $\mathbf{AmountOf}(np, \text{quantity}, \text{nu}, 1)$ 的事实，我们有一个函数 f ，将“黑猫”的实例转换为“一只黑猫”的实例：

$$\text{isa} \triangleq \mathbf{NPisOneNP}(\text{blackCat}, \text{prf}) : \mathbf{isA}(\text{blackCat}, \mathbf{AmountOf}(\text{blackCat}, \text{quantity}, \text{nu}, 1))$$

$$f \triangleq \mathbf{El}_{\mathbf{isA}}(\text{isa}) : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\text{blackCat}) \rightarrow \mathbf{El}_{\mathbf{NP}}^{\mathbf{B}}(\mathbf{AmountOf}(\text{blackCat}, \text{quantity}, \text{nu}, 1))$$

给定“两只黑猫”的两个实例，我们可以通过 $\oplus$ 轻松获得“两只黑猫”的一个实例：

$$f(\text{tom}, \text{tomIsBlack}) \oplus f(\text{felix}, \text{felixIsBlack}) : \mathbf{AmountOf}(\text{blackCat}, \text{quantity}, \text{nu}, 2)$$

请注意，我们的框架自动确定“两只黑猫”是有界的：一个 $\mathbf{AmountOf}$ 一个 $\mathbf{NP}^{\mathbf{U}}$ (例如黑猫) 是一个 $\mathbf{NP}^{\mathbf{B}}$ 。

最后，对于性质 (vi)---两只黑猫是两只猫的一个黑色实例---我们断言 $\oplus$ 保持了交集形容词。虽然这个规则看起来令人生畏，但它只是简单地陈述：如果我们有一个 $m np$ 的实例 p ，第二个实例 n 的 q 也是同类 np （在相同的程度和单位），并且两个 p, q 都满足某个交集形容词 adj ，那么 $p \oplus q$ 也满足 adj ：

$$\frac{\begin{array}{l} \Gamma \vdash np : \mathbf{NP}^U \quad \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d) \quad \Gamma \vdash m, n : \mathbf{Nat} \\ \Gamma \vdash p : \mathbf{El}_{\mathbf{NP}}^B(\mathbf{AmountOf}(np, d, u, m)) \quad \Gamma \vdash q : \mathbf{El}_{\mathbf{NP}}^B(\mathbf{AmountOf}(np, d, u, n)) \\ \Gamma \vdash adj : \mathbf{IntAdj} \quad \Gamma \vdash prf : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(adj) ((\mathbf{B}, \mathbf{AmountOf}(np, d, u, m)), p)) \\ \Gamma \vdash prf'' : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(adj) ((\mathbf{B}, \mathbf{AmountOf}(np, d, u, n)), q)) \end{array}}{\Gamma \vdash \oplus \mathbf{PreservesIA}(prf, prf'') : \mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(adj) ((\mathbf{B}, \mathbf{AmountOf}(np, d, u, m+n)), p \oplus q))}$$

Example 1.5.5. 为了说明性质 (vi)，我们从例子 1.5.4 继续，勾勒出如何从黑色的猫 Tom 和 Felix 获取两个猫的一个黑色实例。像往常一样，这样的实例是两个猫的一个实例的对偶 $\mathbf{AmountOf}(\text{cat}, \text{quantity}, \text{nu}, 2) : \mathbf{NP}^B$ 以及提供给定实例是黑色的一个证明。因为我们规定猫是一个可数名词短语，所以任何猫都是一只猫：特别地， $\text{tom} : \mathbf{El}_{\mathbf{NP}}^U(\text{cat})$ 可以被视为一只猫：

$$\mathbf{El}_{\mathbf{isA}}(\text{isa}) \text{ tom} : \mathbf{El}_{\mathbf{NP}}^B(\mathbf{AmountOf}(\text{cat}, \text{quantity}, \text{nu}, 1))$$

同样适用于 $\mathbf{El}_{\mathbf{isA}}(\text{isa}) \text{ felix}$ 。此外，因为交叉形容词遵循 $\mathbf{isA}$ ，我们知道 tom 作为一只猫也是黑色的（同样适用于 felix）：

$$\mathbf{IARespectsIsA}(\text{isa}, \text{black}, \text{tom}, \text{tomIsBlack}) :$$

$$\mathbf{Prf}(\mathbf{El}_{\mathbf{IA}}(\text{black}) ((\mathbf{B}, \mathbf{AmountOf}(\text{cat}, \text{quantity}, \text{nu}, 1)), \mathbf{El}_{\mathbf{isA}}(\text{isa}) \text{ tom}))$$

将这些部分组合起来，我们获得两个猫的一个实例，作为 tom 作为一只猫与 felix 作为一只猫的总和，

$$(\mathbf{El}_{\mathbf{isA}}(\text{isa}) \text{ tom}) \oplus (\mathbf{El}_{\mathbf{isA}}(\text{isa}) \text{ felix}) : \mathbf{El}_{\mathbf{NP}}^B(\mathbf{AmountOf}(\text{cat}, \text{quantity}, \text{nu}, 2))$$

并通过将 $\oplus \mathbf{PreservesIA}$ 应用于之前构造的两个证明，得出上述实例为黑色的证明，证明 tom（分别 felix）作为一只猫是黑色的。

2 动词域

在已经解释了名词域中的有界性和无界性后，我们将注意力转向我们工作的主要焦点：动词域中的终止性、非终止性和达到性。我们对终止性的处理依赖于事件语义学的一种变体，在这种变体中，我们将事件本身视为类型，即它们出现的集合。这与之前将事件语义学作为类型理论方法的方式不同，其中事件不是作为类型处理，而是作为新达维兹事件语义学中的变量或者作为命题的证明对象。

我们并不是使用单一类型的事件宇宙，而是根据 $\mathbf{?}$ ，考虑一组以两个语义宏角色索引的依赖事件类型族，我们称之为施事者和承事者。在第 2.1 节中，我们定义了施事者和承事者的类型，其中承事者的类型进一步由界限性进行索引。在第 2.2 节中，我们定义了事件的类型，以及一个将每个事件发送到其出现类型的类型族。在第 2.3 节中，我们展示了如何通过我们的依赖事件类型组合性地确定事件的（非）终止性，这是依据事件的承事者类型进行确定的。

在第 2.4 节中，我们介绍了状态的概念，这是事件的静态对应部分。在第 2.5 节中，我们将结果状态与每个结果性事件相关联，并将达到顶点的事件定义为一旦发生就会获得其结

果状态的结果性事件。在第 2.6 节中，我们讨论了如何将事件视为类型，从而使我们能够容易地根据 ? 模型副词修改。最后，在第 2.7 节中，我们讨论了如何对前几节未覆盖的事件之间的各种蕴涵进行建模。

我们注意到，我们的事件语义方法除了能够解释与终止性和完成性相关的问题之外，还提供了一个关于事件及其结构的本体论。在类型理论的背景下，这个领域的研究很少。（我们所知道的唯一其他工作是 ? 的研究。）这种处理方式也可以看作是词汇分解动词的类型论类比，正如 ??? 中所研究的那样。我们类型论方法的好处是，它为我们开发集合论语义提供了一条途径（见第 ?? 节），而非类型论的词汇分解研究完全不提供任何语义。

我们的事件语义保留了新戴维森事件语义的主要优点（特别是对状语修饰及其相关蕴涵的直接模型），同时避免了其一个主要不悦目的特征：在每个句子的表示中都需要存在量词。通常这是通过在动词的词汇语义中包括存在量词或使用存在闭包规则来实现的，这就像事件语义旨在克服的包含状语修饰的蕴涵的意义公设一样随意。新戴维森事件语义中存在量词的必要性也引发了著名的事件量化问题(???)，而由于量词的缺失，这一问题在我们的情形中并未出现。尽管这并不是我们方法的独有特征，我们也同样避免了新戴维森事件语义中产生的非组合性问题，如在 ?? 中讨论的那样。

