arXiv:2304.01331v1 [cs.CL] 3 Apr 2023

不使用词典创建自定义事件数据：实用技巧集 *

Andrew Halterman†、Philip A. Schrodt‡、Andreas Beger§、Benjamin E. Bagozzi¶、Grace I. Scarborough‖

2023年4月5日

摘要

事件数据，即从文本中自动提取的“谁对谁做了什么”的结构化记录，是国际政治学者的重要数据来源。开发新事件数据集的成本很高，尤其是使用依赖手工构建词典的自动化系统时。因此，大多数研究者会采用 ICEWS 等大型现有数据集，而不是开发针对具体研究问题优化的定制事件数据集。本文介绍了一套高效生成定制事件数据的“技巧集”，借助自然语言处理（NLP）的最新进展，使研究者能够快速生成定制事件数据集。本文介绍了相关技术，包括利用主动学习训练事件类别分类器；使用大语言模型、标准机器学习分类器和自然语言处理领域预训练的“问答”模型，在文本中识别行为体和行动接受者；以及将行为体的提及与相应的维基百科条目关联起来，从而对其进行分类。我们介绍了如何运用这些技术构建新的 Polecat 全球事件数据集，该数据集旨在取代 ICEWS；同时也举例说明学者如何快速生成规模较小的定制事件数据集。我们公开了示例代码和模型，供他人实现这些新技术。

*本研究由政治不稳定工作组（PITF）赞助。PITF由中央情报局资助。本文所表达的观点仅代表作者本人，不代表美国政府的观点。Plover的资金最初部分由美国国家科学基金会项目SBE-1539302提供，项目名称为“RIDIR：面向大数据社会科学研究的政治事件数据现代化”。我们谨此致谢，并对Leidos方面参与此项开发工作的John Berrie、Heather Carter、Kathleen Krupenvich和Jevon Spivey深表感谢。所有错误及解读均由我们自行承担。

†密歇根州立大学，ahalterman0@gmail.com
‡Parus AnalyticsLLC, schrodt735@gmail.com§Basil Analytics, adbeger@gmail.com
¶特拉华大学, bagozzib@udel.edu
‖Leidos，grace.i.scarborough@leidos.com

1 引言

事件数据，即从文本中自动提取的“谁对谁做了什么”的结构化记录，是国际政治学者的重要数据来源。开发新事件数据集的成本很高，尤其是使用依赖手工构建词典的自动化系统时。因此，大多数学者会采用 ICEWS 等大型现有数据集，而不是开发针对具体研究问题优化的定制事件数据集。

本文介绍了一套用于高效生成定制事件数据的“技巧集”，既借鉴了自然语言处理（NLP）的最新进展——这些进展使研究人员能够快速生成定制事件数据集——也依托政治不稳定工作组（PITF）资助下正在开展的工作，创建了 NGEC 系统。该系统采用 Plover事件数据本体，构建了一个新的全球近实时数据集 Polecat，覆盖范围遍及全球，时间跨度从2010年至今。[1] 我们介绍了以下技术：利用主动学习训练事件类别分类器；使用大语言模型、标准机器学习分类器以及自然语言处理领域预训练的“问答”模型，从文本中识别行为体和行动接受者；以及将行为体的提及与相应的维基百科条目关联起来，以便对其进行分类。我们还初步比较了手工编码的 BFRS数据集与我们的数据集；该 BFRS数据集记录了巴基斯坦自1988年至2011年11月的政治暴力事件（Bueno de Mesquita 等，2015）。最后，我们讨论了系统开发和部署过程中涉及的各类实际操作问题。我们公开了用于实现本文所述基于属性和实体的技术的 Python 代码；使用者需自行构建文本语料库并进行标注。[2]

[1]Halterman 等人（2023）对 Plover/Polecat 进行了详细介绍。[2]此代码可在以下网址找到：https://github.com/philip-schrodt/NGEC 和 https://github.com/ ahalterman/NGEC。

1.1 事件抽取概述

从文本生成事件记录，需要从文档中提取信息、对其进行分类，并将这些信息整合为一条事件记录。我们的方法中，事件抽取包括以下步骤：

1. **事件分类**：给定一份文档，我们希望识别文本中报道的事件类型。例如，我们可能会识别出某份文档报道了一起 PROTEST 和一起动员事件。[3]

2. **子事件（模式）分类**：给定一份文档和一组已检测到的事件，接下来我们希望识别这些事件的子类别（“模式”）。例如，对于一份事件分类器识别出一起 PROTEST 事件的文档，我们会进一步识别该 PROTEST 事件属于骚乱、示威、绝食抗议等哪一种。

3. **语境分类**：研究人员通常希望了解事件发生的**语境** 。例如，新闻报道中的 PROTEST 可能涉及政治腐败、环境或民主运动。了解事件的**语境** 有助于研究人员理解该事件的作用，或将分析范围限定为与特定语境相关的一组事件。

4. **事件属性识别**：我们将政治事件概念化为具有若干属性，具体而言，事件的

- **施事者**：哪个人物、组织或实体正在发起或执行该事件?
- **接收者**：事件的对象是哪个实体?
- **地点**：事件发生在哪里?
- **日期**：事件发生于何时?

[3] 在全文中，我们都会提及Plover事件本体。不过请注意，正如我们在复现BFRS数据集的章节中所展示的，这些步骤同样适用于其他事件本体或自定义事件本体。

我们希望找出文档中报告每个事件-模式组合相关信息的短语。

5. **参与者与地点消歧**：根据上一步提取的短语，我们希望将对同一人物、组织、实体和地点的不同提及对应到外部知识库中的唯一标识符。实际操作中，这需要为每个实体或地点找到对应的维基百科条目或 GeoNames地名录中的词条。

6. **实体分类**：最后，在将实体与其维基百科词条对应后，我们希望根据实体的政治角色对其进行分类。例如，我们可以为所有军事行为体分配 MIL代码，以便更轻松地分析数据。

本文的大部分内容介绍了执行这些任务的新技术，并将其与以往通常基于词典的方法进行比较。将这些步骤结合起来，构成完整的处理流程后，便可从新闻文本中生成事件数据。本文主要讨论如何运用这些步骤生成新的 Polecat数据集，但这些步骤也很容易用于生成定制数据。此外，每个步骤都可以单独使用，以满足特定的研究需求。[4]

1.2 基于词典的编码器的局限性

大多数现有的大规模自动化事件数据集依赖关键词匹配技术来识别事件类型并为行为体分配标签（King和Lowe，2003；Boschee等人，2015）。举例来说，基于词典的编码器会将“threw weight behind”这样的动词短语与词典进行比对，该词典将短语映射到相应的事件类别（在 ICEWS、Phoenix 等使用的CAMEO 本体中，则映射到一个外交合作实例）。行为体也以类似方式解析，将一般性术语（“士兵”）或具体姓名（“Stanley McChrystal”）映射到其军[事] 角色。因此，大多数事件[4]由于本文引用了许多现有数据集和此前编码人员的成果，我们在文末附上术语表

of th论文。

编码员使用事件类别词典将动词短语映射到事件类型，并使用行为体词典将政治行为体的提及映射到其所属国家和类别代码。使用词典识别事件有几个优点：词典在技术上易于实现、易于解读，而且在大型文本语料库上运行速度很快。某些事件类型和行为体只需少量独特模式即可轻松识别，因此，对于聚焦于少数行为体和离散事件的定制事件数据集而言，词典是一种实用工具，例如叙利亚的军事攻势（Halterman，2019a）、车臣的叛乱组织与政府之间的暴力（Toft和Zhukov，2015年），或印度的族群间暴力（Brathwaite和Park，2018年）。

不过，基于词典的编码器存在若干主要局限：它们会遗漏关键词列表中未包含的事件描述和行为体；开发成本高、耗费大量人力；需要不断更新，以反映不断变化的事件类型和政治职务；而且难以结合语境理解词语。

词典方法识别某一事件或行为体所有实例（即召回率）的能力，受限于其对精确匹配的依赖。[5] 如果词典中未包含某个动词短语或行为体，就无法对其进行编码。Althaus、Peyton 和 Shalmon（2018年）将基于词典的机器编码事件与一个金标准进行比较；该金标准是一个由人工编码的集合，涵盖尼日利亚抗议活动和暴力相关的六种事件类型。与金标准相比，基于词典的 Phoenix事件数据集召回率很低：与人工编码数据进行比较时，它对五种事件类型的识别率在 0% 至 30% 之间。召回率低，往往是因为事件词典缺少相应的模式。例如，作为 Phoenix 数据集基础的 Cameo 开源动词词典，对于要求改变政策、性侵犯、实施路边爆炸袭击等后果显著的事件类型，完全没有动词模式；对于数量多得多的其他事件类型，也仅提供少量模式。[6]

由于高召回率的事件编码需要非常详尽的词典，开发词典

[5]“召回率”指编码人员识别出的真实事件所占的比例。“精确率”指编码的事件中确为真实事件的比例。[6] 关于 CAMEO 动词词典存在的缺漏分析，请参阅：https://andrewhalterman.com/post/cameo-dictionary-coverage/。

既耗费成本又需要大量人力。对于涵盖多种事件类型和众多语境的数据集，难度还会进一步增加。Cameo 动词词典从1993年到2016年持续开发，但识别事件的召回率仍然相对较低（Althaus、Peyton 和 Shalmon，2018年）。Halterman 等人（2018年）报告称，将英语 Cameo 动词词典中的9,000个模式翻译成阿拉伯语，由一支由15名阿拉伯语学生组成的团队耗费了总计750小时。即使任务已有部分自动化的进展，手动添加其他模式并创建行为体词典仍额外耗费了数百小时（Liang 等人，2018年）。创建事件数据词典所需的巨大投入，使得任何从头开始创建词典的新项目都令人望而生畏。

最后，关键词匹配技术难以解读词语在语境中的含义。例如，“驶入城镇”在前面是“一支救援车队”还是“一列装甲车”时，含义截然不同。简单的文本匹配技术无法结合句子的语境，正确编码含义模糊的短语。为避免基于词典的编码人员所带来的成本，我们提出一种新型、通用、端到端的事件编码器，它基于使用一组新的标注数据训练的统计分类器。

2 无词典创建事件数据

以下各节将介绍我们在从文本编码事件数据时各步骤中采用的新技术：识别事件及子事件类型、识别事件的语境、提取报告事件所涉实体的文本、将这些实体消歧至唯一标识符，以及以实用的方式对实体进行分类。这些步骤共同构成“新一代事件编码器”（NGEC）。整个处理流程的概览见图1。我们还提供了用于实现属性识别和行为体消歧步骤的研究代码。在这些章节中，我们将讨论如何依据 Plover 本体生成 Polecat 全球事件数据集；Halterman 等人（2023）对此有更详细的介绍。不过，编码器中的各个步骤也很容易适配其他编码方案。