2.1 预备知识

在我们的框架中，我们将通过两个参数（参与者和承受者）来索引可能性的类型（特别是事件，正如 Remark ?? 中所讨论的那样，事件是动态的可能性）。如第 0.2 节所讨论的，参与者是事件中最具代理性的一方，而承受者是最具受动性的一方。例如，在句子 John ate apples 中，John 是参与者而苹果是承受者。

可能有人会倾向于假设行动者和接受者的类型都是 NP 类型，但这样有三个问题。首先，像 John 这样的实体（参见章节 1.2）可以作为行动者或接受者，但在我们的框架中，它们不是名词短语（另请参见备注 1.1.1）。其次，一些事件（例如，那些由不及物动词描述的事件）缺乏行动者或接受者，例如 John died（缺乏行动者）或 John ran（缺乏接受者）。最后，我们希望使用类型依赖性来追踪接受者的界限性，因为这决定了事件的终结性。

考虑到这些问题，我们将演员的类型 Act 声明为名词短语类型 NP^b （对于任何 $b : \text{Bd}$ ）、实体类型 Entity 和一个“虚拟演员” act_* 的不相交联合：

$$\frac{}{\Gamma \vdash \text{Act type}} \quad \frac{\Gamma \vdash b : \text{Bd} \quad \Gamma \vdash np : \text{NP}^b}{\Gamma \vdash \text{act}_{\text{NP}}(np) : \text{Act}} \quad \frac{\Gamma \vdash e : \text{Entity}}{\Gamma \vdash \text{act}_{\text{Entity}}(e) : \text{Act}} \quad \frac{}{\Gamma \vdash \text{act}_* : \text{Act}}$$

Example 2.1.1. 以下三个句子例示了三种类型的行为者：

三位女士把气球戳破了。 。_ 约翰把气球戳破了。 。_ 气球爆了。假设相关常量已被添加到理论中，2.1.1 的行为者是

$$\text{act}_{\text{NP}}(\text{AmountOf}(\text{woman}, \text{quantity}, \text{nu}, 3)) : \text{Act}$$

，2.1.1 的行为者是 $\text{act}_{\text{Entity}}((U, \text{human}), \text{john}) : \text{Act}$ ，以及 2.1.1 的行为者是虚拟行为者 $\text{act}_* : \text{Act}$ 。

对于承受者，对于任何 $b : \text{Bd}$ ，我们引入一个类型 Und^b ，称为“具有边界 b 的承受者”。类型 Und^{B} 的术语是有界承受者，而类型 Und^{U} 的术语是无界承受者。

$$\frac{\Gamma \vdash b : \text{Bd}}{\Gamma \vdash \text{Und}^b \text{ type}}$$

仍然有三种来源的承受者——名词短语、实体和虚拟承受者——但现在我们必须指定每种承受者的界限性。 b -名词短语产生 b -承受者，实体产生有界承受者，而虚拟承受者是无界承受者：

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b}{\Gamma \vdash \mathbf{und}_{\mathbf{NP}}(np) : \mathbf{Und}^b \text{ type}} \quad \frac{\Gamma \vdash e : \mathbf{Entity}}{\Gamma \vdash \mathbf{und}_{\mathbf{Entity}}(e) : \mathbf{Und}^{\mathbf{B}}} \quad \frac{}{\Gamma \vdash \mathbf{und}_* : \mathbf{Und}^{\mathbf{U}}}$$

Example 2.1.2. 考虑以下四个句子：

三个人死亡。。约翰死了。。气球(连续数小时)爆裂。。约翰跑了。假设相关的常量已添加到理论中，2.1.2有一个有限的施受者 $\mathbf{und}_{\mathbf{NP}}(\mathbf{AmountOf}(\mathbf{human}, \mathbf{quantity}, \mathbf{nu}, 3)) : \mathbf{Und}^{\mathbf{B}}$ ，因为三个人是一个有限的名词短语；2.1.2有一个有限的施受者 $\mathbf{und}_{\mathbf{Entity}}((\mathbf{U}, \mathbf{human}), \mathbf{john}) : \mathbf{Und}^{\mathbf{B}}$ ，因为所有实体都产生有限的施受者。无限的施受者包括无限的名词短语，如 $\mathbf{und}_{\mathbf{NP}}(\mathbf{balloon}) : \mathbf{Und}^{\mathbf{U}}$ 在 2.1.2 中，以及在 2.1.2 中的虚拟施受者 $\mathbf{und}_* : \mathbf{Und}^{\mathbf{U}}$ 。

如同在第 1.1 节中的 $\mathbf{NP}$ ，我们可以定义一种具有任意界限的承受者类型：

$$\frac{}{\Gamma \vdash \mathbf{Und} \triangleq \sum_{b:\mathbf{Bd}} \mathbf{Und}^b \text{ type}}$$

因为施事者和受事者的存在纯粹是为了指示事件的类型，在解释我们的框架如何建模事件之后，它们定义中的一些细微差别可以得到最佳说明。我们将在第 2.2 节中解释虚拟施事者和受事者的目的；在第 2.3 节中，我们将解释为什么实体受事者是有界的而虚拟受事者是无界的。

2.2 事件与蕴含

有了各种类型的施事者和受事者，我们现在声明事件类型 $\mathbf{Evt}^b(a, und)$ ，施事者 $a : \mathbf{Act}$ 和 b -受事者 $und : \mathbf{Und}^b$ （其中 $b : \mathbf{Bd}$ ）。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^b}{\Gamma \vdash \mathbf{Evt}^b(a, und) \text{ type}}$$

事件的例子包括例句 2.1.1 和 2.1.2。

正如我们为每一个名词短语 $np : \mathbf{NP}^b$ 关联一个其实例的类型 $\mathbf{El}_{\mathbf{NP}}^b(np)$ 一样，我们将为每一个事件 $evt : \mathbf{Evt}^b(a, und)$ 关联一个其发生的类型 $\mathbf{El}_{\mathbf{Evt}}(e)$ 。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^b \quad \Gamma \vdash evt : \mathbf{Evt}^b(a, und)}{\Gamma \vdash \mathbf{El}_{\mathbf{Evt}}(evt) \text{ type}}$$

在我们的框架中，类型 $\mathbf{El}_{\mathbf{Evt}}(evt)$ 扮演着两个重要角色。首先，它为我们提供了一种方法来表示事件 $evt : \mathbf{Evt}^b(a, und)$ 是否已经发生，即 $\mathbf{El}_{\mathbf{Evt}}(evt)$ 是非空的（至少有一个元素）。其次，将事件视为类型而不仅仅是命题，将使我们在第 2.6 节中能够解释以 ? 中建议的精神进行的状态修饰。

Example 2.2.1. 为了对事件“John 吃苹果”进行建模，我们的框架必须至少包括在第 ?? 节中之前考虑的以下几个常量：

$$\frac{}{\Gamma \vdash \mathbf{human} : \mathbf{NP}^{\mathbf{U}}} \quad \frac{}{\Gamma \vdash \mathbf{john} : \mathbf{El}_{\mathbf{NP}}^{\mathbf{U}}(\mathbf{human})} \quad \frac{}{\Gamma \vdash \mathbf{apple} : \mathbf{NP}^{\mathbf{U}}}$$

我们可以将及物动词“吃”建模为事件的从属积，如下所示：

$$\frac{}{\Gamma \vdash \text{eat} : (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(a, \mathbf{snd}(u))}$$