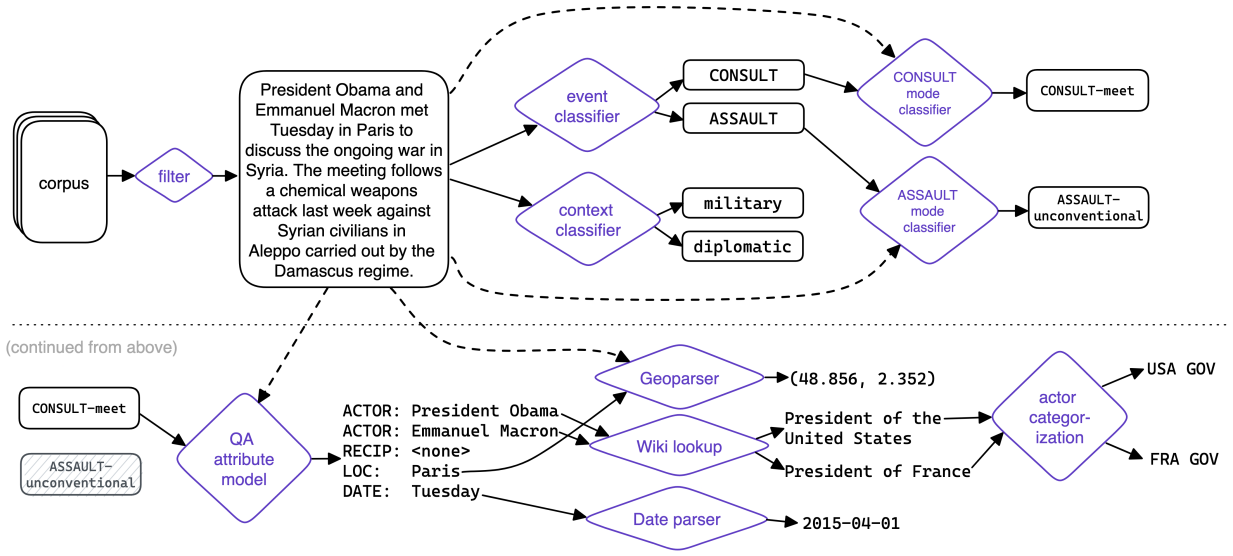

图1： 整个事件编码流程概览。模型步骤以蓝色显示，示例输出以等宽字体显示。由于篇幅所限，处理流程分为两行展示，并省略了非常规袭击事件的完整编码。文档引出的虚线表示使用整个文档的处理步骤。

2.1 基于大型语言模型的分类器进行事件分类

我们的事件数据流程首先识别文档中报告的事件。为此，我们采用一种文档级分类器，具体而言，是在一个语料库上微调的 Transformer 模型；该语料库中的文档已由我们标注了文本所报告的事件类型。下面来看一篇简短文档：

> 奥巴马总统和埃马纽埃尔·马克龙于周二在巴黎会晤，讨论叙利亚持续不断的战争。此次会晤发生在上周大马士革政权对阿勒颇叙利亚平民发动化学武器袭击之后。

在 Plover 事件本体中，我们希望能够从第一句话判断出发生了一次会晤（CONSULT）。从第二句话中，我们可以判断发生了一起“袭击”事件。

与以往的编码员不同，我们的处理流程以文档为单位，而不是逐句处理。这样做的好处是，不会对同一篇新闻报道中跨多个句子描述的同一事件进行重复编码，并且有助于识别

事件属性（如下所述）。

为识别报道中的事件类型和模式，我们使用基于 Transformer 的大语言模型（LLM）。近年来，BERT（Devlin 等，2019）、GPT（Radford 等，2019）等大语言模型及其变体，彻底改变了自然语言处理。它们通过在大型文本语料库上训练，以推断被“掩盖”的词语为目标。借此，它们生成的文档表示能够编码文本中的大量语义和句法信息，适用于下游任务。这些通用模型随后可以针对特定问题领域，在规模小得多的训练案例集上进行微调，例如文档分类、命名实体识别、文档相似度和问答模型（这些任务都用于我们的处理流程）。大语言模型高效的 Transformer 架构（Vaswani 等，2017）使其能够使用比长短期记忆网络（LSTMs）等早期模型大得多的语料库进行训练，也让它们在实际应用中更高效。

BERT（Devlin 等，2019年）是一种应用非常广泛的Transformer模型，于2019年首次发表。该模型使用一个包含34亿个词的语料库进行训练，这些词取自英文维基百科以及未发表作品的作者创作的免费小说。[7] BERT的训练和运行速度相对较慢，因此我们选择了DistilBERT（Sanh 等，2019a），这是BERT的一个较小版本，仅以准确率小幅降低为代价，运行速度却快得多。

Transformer模型在微调之前会先用大量文本进行预训练，这使它们相较于基于词典的系统具有两个显而易见的优势，以及一个意料之外的优势。首先，由于它们见过出现在非常相似语境中的不同词语，因此能够轻松推断出同一概念的同义词和替代表述；而在基于词典的方法中，这些内容必须明确列出。因此，许多词典模式都非常脆弱：措辞上的变化即使不会引起人类读者的注意，也会导致模式匹配失败。对于机器翻译的文本或由非母语者撰写的文本而言，这尤其成问题，因为这两种文本往往会使用一些

[7]训练BERT需要在配备64个图形处理器的专用系统上计算约4天，预计成本约为7000美元，这凸显了重复使用现有模型的价值。

字面上正确、但母语者通常不会优先选用的词语。

其次，这些模型会考察词语在语境中的用法，因此能够消解一个词的多种含义。例如，“驶入”一词在“坦克”（=夺取领土）和“救援车队”（=提供援助）这两种语境中含义不同。在基于词典的系统中，必须借助少量通常距离较近的词语来辨别相互竞争的含义，而且这类系统往往容易受到措辞细微变化的影响。相比之下，Transformer模型所犯的错误往往更接近人类会犯的错误。例如，事件类型 MOBILIZE 在 ICEWS 数据中极少出现，但在我们的 Polecat数据集中相当常见，通常与 PROTEST、COERCE 或 ASSAULT 等使用武力的事件同时出现。以 PROTEST 为例，描述警察驱散抗议者的文本隐含地假定：既然警察在场，他们必定先被动员起来。相比之下，基于词典的系统则需要明确提及动员，而大多数文章不会这样写，因为读者通常被认为知道，警察要出现在抗议现场，就必须先被动员起来。

事件的机器学习分类还提供了另一项优于基于词典的方法的优势。基于词典的系统执行二元分类：模式要么符合文本，要么不符合。相比之下，机器学习系统会生成一个分类分数，该分数通常与分类正确的可能性成正比。这意味着可以针对不同程度的假阳性和假阴性对分类结果进行校准。我们一直在对此进行广泛试验，因为当前的系统往往会过度分类。因此，虽然我们会存储所有得分为正的情况，但只报告得分高于第90百分位阈值的情况。

2.2 事件类别分类器

事件类别分类器是系统的核心。分配给文本的事件类别决定了后续的模式和实体处理，也决定了是否认为报道包含事件。鉴于正确完成这些初始

步骤至关重要，我们采用了基于 Transformer 的模型。我们发现，与考虑过的其他分类器相比，该模型准确率更高，但计算成本也更高。

2.2.1 NGEC 处理中 512 个词元限制的讨论

虽然我们的编码人员是在文档层面而非句子层面进行操作，但我们所使用的分类器模型存在技术限制，因此在某些情况下，我们只能处理比整篇报道更短的文本片段。我们用于 NGEC 事件编码的 BERT 系列 Transformer 模型对可处理的文本量设有 512 个“词元”的限制。大多数 Transformer 模型所使用的 WordPiece 分词器（Devlin 等，2019 年）会将文本拆分为单词、数字、标点符号，以及至关重要的“子词片段”，以处理罕见单词和训练过程中未遇到的单词。例如，短语“the cities of Kyiv and Ulaanbaatar.”会生成 11 个词元： [“the”、“cities”、“of”、“kyiv”、“and”、“ul”、“##aan:”、“##ba”、“##ata”、“##r”、“。”], 其中，“##”符号表示在单词内部进行拆分。因此，根据标点符号的数量以及罕见单词或专有名称的数量，512 个词元可能对应少于 450 个单词。我们在预处理文本、去除日期行及其他系统性无关材料后，仅使用文本中的前 1024 个字符，以此近似实现 512 个词片段的限制。[8]

2.2.2 预处理筛选

虽然我们最初预计 Transformer模型会消除对文本内容进行预过滤的必要，但事实证明并非如此，至少有两个原因。首先，正如下文将指出的，Transformer模型往往会过度泛化（这是大语言模型的一个普遍问题）；鉴于我们可用的训练案例数量相对较少，而且未必具有代表性，它们尤其容易出现这一问题。其次，过滤器和

[8]我们目前正在更精确地实施这一限制，一方面尽可能纳入 512 个词元所能容纳的文本，另一方面也确保事件输出中包含确切的编码文本。实际上，512 个词元的上限并不会造成太大限制。在抽取的 20,000 篇故事文本中，72% 的文章短于 512 个词元；在其余案例中，有 50% 的未编码字符大约相当于编码文本字符数的 2/3 到 100%；其余案例则呈现出很长的长尾。

预处理步骤的计算成本相对较低，而在这一阶段可以省去的任何内容，都无需再经过成本更高的事件和实体指派步骤。

尽管项目中使用的源文本相对干净，但 20+ 年度积压文件涉及来自数百个来源、在各种系统下积累的数千万篇文本。因此，我们采用了各种常规的过滤方法——主要是字符串模式匹配和基于示例的支持向量机——从文档中移除日期行、编辑评论及其他无关信息。我们还会检测并移除某些类型的故事文本，包括一次报道多个故事的“综合”文章、财经报道，以及我们检测到主要涉及犯罪的报道。有关这些步骤的更多细节见附录。我们还针对非事件问题引入了一种变通方案，即报道声称某个事件并未不 发生。我们使用 spaCy 的词性标注器识别否定动词的实例，并从报道中移除这些句子。我们希望用一个更先进的系统替代这一系统，使其能够更直接地处理否定，例如返回带有“否定”标签的事件，或生成适当考虑否定的事件（例如，“将不签署该条约” $\rightarrow$ 拒绝）。

2.2.3 分类模型的估计与选择

完成筛选后，模型估计和分类的过程相当直接。格式整理和预处理步骤会将待分类的故事文本转换为适合训练的格式。随后，我们使用 Hugging Face 的 transformers 库，为每种事件类型分别拟合一个二元分类器。具体而言，我们使用默认超参数，在标注数据上对预训练的 ‘distilbert-base-uncased’ (Sanh et al., 2019*b*) 模型进行 3 个训练轮次的微调。

由于随机拆分样本训练存在随机性——有些训练集/测试集比其他集合更具代表性——而且模型拟合本身具有随机性，因此我们针对每个类别拟合 8 个模型，然后选出一个最接近反映整个集成模型分类表现的“共识模型”。随后，在实际分类中使用这个“共识”模型。