根据第 2.1 节中的行为者和承受者规则，我们有

$$\begin{aligned} \text{john}_{\mathbf{Act}} &\triangleq \mathbf{act}_{\mathbf{Entity}}((U, \text{human}), \text{john}) : \mathbf{Act} \\ \mathbf{und}_{\mathbf{NP}}(\text{apple}) &: \mathbf{Und}^U \end{aligned}$$

因此术语

$$\text{jaa} \triangleq \text{eat } \text{john}_{\mathbf{Act}} (U, \mathbf{und}_{\mathbf{NP}}(\text{apple})) : \mathbf{Evt}^U(\text{john}_{\mathbf{Act}}, \mathbf{und}_{\mathbf{NP}}(\text{apple}))$$

表示事件“John 吃苹果”。类型 $\mathbf{El}_{\mathbf{Evt}}(\text{jaa})$ 然后表示 John 吃苹果事件的实例或发生的集合，即 John 吃苹果在特定时间和地点的多次事件。

每个事件都是类型 $\mathbf{Evt}^b(a, \mathbf{und})$ 的项这一事实似乎表明每个事件都有一个施事者和一个受事者，但当然自然语言中的句子往往缺少它们中的一个，要么是因为事件本身没有施事者或受事者，要么是因为施事者或受事者被省略。在前一种情况下，我们用虚拟项 $\mathbf{act}_\star : \mathbf{Act}$ 和 $\mathbf{und}_\star : \mathbf{Und}^U$ 来模拟施事者或受事者的缺失。

Example 2.2.2. 考虑句子 John died (例子 2.1.1)，其没有施事，以及 John ran (例子 2.1.2)，其没有受事。由于动词 die 从来没有施事，我们在我们的框架中通过以下事件的依赖乘积来对其建模，这可以有任意的受事但总是有虚拟施事 $\mathbf{act}_\star$ ：

$$\frac{}{\Gamma \vdash \text{die} : (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(\mathbf{act}_\star, \mathbf{snd}(u))}$$

同样地，动词 run 可以有任意的施事但总是有虚拟受事 $\mathbf{und}_\star$ ：

$$\frac{}{\Gamma \vdash \text{run} : (a : \mathbf{Act}) \rightarrow \mathbf{Evt}^U(a, \mathbf{und}_\star)}$$

因此，我们将句子 John died 和 John ran 建模为以下事件：

$$\begin{aligned} \text{die} (B, \mathbf{und}_{\mathbf{Entity}}((U, \text{human}), \text{john})) &: \mathbf{Evt}^B(\mathbf{act}_\star, \mathbf{und}_{\mathbf{Entity}}((U, \text{human}), \text{john})) \\ \text{run} (\mathbf{act}_{\mathbf{Entity}}((U, \text{human}), \text{john})) &: \mathbf{Evt}^U(\mathbf{act}_{\mathbf{Entity}}((U, \text{human}), \text{john}), \mathbf{und}_\star) \end{aligned}$$

Remark 2.2.3. 为了避免虚拟施事和受事，可以通过添加没有施事（或受事）的原始事件类型来实现，然后将所有事件的类型定义为以下不相交的合集：具有指定施事和受事的事件，具有指定施事但没有受事的事件，以及没有施事但具有指定受事的事件。这样的处理将更为复杂，而收益甚微。此外，正如我们将在例子 2.2.5 中指出的，虚拟施事和受事为我们提供了一种方便的方法来模拟诸如“pop”这样的动词，这些动词既可以用作及物动词，也可以用作不及物动词。

句子中可能缺少施事者或受事者的另一个原因是，尽管句子中存在这一施事者或受事者，它可能被省略。例如在句子“John ate”中，必定是 John 吃了某些东西，但具体吃了什么并未明确说明。

不仅是承受者，行动者也可以不明确。展示所谓因果-非因果交替的动词提供了例子。例如，根据采用的视角不同，句子“气球爆炸了”可以被解释为描述了一个没有行动者或行动

者不明确的事件。第一个观点由句子的语法暗示——没有明显的行动者——但也有解释句子为描述一个行动者不明确的事件的理由。首先，气球不能自己爆炸；一定有某种外部原因导致气球的爆炸，不论是环境条件（在这种情况下，根据我们的术语，环境可以被看作是一个行动者，因为“行动者”的概念包含了所有“施动性”的语义角色）还是某个有意识的行动者使其爆炸。此外，如果有人看到，例如约翰在爆气球，他们可以恰当地使用“气球爆炸了”这个句子。在这种情况下，说话人出于某种原因选择压制行动者的信息（但他们仍然知道谁是行动者）。这对应于句子“有人爆了气球”的特定解读，其中“特定”意味着说话人知道那个“有人”指的是约翰。

我们可以使用 Σ 类型来说明未指定的行为者或承受者。例如，我们定义事件 $\mathbf{Evt}_A(a)$ 的类型，其中行为者 $a : \mathbf{Act}$ 已被指定但承受者未被指定，由一个承受者 $u : \mathbf{Und}$ 和一个具有行为者 a 和承受者 u 的事件对组成。同样，我们定义事件 $\mathbf{Evt}_{\mathbf{Und}}(und)$ 的类型，其中承受者 $und : \mathbf{Und}^b$ 已被指定但行为者未被指定，甚至定义事件 $\mathbf{Evt}$ 的类型，其行为者和承受者均未被指定。注意，未指定的行为者/承受者可以是但不需要是虚拟的行为者/承受者，因此这些类型不是 Remark 2.2.3 中讨论的那些。（参见 Example 2.2.5 和 Remark 2.2.6，以获得更多关于未指定和虚拟行为者/承受者之间区别的讨论。）

Σ 类型的规则允许我们定义可以抑制行为者和/或承受者的函数，例如

$$\lambda \text{evt}.((b, und), \text{evt}) : \mathbf{Evt}^b(a, und) \rightarrow \mathbf{Evt}_A(a)$$

。此外，通过将第二投影

$$\mathbf{snd} : (\text{evt} : \mathbf{Evt}_A(a)) \rightarrow \mathbf{Evt}^{\mathbf{fst}(\mathbf{fst}(\text{evt}))}(a, \mathbf{snd}(\mathbf{fst}(\text{evt})))$$

与先前断言的 $\mathbf{El}_{\mathbf{Evt}}$ 后组合，我们可以将其出现的类型 $\mathbf{El}_{\mathbf{Evt}_A}(\text{evt}_A)$ 与每个 $\text{evt}_A : \mathbf{Evt}_A(a)$ 关联起来。我们同样定义 $\mathbf{El}_{\mathbf{Evt}_{\mathbf{Und}}}^b(\text{evt}_{\mathbf{Und}}) \triangleq \mathbf{El}_{\mathbf{Evt}}(\mathbf{snd}(\text{evt}_{\mathbf{Und}}))$ 。

Example 2.2.4. 继续从例子 2.2.1 推进，我们可以将事件约翰吃作为类型 $\mathbf{Evt}_A(\text{john}_{\mathbf{Act}})$ 的一个项来建模。这样的项是一个包含一个承受者 $u : \mathbf{Und}$ 和一个涉及实体约翰和承受者 u 的事件 $\text{evt} : \mathbf{Evt}^{\mathbf{fst}(u)}(\text{john}_{\mathbf{Act}}, \mathbf{snd}(u))$ 的对 (u, evt) 。（这里我们忽略动词“吃”对其承受者类型有一些限制的事实，因为建模这种选择限制并不是本文的重点。）

因为一个没有明确承受者的事件 $(u, \text{evt}) : \mathbf{Evt}_A(\text{john}_{\mathbf{Act}})$ 的发生类型 $\mathbf{El}_{\mathbf{Evt}_A}(u, \text{evt})$ 被定义为其具有明确承受者的基础事件 $\mathbf{El}_{\mathbf{Evt}}(\text{evt})$ 的发生类型，所以“约翰吃”的发生集合正是约翰吃了某物的发生集合，也就是说，这个特定的约翰吃了某个承受者的发生集合。

重新表达示例 2.2.4 结尾的观察，如果 John 吃了苹果这件事是真的，那么 John 吃了这个事实也是真的，但反过来则不一定成立。因此，我们在涉及像“吃”这样的动词的句子之间自动有单向蕴涵，这些动词可能描述事件一带有未指定的承受者。

Example 2.2.5. 我们现在考虑一个因果交替的例子，即 John popped balloons 与 Balloons popped，以及相关的蕴涵关系。正如在本节早些时候讨论的那样，像 Balloons popped 这样的句子可以被理解为描述一个没有演员或者演员未指明的事件。在前一种情况下，句子会被解释为气球自己爆了，没有任何东西或人导致这种结果，而在后一种情况下，句子会被解释为气球炸了，事实上，是某人/某物炸了它们（这一某人/某物是用来特指的——即，说话者知道）。因此，在前一种情况下，John popped balloons $\Rightarrow$ Balloons popped 的蕴涵不应该存在（因为 John 把气球炸了与气球自己炸了是矛盾的），而在后一种情况下，只有当说话者心目中的那个人是 John，蕴涵才成立。

现在我们展示我们的理论如何解释这一点。为了对这些事件建模，我们假设：

$$\frac{}{\Gamma \vdash \text{human} : \mathbf{NP}^U} \quad \frac{}{\Gamma \vdash \text{john} : \mathbf{El}_{\mathbf{NP}}^U(\text{human})} \quad \frac{}{\Gamma \vdash \text{balloon} : \mathbf{NP}^U}$$

$$\frac{}{\Gamma \vdash \text{pop} : (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\mathbf{fst}(u)}(a, \mathbf{snd}(u))}$$

因为我们对虚拟承受者的处理，pop 代表了英语动词 pop 的及物和不及物两种形式。再次书写

$$\text{john}_{\text{Act}} \triangleq \text{act}_{\text{Entity}}((U, \text{human}), \text{john}) : \text{Act}$$

我们可以表示 John popped balloons 为

$$\text{evt}_1 \triangleq \text{pop john}_{\text{Act}}(U, \text{und}_{\text{NP}}(\text{balloon})) : \text{Evt}^U(\text{john}_{\text{Act}}, \text{und}_{\text{NP}}(\text{balloon}))$$

我们可以表示 Balloons popped（实际上，是 John 炸了它们）为

$$\text{evt}_2 \triangleq (\text{john}_{\text{Act}}, \text{pop john}_{\text{Act}}(U, \text{und}_{\text{NP}}(\text{balloon}))) : \text{Evt}_{\text{Und}}^U(\text{und}_{\text{NP}}(\text{balloon}))$$

最后，我们可以表示 Balloons popped by themselves，没有任何东西或人导致它们炸为

$$\text{evt}_3 \triangleq (\text{act}_{\star}, \text{pop act}_{\star}(U, \text{und}_{\text{NP}}(\text{balloon}))) : \text{Evt}_{\text{Und}}^U(\text{und}_{\text{NP}}(\text{balloon}))$$

注意 evt_2 和 evt_3 具有相同的类型 $\text{Evt}_{\text{Und}}^U(\text{und}_{\text{NP}}(\text{balloon}))$ 因为它们都是演员未指定的事件，气球为承受者，但在第一种情况下，演员是 John，而在第二种情况下，演员是虚拟演员。

在我们的框架中，我们可以通过一个函数类型来表达蕴涵 $\text{John popped balloons} \Rightarrow \text{Balloons popped}$ （实际上，是 John 炸了它们）。 $\text{El}_{\text{Evt}}(\text{evt}_1) \rightarrow \text{El}_{\text{Evt}_{\text{Und}}^U}(\text{evt}_2)$ 。（这种函数在标准命题作为类型的解释中证实了一种蕴涵；实际上，如果至少存在一个 evt_1 的出现，那么通过这样一个函数就至少存在一个 evt_2 的出现。）在这种情况下，因为 John 是 evt_2 中被忽略的参与者，所以一旦我们展开定义，这两个在蕴涵中的类型是相同的，因此恒等函数足以确立该蕴涵，以及反向蕴涵 $\text{El}_{\text{Evt}_{\text{Und}}^U}(\text{evt}_2) \rightarrow \text{El}_{\text{Evt}}(\text{evt}_1)$ 。然而，蕴涵 $\text{John 戳破了气球} \Rightarrow \text{气球自己爆了}$ ，没有任何东西或人导致这一点， $\text{El}_{\text{Evt}}(\text{evt}_1) \rightarrow \text{El}_{\text{Evt}_{\text{Und}}^U}(\text{evt}_3)$ ，在我们的框架中不成立，因为这两种类型之间没有关系。我们也没有 $\text{John 戳破了气球} \Rightarrow \text{气球爆了}$ （事实上，Mary 击破了它们）。

Remark 2.2.6. 我们如何确定句子“气球爆破”是指有人爆破气球还是气球自行爆破——也就是说，句子中是有一个虚假受事者还是一个被抑制的非虚假受事者？如在 Remark ?? 中一样，我们的答案是，必须解决是谁或什么使气球爆破的歧义，因为通常情况下，任何句子在建模之前都必须被消除歧义。

2.3 终止事件与非终止事件

在此，我们可以简单地定义完结事件和未完事件的类型。回忆一下 0.1 节中提到，完结事件是具有内在终点（或者等价地，自然关联的结果状态）的事件，例如“约翰喝了汤”。如 0.2 节所讨论，我们将关注点限制在内在终点来自于承受者（而非附加成分）的情况，因此我们说，事件是完结的（相应地，未完的）当且仅当其承受者是有限的（相应地，无限的）。

因为我们这种类型的事件已经跟踪其承受者的界限性，所以我们可以简单地将具有参与者 $a : \text{Act}$ 和有界承受者 $\text{und} : \text{Und}^B$ 的终结事件类型 $\text{Tel}(a, \text{und})$ 定义为事件类型 $\text{Evt}^B(a, \text{und})$ ，对于非终结事件类型也是类似的：使用 Σ 类型，我们可以再次定义在没有或只有一个参与者或承受者被指定的情况下的终结/非终结事件类型： $\text{Tel}_A(a)$ ， $\text{Tel}_{\text{Und}}(\text{und})$ ， Tel ， $\text{Atel}_A(a)$ ， $\text{Atel}_{\text{Und}}(\text{und})$ ，和 Atel 。

回忆一下从第 2.1 节开始，有三种承受者：名词短语，其界限取决于基础名词短语的界限；实体，它们总是有界的；以及虚拟承受者，它总是无界的。现在我们依次重新考虑这些承受者。

Example 2.3.1. 重新审视 John ate apples 这个例子（例子见 2.2.1），记得 eat 的类型是：

$$\Gamma \vdash \text{eat} : (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(a, \mathbf{snd}(u))$$

，而 apple 是一个无界名词短语，因此是 $\mathbf{und}_{\text{NP}}(\text{apple}) : \mathbf{Und}^{\text{U}}$ 。事件 John ate apples 因此被建模为一个非终止事件

$$\text{eat john}_{\text{Act}}(\text{U}, \mathbf{und}_{\text{NP}}(\text{apple})) : \mathbf{Atel}(\text{john}_{\text{Act}}, \mathbf{und}_{\text{NP}}(\text{apple}))$$

，这正如在第 0.2 节，0.2 所描述的 in -副词测试所预测的那样。另一方面，three apples 是一个有界名词短语

$$\text{threeApples} \triangleq \mathbf{AmountOf}(\text{apple}, \text{quantity}, \text{nu}, 3) : \mathbf{NP}^{\text{B}}$$