所有估计均采用正例与负例训练案例数量为 1:1 的“平衡”比例。由于没有任何类别的实际出现频率达到 50%，这一决定是否会产生重要影响尚未可知。[9]

2.3 模式与语境分类

识别出报道中的事件后，我们会再进行两个分类步骤：如果所分配的事件类别包含模式，就识别其模式；如适用，还会识别**语境**，以说明事件发生的“原因”。Plover事件本体将模式概念化为各事件的“方式”。例如，识别出指控事件后，我们还要确定指控采取了何种形式（指称、不赞成或调查）。为此，对于上一步在报道中识别出的事件所包含的每种模式，我们都会应用一个针对该模式的二元分类器。同样，所有**语境**的二元分类器都会应用于至少产生一个事件的所有文本；**语境** 独立于事件类别。[10] 这些分类器的开发总体上遵循与事件分类器相同的模式，但我们发现，支持向量机的运行时间远短于Transformer模型，且不需要图形处理器，因此已足够胜任，实施和编码也能快得多。

与事件类别一样，我们估算了一组模型——在最近一次迭代中为16个，模式的估算约需半小时，语境所需时间则按比例更少；随后选取F1 分数低于1.0的最佳模型。[11] 除极少数例外，准确率指标均远高于经过充分训练、能够持续工作的人工编码员所能达到的0.70至0.80范围，而且精确率和召回率指标通常相当均衡。表A2和A3展示了这些分类器的表现，以及训练集规模的巨大差异；对于常见的

[9]关于标注过程的更多细节，请参阅 B 节。[10]我们所标注的数据的性质决定了必须使用二元分类器，而非多类分类器。详情请参阅 B 节。[11]最后这一决定是出于对过拟合的担忧，因为随机训练集/测试集划分过于简单：这种担忧可能合理，也可能不合理，今后可以重新审视。

行为，例如**经济**、**军事**和**外交**，以及强制逮捕、抗议-示威和请求-协助等类别，训练集规模相当大；但对于其他一些类别，训练集规模则非常小。[12] 再次说明，这些指标来自分样本测试，而非依据第5节所述、针对大型独立验证集的测试；不过，这些表格中的内容并未显示该方法论存在严重不足。

2.3.1 后处理事件、模式和语境

最后，在初次部署我们的编码人员时，我们注意到模式和语境的假阳性编码数量很高。虽然我们希望通过更好的训练数据和模型来消除这些问题，但目前我们使用一个辅助关键词过滤器，去除一些模式和情境标签。更多讨论见 C.1 节。

2.4 识别　　使用问答模型识别事件属性

识别出事件及其模式后，我们会识别参与该事件的行为体。Plover 本体为事件定义了以下“属性”：

- 施事者：实施该事件的实体。
- 受事者：根据事件类型，作为该事件的接收者、目标、受害者或受益者的实体。
- 地点：事件据报告发生的位置
- 日期：事件据称发生的时间。

具体而言，我们希望提取文本中报告某个事件的施事者、受事者、地点和日期的短语。图2展示了我们期望输出的一个示例。请注意，一份文档可能报告多个事件，我们会提取与每个事件的各项属性相对应的文本。

[12]许多袭击模式的样本量较小，部分原因是编码环境造成的：这项工作是在我们建立街头犯罪过滤器之前进行的，当时有效案例的产出率相当低，不过，袭击-绑架和袭击-暗杀等某些模式在整个语料库中本来就很少见。

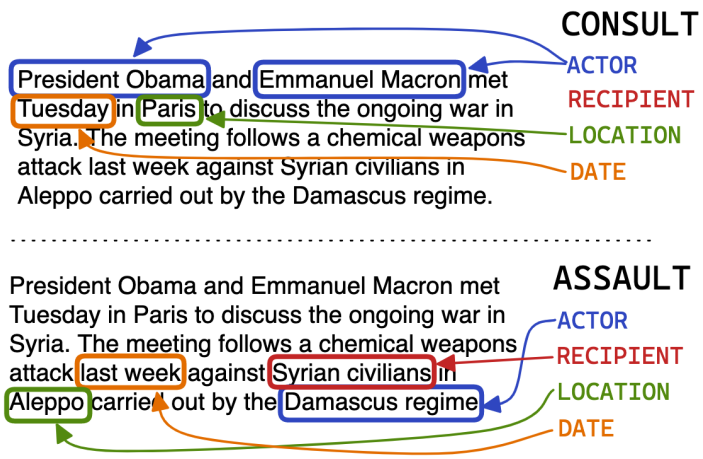

图2：属性识别示例。请注意，该文档包含两个独立的事件（*CONSULT* 和袭击），每个事件都有自己的属性。*CONSULT* 事件有两个施事者，没有接收者；本体允许某些事件出现这种情况。

为了确定每项属性对应的文本跨度，我们采用抽取式问答（QA）框架。自然语言处理中的抽取式问答模型接收（1）一段文本和（2）一个用自然语言撰写的问题，并从该文本段落中返回最能回答该问题的文本跨度。我们针对每种事件-模式组合编写一组问题，询问其施事者、受事者、地点和日期。表1展示了 PROTEST 的两种模式对应的示例问题。对于某些事件-模式，我们会编写多个问题，以便更好地引出正确的文本跨度。我们还会编写一些问题，使用先前问题的答案；这会大幅增加编码时间，但也能显著提高准确率，尤其是在识别受事者方面。

请注意，在处理流程的这一阶段，我们的事件和模式分类器已经识别出文档中存在事件—模式，但尚未确定报道该事件的具体文本段落。采用问答方法识别属性跨度的一个优势在于，问答模型能够处理新闻报道中常见的结构：首句以一般性术语描述事件，后续句子则提供更多细节。问答模型通常更倾向于识别实体更具体的提及，也就是说，它可以从报道的第二句或第三句中提取信息，而不是返回首句中的一般性行为体。这对于事件地理定位尤其有用，因为更

事件	属性	问题
抗议-示威	施事者	“谁举行了示威？”
	施事者	“谁针对{recip text}举行了示威？”
	受事者	“示威的目标是谁？”
	受事者	“{actor text} 针对谁进行了示威？”
	地点	“示威在哪里举行？”
抗议-暴乱	施事者	“谁参与了骚乱？”
	施事者	“谁对{recip text}发动了骚乱？”
	受事者	“骚乱针对的是谁？”
	受事者	“{actor text}对谁发动了骚乱？”
	地点	“骚乱发生在哪里？”

表1： 事件/模式组合的示例问题。请注意，每种事件-模式组合可以包含多个问题，而且问题可以使用先前问题的答案。

详细的地点通常不会在导语句中提及。例如：

> 一名塔利班官员称，周三，阿富汗北部一所宗教学校遭炸弹袭击，造成至少10名学生死亡。听到爆炸声的一名当地居民告诉美联社，爆炸发生在位于艾巴克的圣战宗教学校，时间大约是下午祷告时段；艾巴克是萨曼甘省首府。[13]

理想情况下，我们希望将“艾巴克”识别为事件地点，而不是第一句话中较笼统的“阿富汗北部”。

我们首先使用在问答数据集上微调的现成 Transformer 模型（Rajpurkar、Jia 和 Liang，2018年）。然而，在包含约 300 个正确事件属性跨度的验证集上评估模型性能时，现有模型的准确率相当低（F1分数介于 0.09 至 0.54 之间，具体取决于模型和属性）。因此，我们又使用新收集的 4,823 条标注以及针对事件专门设计的问答，对一个模型进行了进一步微调。在这些新样例上继续微调后，

[13]Rahim Faiez，《塔利班：阿富汗宗教学校遭轰炸，10人死亡》，美联社，2022年11月30日。https://apnews.com/article/afghanistan-bombings-taliban-3dbcb584545e8f8a2ac69185a4953a08

Span	属性	问答模型得分
一群印度教民族主义者	施事者	0.433
	施事者	0.502
	受事者	0.502
印度教民族主义者	施事者	0.452
德里	LOC	0.939
德里上周	DATE	0.751
穆斯林商店	施事者	0.131
	施事者	0.131
	受事者	0.179
	受事者	0.755
	受事者	0.131
	受事者	0.103

表 2： 问答属性模型的输出示例。问答分数将被求和，并使用最优线性和分配方法将文本片段分配给属性。

我们得到 行为体的 F1 分数为 0.86，受事者为 0.73，地点为 0.91。[14]

作为整个流程的示例，请看这个句子：“上周，印度教民族主义者在德里骚乱，袭击了穆斯林商店。”我们通过三轮提问来识别事件属性：

1. 第一轮使用通用问题（“谁袭击了某人？”，“骚乱针对谁？”）

2. 第二轮使用步骤 1 中检测到的行为体：（“印度教民族主义者袭击了谁？”）

3. 最后一轮使用步骤 2 中检测到的受事者：（“谁袭击了穆斯林商店？”）

由于我们针对每个属性提出多个问题，并进行了三轮提问，因此会得到许多可能的文本片段–属性组合，每种组合都有一个置信度得分

[14]微调后表现最佳的模型是 roberta-base-squad2。该模型此前已在 SQuAD 2 上进行微调，并进一步在我们的新数据上进行了微调。

该得分由问答模型给出。我们希望为每个属性分配最佳文本片段，并避免同一文本片段填充多个属性。为此，我们对每个文本片段对应的问答模型得分求和，然后使用匈牙利算法（Kuhn，1955）为每个属性找到最优文本片段。

2.4.1 事件抽取问答研究的更广泛文献

政治学家长期以来将事件抽取描述为回答“谁在何时何地对谁做了什么？”这类问题，因此，问答模型自然有助于事件抽取。自然语言处理领域近期的研究已经将问答模型应用于事件抽取。FitzGerald 等人（2018年）提出使用问答模型进行语义角色标注，这项任务与事件抽取类似。此后，Du 和 Cardie（2020年）从问答框架的角度对事件抽取进行了概念化。未来的研究可以更充分地利用文章中存在的其他语境信息。人们通常会用简短的政治角色描述来介绍不熟悉的政治行为体。由于难以识别这些文本片段，当前的编码人员不会尝试识别行为体的同位语或后置修饰语，但自然语言处理领域的持续研究或许能为此提供有益的起点（Kang 等人，2019年）。

2.5 使用神经相似度将实体链接至维基百科和GeoNames

经过前面的步骤，我们现在已经从文本中识别出事件，以及对应于行动施事者和接收者的短语。回到贯穿全文的示例，我们希望将提取出的实体提及（“奥巴马总统”“大马士革政权”）消歧为唯一标识符，从而将同一人物的不同提及（例如“奥巴马总统”“Barry Obama”）归并为同一个规范名称（“巴拉克·奥巴马”）。在自然语言处理文献中，这通常称为实体消歧。正确完成这一步，就能大幅减少数据中不同施事者字符串的数量，也让研究特定人物或组织变得容易得多。我们使用维基百科作为