，依据第 1.4 节，因此事件 John ate three apples 被建模为一个终止事件

$$\text{eat john}_{\text{Act}}(\text{B}, \mathbf{und}_{\text{NP}}(\text{threeApples})) : \mathbf{Tel}(\text{john}_{\text{Act}}, \mathbf{und}_{\text{NP}}(\text{threeApples}))$$

，这再次如第 0.2 节，0.2 所描述的 in -副词测试所预测的那样。那么实体行为受体呢？回顾一下，实体是名词短语的一个实例，事件 John ate a particular apple 和 John ate a particular three apples 都有一个实体作为行为受体。假设 $a_1, a_2, a_3 : \mathbf{El}_{\text{NP}}^{\text{U}}(\text{apple})$ 是三个特定的苹果，那么

$$\mathbf{und}_{\text{Entity}}((\text{U}, \text{apple}), a_1) : \mathbf{Und}^{\text{B}}$$

是作为行为受体的特定 apple a_1 ，而

$$\mathbf{und}_{\text{Entity}}((\text{B}, \text{threeApples}), f(a_1) \oplus f(a_2) \oplus f(a_3)) : \mathbf{Und}^{\text{B}}$$

是作为行为受体的特定三个 apples a_1, a_2, a_3 ，其中上述术语中的 f 是一个将一个苹果转换为一个苹果的函数，沿着例子 ?? 的思路。重要的是，尽管 apple 是无界的且 three apples 是有界的，但 a particular apple 和 a particular three apples 都是有界的，因为所有实体都是有界承受者。因此，我们的框架正确地检测到 John ate a particular apple 和 John ate a particular three apples 是终止事件。

Remark 2.3.2. 请注意，我们对裸复数的处理与广泛用于模拟复数性的 (?) 集合论框架非常不同。在集合论框架中，裸复数 apples（苹果）被解释为所有苹果的集合 (? , 44)，而“约翰吃苹果”的语义涉及到约翰吃了某一个特定的苹果的集合。这实际上将苹果与一些特定数量 n 的苹果等同起来，这是不理想的，因为前者产生非限时性事件，而后者产生限时性事件。

最后，虚拟受事 $\mathbf{und}_* : \mathbf{Und}^{\text{U}}$ 是无界的，因为，至少在英语中，带有未指明受事的句子似乎总是非终结性的。例如，约翰唱歌中的未指定受事似乎指的是歌曲而不是一首歌；约翰唱歌是非终结性的（根据 in -副词的诊断），如果未指定受事是一个歌曲，那么它就应该是终结性的。此外，从不带受事的动词，如约翰跑步 2.1.2，似乎总是非终结性的——至少如果不加入附加成分，这超出了我们分析的范围。

2.4 态

到目前为止，我们对动词领域的讨论集中在事件上，比如“气球爆裂了”，这些事件表达动态的可能性。现在我们将注意力转向静态的可能性，比如“气球被爆裂了”，我们称之为状态。我们引入状态有两个原因。首先，我们的目标之一是建立一个合理的可能性本体论，而

任何这样的本体论都应包括状态。其次，在第 2.5 节中，我们将把顶点事件描述为达到其结果状态的终极事件。

作为事件的静态类比，状态的类型反映了在第 2.2 节中定义的事件类型：对于任何演员 $a : \mathbf{Act}$ 和 b 承受者 $und : \mathbf{Und}^b$ （其中 $b : \mathbf{Bd}$ ），我们有一个涉及该演员和承受者的状态类型 $\mathbf{State}^b(a, und)$ 。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^b}{\Gamma \vdash \mathbf{State}^b(a, und) \text{ type}}$$

对于任何状态 $s : \mathbf{State}^b(a, und)$ ，我们可以考虑命题 $\mathbf{El}_{\mathbf{State}}(s) : \mathbf{Prop}$ ，即状态 s 成立。注意，不同于事件 $evt : \mathbf{Evt}^b(a, und)$ ，它们导致了一种发生类型 $\mathbf{El}_{\mathbf{Evt}}(evt)$ ，一个状态仅导致一个命题；这是因为我们不认为一个状态能够以多种方式成立。

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^b \quad \Gamma \vdash s : \mathbf{State}^b(a, und)}{\Gamma \vdash \mathbf{El}_{\mathbf{State}}(s) : \mathbf{Prop}}$$

使用 Σ 类型，我们还可以定义状态类型，其中一个或没有指明执行者或承受者—— $\mathbf{State}_A(a)$ 、 $\mathbf{State}_{\mathbf{Und}}^b(und)$ 和 $\mathbf{State}$ ——每个都有一个基础状态为类型 $\mathbf{State}^b(a, und)$ ，因此也关联到一个命题。与事件类似，被抑制的参数可能是也可能不是虚拟参数。

例如，句子“John loves Mary”表达了一种状态，其承受者是 Mary，行为者是 John；句子“Mary is loved”表达了一种状态，其承受者是 Mary，行为者被隐藏（但可能是 John）。与事件（John ran）一样，有些状态可能需要一个虚拟的参与者：句子“The balloon is popped”是一个状态，其承受者是气球，行为者必须是虚拟的。

2.5 高潮事件

最后我们将注意力转向顶点性。回忆一下，终结事件是具有自然关联的结果状态的事件，我们将顶点事件定义为获得其结果状态的终结事件。注意，对于非终结事件是否为顶点事件的问题，甚至没有意义，因为不存在要获得的结果状态。我们的定义与提到顶点性的之前工作（如 ?）一致。在其他关于词汇语义的工作中（如 ?），顶点事件与终结事件等同，其中终结（对我们而言，顶点）事件的定义特性是它们蕴涵结果状态。

我们注意到，无终止事件可以涉及状态的改变；它们只是不包含结果状态。例如，事件“John 弹破气球”涉及状态改变，因为某些气球被弹破，但它没有结果状态，因此是无终止的。要看到它没有结果状态，可以观察到继续说“...，然后他完成了弹破所有气球”会显得奇怪；这是因为“完成”指的是一个终点，而“John 弹破气球”并不包含一个终点。相反，有限止事件“John 弹破了三个气球”有一个明确的结果状态，即那三个气球被弹破，事实上，该事件由于达到那个状态而告终。

我们首先为每一个目的事件关联一个结果状态。

$$\frac{\Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^B \quad \Gamma \vdash evt : \mathbf{Tel}(a, und)}{\Gamma \vdash \mathbf{Result}(evt) : \mathbf{State}^B(\mathbf{act}_*, und)}$$

注意，具有行为者 a 和受事者 und 的事件所产生的状态涉及相同的受事者，但行为者总是虚构的 $\mathbf{act}_*$ 。确实，描述某些事件产生状态的句子似乎总是具有这样一个特性，即它们没有指定行为者的对应项。另一方面，并非每个 $s : \mathbf{State}^B(\mathbf{act}_*, und)$ 都是终止事件的结果状态；反例包括“苹果是好吃的”和“约翰是累的”。

手握结果状态的概念后，我们现在可以正式定义一个限量事件的完成性，即该事件如果发生，则它的结果状态成立的命题。根据我们使用的 **Prop** 的哪种形式定义，这个 **isCul** 的定义可能显然是一个命题（因为它是形式为 $X \rightarrow \mathbf{Prf}(Y)$ ，其中 Y 是一个命题）或者我们需要去证明它。在我们的 Agda 形式化中，我们将 **Prop** 定义为同伦命题的类型(?)，并使用函数外延性来证明 **isCul** 是一个同伦命题。

最后，对于任何行为者 $a : \mathbf{Act}$ 和有界体验者 $und : \mathbf{Und}^B$ ，我们将终结事件 $\mathbf{Cul}(a, und)$ 定义为一个完成事件及其完成证明的对：

$$\frac{\Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^B}{\Gamma \vdash \mathbf{Cul}(a, und) \triangleq \sum_{\text{evt} : \mathbf{Tel}(a, und)} \mathbf{Prf}(\mathbf{isCul}(\text{evt})) \text{ type}}$$

我们可以再次定义终结事件的类型，其中行为者或体验者之一或都未被指定—— $\mathbf{Cul}_A(a)$ 、 $\mathbf{Cul}_{\text{Und}}(und)$ 和 $\mathbf{Cul}$ ——并将每种类型与其发生的类型（即底层完成事件的发生类型）关联。注意，终结事件的体验者必须是有界的，因为完成事件是具有有界体验者的事件。特别地，体验者不能是虚拟体验者。

Example 2.5.1. 请考虑句子 John 弹破了三个气球，这是一个终结且达到顶点的事件，其结果状态是三个气球被弹破。基本设置与 Example 2.2.5 中相同。我们假设：

$$\frac{}{\Gamma \vdash \text{human} : \mathbf{NP}^U} \quad \frac{}{\Gamma \vdash \text{john} : \mathbf{El}_{\mathbf{NP}}^U(\text{human})} \quad \frac{}{\Gamma \vdash \text{balloon} : \mathbf{NP}^U}$$

$$\frac{}{\Gamma \vdash \text{pop} : (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(a, \text{snd}(u))}$$

从这些我们可以定义行为者 John，经历者三个气球（其被框架自动确定为有限），以及（自动确定为）表示 John 弹破了三个气球的终结事件 evt_1 。

$$\text{john}_{\mathbf{Act}} \triangleq \mathbf{act}_{\mathbf{Entity}}((U, \text{human}), \text{john}) : \mathbf{Act}$$

$$\text{threeBalloons}_{\mathbf{Und}} \triangleq \mathbf{und}_{\mathbf{NP}}(\mathbf{AmountOf}(\text{balloon}, \text{quantity}, \text{nu}, 3)) : \mathbf{Und}^B$$

$$\text{evt}_1 \triangleq \text{pop } \text{john}_{\mathbf{Act}} (\mathbf{B}, \text{threeBalloons}_{\mathbf{Und}}) : \mathbf{Tel}(\text{john}_{\mathbf{Act}}, \text{threeBalloons}_{\mathbf{Und}})$$

为了模拟 pop 的顶点性，我们增加三个进一步的假设：(1) 一个由经历者索引的状态家族， popped ，表示静态谓词 be popped；(2) 一个陈述 **Result** 的方程，指出对有限经历者的 pop ping 事件的结果是 popped 的状态；以及 (3) 公理 popC ，表明任何一次发生有限经历者的 pop ping 事件会导致 popped 状态的发生。

$$\frac{}{\Gamma \vdash \text{popped} : (und : \mathbf{Und}^B) \rightarrow \mathbf{State}^B(\mathbf{act}_*, und)}$$

$$\frac{\Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash und : \mathbf{Und}^B}{\Gamma \vdash \mathbf{Result}(\text{pop } a (\mathbf{B}, und)) \triangleq \text{popped } und : \mathbf{State}^B(\mathbf{act}_*, und)}$$

$$\frac{}{\Gamma \vdash \text{popC} : (a : \mathbf{Act}) \rightarrow (und : \mathbf{Und}^B) \rightarrow \mathbf{El}_{\mathbf{Evt}}(\text{pop } a (\mathbf{B}, und)) \rightarrow \mathbf{Prf}(\mathbf{El}_{\mathbf{State}}(\text{popped } und))}$$

注意，这三个假设都仅涉及有限的经历者。从这些假设中我们可以得出，当经历者是有限时，pop 是终结的。我们可以通过给 pop 限制在有限经历者上的更精细的类型来表达这一点；通过展开定义，以下术语类型检查可以直接得出。

$$\text{pop}^B \triangleq \lambda a. \lambda und. (\text{pop } a \text{ und}, \text{popC } a \text{ und}) : (a : \mathbf{Act}) \rightarrow (und : \mathbf{Und}^B) \rightarrow \mathbf{Cul}(a, und)$$

使用 pop^B ，我们可以很容易地证明事件 John 弹破了三个气球和三个气球自己被弹破都是终结的：

$$\begin{aligned}\text{evt}_1 &\triangleq \text{pop}^B \text{ john}_{\text{Act}} \text{ threeBalloons}_{\text{Und}} : \mathbf{Cul}(\text{john}_{\text{Act}}, \text{threeBalloons}_{\text{Und}}) \\ \text{evt}_2 &\triangleq \text{pop}^B \text{ act}_* \text{ threeBalloons}_{\text{Und}} : \mathbf{Cul}(\text{act}_*, \text{threeBalloons}_{\text{Und}})\end{aligned}$$

此外，我们可以证明如果发生了 John 弹破了三个气球的事件，这意味着三个气球已被弹破的状态：

$$\text{snd}(\text{evt}_1) : \mathbf{El}_{\text{Evt}}(\text{evt}_1) \rightarrow \mathbf{Prf}(\mathbf{El}_{\text{State}}(\text{popped threeBalloons}_{\text{Und}}))$$

Example 2.5.2. 英语动词如 pop 和 eat 是英语中最常见的一类动词：当它们的承受者是有界时，会产生限时和终结事件，否则会产生非限时事件。在例子 2.5.1 中，我们看到我们可以通过 pop^B 的类型捕捉有界 pop 的终结性。在这个例子中，我们展示依赖类型理论如何在单一类型中同时捕捉有界 pop 的终结性和无界 pop 的非限时性。这需要依赖类型理论的一些更高级的特性，即依赖模式匹配 (?) 和类型宇宙 **Type**。使用依赖模式匹配，我们可以根据承受者 $u = (b, \text{und}) : \mathbf{Und}$ 的有界性 b 来定义类型族 **CulOrAtel**。具体来说，当 $b = B$ 时，我们定义 **CulOrAtel** $a (b, \text{und})$ 为 **Cul**(a, und)，而当 $b = U$ 时为 **Atel**(a, und)：

$$\begin{aligned}\mathbf{CulOrAtel} &: (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Type} \\ \mathbf{CulOrAtel} \ a \ (B, \text{und}) &\triangleq \mathbf{Cul}(a, \text{und}) \\ \mathbf{CulOrAtel} \ a \ (U, \text{und}) &\triangleq \mathbf{Atel}(a, \text{und})\end{aligned}$$

然后，我们通过依赖模式匹配定义一个函数 $\text{pop}^{\cdot}$ ，对于任何行为者 $a : \mathbf{Act}$ 和承受者 $u : \mathbf{Und}$ 产生一个类型为 **CulOrAtel** $a u$ 的术语：

$$\begin{aligned}\text{pop}^{\cdot} &: (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{CulOrAtel} \ a \ u \\ \text{pop}^{\cdot} \ a \ (B, \text{und}) &\triangleq \text{pop}^B \ a \ \text{und} \\ \text{pop}^{\cdot} \ a \ (U, \text{und}) &\triangleq \text{pop} \ a \ (U, \text{und})\end{aligned}$$

为了验证 $\text{pop}^{\cdot}$ 的每个条款类型检查，我们必须根据承受者的有界性展开 **CulOrAtel** $a u$ 的定义。 $\text{pop}^{\cdot}$ 的类型完全捕捉了 pop 就其限时性和终结性而言的行为。因此，与其假设 pop、popped 和 popC，我们可以更直接地通过一个函数 $(a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{CulOrAtel} \ a \ u$ 建模 pop (eat 等)。