外部知识库和规范实体名称的来源。对于专有名称，我们采用下文所述的方法。一般性术语（例如“叙利亚平民”）不会链接到维基百科，处理这类实体的方法见第 2.6 节。

2.5.1 实体消歧的前期研究

以 ICEWS 编码器或 PETRARCH2 为代表的早期事件编码器，不会为每个检测到的实体返回一个唯一的规范名称。相反，它们会将行为体提及直接映射到更高层级的类别，例如“政府”或“商业”（下文将对此进行介绍）。因此，无法在类别内部进行更细致的分析，例如考察一国军队中的哪些行为体彼此互动；这也限制了以往事件数据集在研究官僚政治、立法机构内部的审议、同一冲突中不同叛乱组织之间的互动，或更广泛的比较政治学时的用途。

2.5.2 创建自定义维基百科索引

我们选择创建维基百科的本地离线版本，而不是直接查询 wikipedia.org。[15] 建立自己的索引可以避开维基百科施加的频率限制，同时让我们能够编写自定义搜索查询，而不必依赖维基百科内置的搜索栏功能。此外，这也使我们的处理流程无需任何外部网络连接即可运行，这在某些情况下是必要的。

我们下载了所有英文维基百科条目的数据转储，首先从中提取所有重定向。重定向是从实体名称的别名指向其维基百科规范条目的硬编码链接。例如，伊斯兰国的维基百科页面有 100 多个重定向，其中包括新闻报道中使用的许多术语，例如“ISIL”“伊拉克和叙利亚伊斯兰国”和“Da‘ish”。获取所有重定向后，我们将每篇条目解析为一种简化格式，其中包含

[15]设置离线维基百科索引的代码位于此处：https://github.com/ahalterman/NGEC/ tree/main/setup/wiki

页面标题、指向该页面的所有重定向、从首段自动提取的别名、页面分类，以及维基百科对该条目的一句话摘要。我们还会提取并解析信息框（如果有），其中包含诸如担任政治职务者的职务名称和任职日期等信息。为节省篇幅，我们保留条目的引言段落，但舍弃其余文本。我们跳过类别页面和消歧义页面。

我们将格式化后的文章载入 Elasticsearch 索引，从而能够按任意字段高效查询文章。剔除重定向页、消歧义页和分类页面后，我们的离线版维基百科共存储了 7,093,722 篇文章，占总计 22,694,085 个页面的一部分。

2.5.3 神经相似度与基于维基百科的行动者编码

要使用待解析的施事者文本查询维基百科索引，我们首先在维基百科索引上进行模糊搜索，在页面名称、别名和重定向中进行查找。我们使用模糊搜索，以更好地处理姓名的不同音译形式、昵称或不完整姓名。[16] 因此，我们通常会从维基百科获得许多可能的匹配结果。为了选出最佳结果，我们结合使用启发式规则和神经相似度方法。首先，我们检查施事者文本是否与某个条目标题或重定向存在唯一的精确匹配；如果有，则直接返回该结果。如果多个维基百科条目存在匹配，我们就采用神经相似度方法：将查询词和维基百科条目的信息嵌入固定维度的编码器中，并计算二者之间的相似度。为了训练模型，我们使用此前收集的维基百科重定向数据。通过重定向词典，我们可以构造示例对，其中一对是真实重定向（“Joseph Biden” $\rightarrow$ “Joe Biden”），另一对是错误重定向（“Joseph Biden” $\rightarrow$ “Jill Biden”）。构造负样本对时，我们会采样相似度较高但不正确的页面名称，以增加任务难度并确保更好的

[16]此流程的代码可在 https://github.com/ahalterman/NGEC/blob/main/NGEC/actor_resolution.py 获取

模型。我们采用对比学习目标（Chopra、Hadsell 和 LeCun，2005）训练神经模型，为每个页面名称创建嵌入，使其与真实重定向相似、与错误重定向相距较远。使用神经相似度方法，可以识别姓名的不同拼写形式，还能处理包含维基百科标题中可能不存在的组成部分的查询姓名，例如头衔、父名、中间名或姓名首字母。

在这一流程结束时，理想情况下，我们会为事件中提及的每个人物或组织获得一个规范维基百科标识符。我们使用 Hoffart 等人（2011）提供的命名实体及其对应维基百科页面数据集来评估该方法。当我们的结果中包含正确的维基百科条目时，我们有 84.0% 的概率识别出正确条目。Hoffart 等人（2011）没有报告原始准确率数值，但提供了精确率-召回率曲线；其最佳方法的精确率约为 0.85，召回率约为 0.70。

2.5.4 地理定位与日期解析

我们采用不同的流程，将地名提及解析为外部知识库中的规范条目。我们使用 GeoNames（Wick 和 Boutreux，2011）作为该外部知识库。Halterman（2023）对地理定位流程进行了更详尽的介绍，这里我们简要概述。我们使用 spaCy 的命名实体识别系统（Honnibal 和 Montani，2017）识别新闻报道中的所有地点。[17] 对于每个地名，我们都会查询存储在离线 Elasticsearch 索引中的 GeoNames。如果出现多个匹配项，我们会使用一个模型选出最佳结果。该模型会综合考虑地名的嵌入表示、文档中的其他地名、文档内容、原始地名与 GeoNames 所列别名之间的相似度，以及提取出的地名所预测的地物代码和国家。神经模型使用数千个标注示例进行训练，以对结果排序，并返回排名最高的匹配项及其置信度。为了确定报道中的哪些地点是每个事件的事件地点（Halterman，2019*b*），我们会找出与