Remark 2.5.3. 在英文中也有完成但未达到终结的事件的例子。完成事件 “John cleaned the table” 的结果状态可以用 “桌子是干净的” 来描述，但这并不意味着 “John cleaned the table” 达到了 “桌子是干净的” 这种状态。在我们的框架中，这类动词被分配如下类型：

$$(a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(a, \text{snd}(u))$$

如我们在第 0.1 节中讨论的，还有一些歧义动词。“wipe” 这个词有两种解释，一种仅仅涉及在某个表面上来回移动布，另一种涉及实际试图将表面清理干净。后一种解释，如 “clean”，在其接受体有界时是完成但未达到终结事件，而在其接受体无界时是非完成事件。前一种解释不涉及任何目的接近，因此在结果事件的完成性上不表现出交替变换；因此，正如在第 0.2 节中提到的，这超出了我们的研究范围。

Example 2.5.4. 我们的框架也可以用来建模不存在于英语中的动词。例如，如在 ? 中讨论过的，俄语中所谓的典型完成体动词形式只能将有限的承受者作为参数，引发必然渐进地达到顶点的终止事件。（如果它们的承受者是一个裸名词，则它会接收有限的解读。）在我们的框架中，我们可以为这样的动词形式指定类型：

$$(a : \mathbf{Act}) \rightarrow (\text{und} : \mathbf{Und}^B) \rightarrow \mathbf{Cul}(a, \text{und})$$

2.6 副词修饰

将事件视为发生的类型，使得可以像形容词修饰一样容易处理副词修饰（见第 1.5 节）。我们将只关注方式副词；关于其他种类副词的依赖类型处理以及方式副词的其他处理，参见 ???。我们注意到，下面的处理是以 ?, 359 的精神进行的，尽管那项工作不关注依赖事件类型；而与我们的处理不同的是，? 将事件建模为命题的证明对象。

正如形容词产生实体的谓词——某个名词短语的实例——我们将副词建模为发生的谓词，或某个事件的实例。发生是名词领域中的实体在动词领域中的等价物（第 1.2 节）。到目前为止，我们仅有机会讨论固定事件 $evt : \mathbf{Evt}^b(a, und)$ 的出现 $occ : \mathbf{El}_{\mathbf{Evt}}(evt)$ ，但我们可以将所有出现 $o : \mathbf{Occ}$ 的类型定义为事件 $e : \mathbf{Evt}$ 与该事件实例的对。（在下面的规则中，请记住 $e : \mathbf{Evt}$ 的形式为 $(a, (u, evt))$ ，其中 $evt : \mathbf{Evt}^{\text{fst}(u)}(a, \text{snd}(u))$ 。）

$$\Gamma \vdash \mathbf{Occ} \triangleq \sum_{e:\mathbf{Evt}} \mathbf{El}_{\mathbf{Evt}}(\text{snd}(\text{snd}(e))) \text{ type}$$

为了反映我们的形容词修饰结构（第 1.5 节），我们要求对于任何事件 evt 的某些事件出现的谓词 P ，“满足 P 的 evt 出现子集，” $\sum_{occ:evt}^{\mathbf{Evt}} P$ ，再次作为具有相同行为者和承受者的事件。也就是说，类似于第 1.1 节中 $\mathbf{NP}^b$ 的规则，我们要求 $\mathbf{Evt}^b(a, und)$ 类型在 Σ -谓词类型下对 $\mathbf{El}_{\mathbf{Evt}}(evt)$ 具有封闭性：

$$\frac{\Gamma \vdash und : \mathbf{Und}^b \quad \Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash evt : \mathbf{Evt}^b(a, und) \quad \Gamma, occ : \mathbf{El}_{\mathbf{Evt}}(evt) \vdash P : \mathbf{Prop}}{\Gamma \vdash \sum_{occ:evt}^{\mathbf{Evt}} P : \mathbf{Evt}^b(a, und)}$$

$$\frac{\Gamma \vdash und : \mathbf{Und}^b \quad \Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash evt : \mathbf{Evt}^b(a, und) \quad \Gamma, occ : \mathbf{El}_{\mathbf{Evt}}(evt) \vdash P : \mathbf{Prop}}{\Gamma \vdash \mathbf{El}_{\mathbf{Evt}}\left(\sum_{occ:evt}^{\mathbf{Evt}} P\right) = \sum_{occ:\mathbf{El}_{\mathbf{Evt}}(evt)} \mathbf{Prf}(P) \text{ type}}$$

然后我们可以通过对 $\mathbf{Occ}$ 的谓词来模拟副词。例如，使用对每一个快速发生事件都成立的谓词 $\text{quick} : \mathbf{Occ} \rightarrow \mathbf{Prop}$ ，我们可以表示一个事件快速发生。

Example 2.6.1. 让我们来说明如何表示事件“John 快速吃苹果”。如示例 2.2.1 中所示，我们对任意参与者和经历者拥有一族事件 eat ，

$$\Gamma \vdash \text{eat} : (a : \mathbf{Act}) \rightarrow (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\text{fst}(u)}(a, \text{snd}(u))$$

我们可以将其应用于参与者 John 和无界经历者苹果以获得代表事件“John 吃苹果”的术语 $\text{jaa} : \mathbf{Evt}^U(\text{john}_{\mathbf{Act}}, \text{apple}_{\mathbf{Und}})$ 。

$$\begin{aligned} \text{john}_{\mathbf{Act}} &\triangleq \mathbf{act}_{\mathbf{Entity}}((U, \text{human}), \text{john}) : \mathbf{Act} \\ \text{apple}_{\mathbf{Und}} &\triangleq \mathbf{und}_{\mathbf{NP}}(\text{apple}) : \mathbf{Und}^U \\ \text{jaa} &\triangleq \text{eat } \text{john}_{\mathbf{Act}} (U, \text{apple}_{\mathbf{Und}}) : \mathbf{Evt}^U(\text{john}_{\mathbf{Act}}, \text{apple}_{\mathbf{Und}}) \end{aligned}$$

我们进一步在我们的框架中添加谓词 quick 。

$$\Gamma \vdash \text{quick} : \mathbf{Occ} \rightarrow \mathbf{Prop}$$

我们可以将 `quick` 应用于任何出现的 $o : \mathbf{Occ}$ ，但我们也可以通过预组合将其限制为一个关于“John 吃苹果”事件出现的谓词 $occ : \mathbf{El}_{\mathbf{Evt}}(\text{jaa})$ ：

$$\lambda occ.\text{quick}((\text{john}_{\mathbf{Act}}, (\text{apple}_{\mathbf{Und}}, \text{jaa})), occ) : \mathbf{El}_{\mathbf{Evt}}(\text{jaa}) \rightarrow \mathbf{Prop}$$

利用这一观察，我们将“John 快速吃苹果”建模为对应于“John 吃苹果”的快速出现子集的事件：

$$\text{jaaQuickly} \triangleq \sum_{occ:\text{jaa}}^{\mathbf{Evt}} \text{quick}((\text{john}_{\mathbf{Act}}, (\text{apple}_{\mathbf{Und}}, \text{jaa})), occ) : \mathbf{Evt}^{\mathbf{U}}(\text{john}_{\mathbf{Act}}, \text{apple}_{\mathbf{Und}})$$

一次“John 快速吃苹果”的出现是 $\mathbf{El}_{\mathbf{Evt}}(\text{jaaQuickly})$ 类型的术语。通过

$$\mathbf{El}_{\mathbf{Evt}}(\text{jaaQuickly}) = \sum_{occ:\mathbf{El}_{\mathbf{Evt}}(\text{jaa})} \mathbf{Prf}(\text{quick}((\text{john}_{\mathbf{Act}}, (\text{apple}_{\mathbf{Und}}, \text{jaa})), occ)) \text{ type}$$

这样的术语是“John 吃苹果”出现的一个出现与一个见证 $prf : \mathbf{Prf}(\text{quick}((\text{john}_{\mathbf{Act}}, (\text{apple}_{\mathbf{Und}}, \text{jaa})), occ))$ 的对，表明 occ 是快速的。特别地，我们可以不费力地证明“John 快速吃苹果”蕴涵“John 吃苹果”：

$$\mathbf{fst} : \mathbf{El}_{\mathbf{Evt}}(\text{jaaQuickly}) \rightarrow \mathbf{El}_{\mathbf{Evt}}(\text{jaa})$$

2.7 进一步的蕴涵

我们已经讨论了几种蕴涵：从及物句到不及物句（例子 2.2.4 和 2.2.5），从终结句到其结果状态（章节 2.5），以及从副词修饰句到非副词修饰句（例子 2.6.1）。所有这些蕴涵都是通过构建来解释的，但我们理论中还有一些其他类型的蕴涵尚未捕捉到。在本节中，我们考虑另外两个原则，每个原则都解释了一类关于事件承受者的蕴涵。

第一类蕴涵是，描述目的性事件的句子必须蕴涵描述对应非目的性事件的句子。例如，约翰吃了三个苹果应该蕴涵约翰吃苹果；也就是说，我们应该能够构建一个类型为

$$\mathbf{El}_{\mathbf{Evt}}(\text{eat john}_{\mathbf{Act}}(\mathbf{B}, \mathbf{und}_{\mathbf{NP}}(\text{threeApples}))) \rightarrow \mathbf{El}_{\mathbf{Evt}}(\text{eat john}_{\mathbf{Act}}(\mathbf{U}, \mathbf{und}_{\mathbf{NP}}(\text{apple}))) \quad (\pounds)$$

的项，其中 $\text{threeApples} \triangleq \mathbf{AmountOf}(\text{apple}, \text{quantity}, \text{nu}, 3) : \mathbf{NP}^{\mathbf{B}}$ 如示例 2.3.1 所示。

总体原则如下：对于任何发生在施事 a 和受事 $(\mathbf{B}, \mathbf{und}_{\mathbf{NP}}(\mathbf{AmountOf}(np, d, u, m)))$ 的事件，相同的事件也应发生在施事 a 和受事 $(\mathbf{U}, \mathbf{und}_{\mathbf{NP}}(np))$ 。我们将以下规则添加到我们的框架中：

$$\frac{\begin{array}{l} \Gamma \vdash d : \mathbf{Degree} \quad \Gamma \vdash u : \mathbf{Units}(d) \\ \Gamma \vdash m : \mathbf{Nat} \quad \Gamma \vdash a : \mathbf{Act} \quad \Gamma \vdash f : (u : \mathbf{Und}) \rightarrow \mathbf{Evt}^{\mathbf{fst}(u)}(a, \mathbf{snd}(u)) \\ \Gamma \vdash np : \mathbf{NP}^{\mathbf{U}} \quad \Gamma \vdash occ : \mathbf{El}_{\mathbf{Evt}}(f(\mathbf{B}, \mathbf{und}_{\mathbf{NP}}(\mathbf{AmountOf}(np, d, u, m)))) \end{array}}{\Gamma \vdash \mathbf{EvtAmtIsNP}(f, occ) : \mathbf{El}_{\mathbf{Evt}}(f(\mathbf{U}, \mathbf{und}_{\mathbf{NP}}(np)))}$$

这一规则有一个值得注意的细微差别：在前提 f 中，我们看到事件（实际上是一组事件）必须适用于任何经历者，无论是否有界，因为前提涉及一个有界的经历者，而结论则是一个不同的、无界的经历者。

很容易验证 $\lambda occ.\mathbf{EvtAmtIsNP}(\text{eat john}_{\mathbf{Act}}, occ)$ 具有类型 $(\pounds)$ ，从而确立了“John 吃了三个苹果”蕴涵“John 吃了苹果”。

第二类蕴涵涉及将一个受动者从一个实体改变为一个有限名词短语：汤姆吃了杰瑞和米奇应该蕴涵汤姆吃了两只老鼠，即

$$\mathbf{El}_{\mathbf{Evt}}(\text{eat tom}_{\mathbf{Act}}(\mathbf{B}, \mathbf{und}_{\mathbf{Entity}}((\mathbf{B}, \text{twoMice}), \text{jerryAndMickey}))) \rightarrow \mathbf{El}_{\mathbf{Evt}}(\text{eat tom}_{\mathbf{Act}}(\mathbf{B}, \mathbf{und}_{\mathbf{NP}}(\text{twoMice}))) \quad (\pounds)$$

假设我们有术语

$$\begin{aligned} \text{tom}_{\text{Act}} &: \mathbf{Act} \\ \text{mouse} &: \mathbf{NP}^U \\ \text{twoMice} &\triangleq \mathbf{AmountOf}(\text{mouse}, \text{quantity}, \text{nu}, 2) : \mathbf{NP}^B \\ \text{jerryAndMickey} &: \mathbf{El}_{\mathbf{NP}}^B(\text{twoMice}) \end{aligned}$$

一般原则由以下规则表示，我们也将其添加到我们的框架中。注意，事件家族 f 只需要适用于任何有界的承受者。使用上述规则，我们可以构造一个类型为 (α) 的项，如下所示：

$$\lambda \text{occ. EvtAmtIsNP}((\lambda \text{und. eat tom}_{\text{Act}}(\mathbf{B}, \text{und})), \text{jerryAndMickey}, \text{occ})$$

，从而建立蕴含关系 $\text{Tom ate Jerry and Mickey} \Rightarrow \text{Tom ate two mice}$ 。

3 Agda 实现

本文中的所有框架规则和示例都已在 Agda 证明助手中形式化(?)；我们的代码可在 <https://doi.org/10.5281/zenodo.15602618> 找到。我们的 Agda 形式化证明了我们框架中的规则在语法上是格式良好的，并且我们所有的示例定义和蕴涵都遵循本文中描述的框架。

我们的开发包括 637 行的 Agda 代码，其中包括空行和注释，并且是完全自包含的。这段代码可以分为以下几个部分：

- 基本导入，**Prop** 的定义，以及函数外延性：30 行。
- 来自第 ?? 节的规则（名义框架）：130 行。
- 第 ?? 节的例子：164 行。
- 第 2 节的规则（口头框架）：113 行。
- 来源于第 2 节的例子：200 行。

我们通过结合公设、沿着假定的等式的重写规则以及普通的数据和函数定义，将框架规则编码到 Agda 中。例如，

⁰¹⁵ 这与第 1.1 节中的以下两个规则具有相同的效果：

$$\frac{\Gamma \vdash b : \mathbf{Bd}}{\Gamma \vdash \mathbf{NP}^b \text{ type}} \qquad \frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}^b(np) \text{ type}}$$

我们的一些框架规则表达了判断等式，例如下面的第二条规则：

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma, p : \mathbf{El}_{\mathbf{NP}}^b(np) \vdash P : \mathbf{Prop}}{\Gamma \vdash \sum_{p:np}^{\mathbf{NP}} P : \mathbf{NP}^b}$$

$$\frac{\Gamma \vdash b : \mathbf{Bd} \quad \Gamma \vdash np : \mathbf{NP}^b \quad \Gamma, p : \mathbf{El}_{\mathbf{NP}}^b(np) \vdash P : \mathbf{Prop}}{\Gamma \vdash \mathbf{El}_{\mathbf{NP}}^b\left(\sum_{p:np}^{\mathbf{NP}} P\right) = \sum_{p:\mathbf{El}_{\mathbf{NP}}^b(np)} \mathbf{Prf}(P) \text{ type}}$$

¹⁵在 Agda 中， $\sim b : \mathbf{Bd}' \rightarrow \dots$ 是 Π 类型的表示法，其参数默认被省略。