[17]具体来说，我们使用 spaCy 3.4 搭配基于 Transformer 的 en_core_web_trf 模型，并识别所有标记为“地缘政治实体”或“地点”的实体。

问答模型识别为事件地点的地点答案重叠的地理定位地名。这样得到的事件地理编码较为保守，因为只有当地理定位实体与前一个问答步骤给出的地点答案重叠时，我们才会将其识别为事件地点。

为确定事件发生的日期，我们使用问答模型识别出的文本跨度，并结合基于规则的系统和文章的发布日期来确定事件日期。[18] 通过识别事件发生在过去某个日期（“星期二”“上周”），并从报道的发布日期倒推，我们可以更准确地判断事件发生的时间。以往的事件数据系统通常将报道的发布日期作为事件日期。然而，报道中的许多事件发生在发布日期的前一天或更早。若根据整篇报道对事件进行编码，问题会更加严重，因为文章末尾常会为提供语境而报道历史事件（Althaus、Peyton 和 Shalmon，2022年）。通过将“去年11月”或“2018年”等短语解析为过去的某个日期，我们有助于减少将历史事件归到当前时间的情况。

2.5.5 局限性与未来工作

在自然语言处理领域，将实体提及对应到维基百科条目是一项既具挑战性又由来已久的任务（Hachey 等，2013年），未来的工作可以提高我们方法的准确率。当前方法的一项主要局限是，在解析实体时，我们没有利用更广泛的报道语境。例如，如果不结合整篇报道中的信息，“众议院”这一提及可能无法对应到正确国家的立法机构。要纳入这些语境信息，就需要修改模型，并补充训练数据。

2.6 实体分类

作为政治学家，我们通常希望将行为体归并为更宽泛的行为体类别，而不是直接使用行为体（消歧后）的名称。例如，我们

[18]具体来说，我们使用 Python 的 dateparser 库。

原始文本	**已消歧行为体**	**编码后的行为体**
奥巴马总统	巴拉克_奥巴马	美国 政府（2009年-2017年）
五角大楼官员	五角大楼_	美国 军方
叙利亚平民 –	叙利亚	平民
大马士革政权 –	叙利亚	政府

表 3： 实体消歧与行动者编码示例。人物和组织的专有名称会被归并为规范形式（例如其维基百科文章标题），而一般性术语则没有对应的维基百科条目。随后，行为体提及会进一步汇总为 *GOV*、*MIL* 或 *CVL* 等高层级代码。

可能希望将某个国家的所有立法者，以及对立法机构本身及其委员会的提及，归入单一的“立法”施事者，以便进行分析。这一步在自然语言处理文献中较不常见，我们将其称为实体分类。表3展示了一个示例。

2.6.1 对行为体分类的既有研究

大多数早期的自动化事件数据系统都使用词典对行为体进行编码，其方式与使用词典对事件类型进行编码类似。基于词典的行动者编码系统会将提取出的行为体与一份预先指定的大型行动者模式列表进行比对，并返回匹配模式对应的代码。使用基于词典的系统所带来的缺点如上所述；但鉴于可能的模式数量庞大，这些缺点在行动者编码中尤为突出。例如，如果词典中没有“示威者”，简单的字符串匹配方法就无法为其编码。专有名称带来的问题更加突出：如果行为体词典中唯一的模式是“巴拉克·奥巴马”，那么“奥巴马总统”就不会被编码。此外，专有名称词典还需要一种基于日期的机制，以便在某人的职务发生变化时返回相应的角色。虽然已有一些自动更新词典的技术（例如 Solaimani 等（2017）；Rad-ford（2019）），但要持续维护一套全面的行为体及其对应代码，仍需投入大量人工劳动。

2.6.2 神经相似度行动者编码

我们采用神经相似性模型为提取出的行为体分配代码。完成上述步骤后，我们得到两类行为体提及：一类是“叙利亚村民”或“政府官员”等泛称行为体，另一类是“乔·拜登”等如今已关联到其维基百科页面的专有名称。对于这两类情况，我们都希望为每个行为体分配一个更宽泛的行为体类别代码。

我们从一个包含通用描述、职务和头衔的文件开始，其中每项都对应一个 PLOVER 行为体代码。例如：

村庄 [~平民]

农业部_农业 [~政府农业]

国防_部长 [~政府军方]

反叛者_首领 [~REB]

情报_服务 [~间谍]

我们从先前的研究中整理出这份列表（Schrodt，2009年；Norris、Schrodt 和 Beieler，2017），并作了一些补充，同时修改了代码，以反映新的 Plover行为体本体。以往基于词典的事件编码器要求词条与文本片段精确匹配，才能生成编码。因此，现有文件包含约 2,000 个独立词条，包括替代拼写和细微变体（例如“防御”与“防卫”、“部长 for”与“部长 of”）。尽管文件规模庞大，基于词典的系统在召回率方面仍存在问题（假阴性）。我们不再采用精确匹配，而是实现了一个神经相似性模型来处理替代术语和拼写。我们使用句子转换器模型（Reimers和Gurevych（2019））为文件中的每个词条生成嵌入，该模型能够生成适用于神经相似性任务的定长嵌入。

对于“叙利亚平民”等泛化行为体，我们首先检测是否提及国家或居民称谓，以确定其所属国家，并将其从文本片段中移除。然后，我们使用相同的神经相似性模型对剩余片段进行嵌入，并计算余弦相似度。

计算施事者嵌入与文件中各个嵌入之间的相似度，以确定最接近的匹配项。如果相似度高于设定的阈值，则返回该条目的代码，作为该文本片段的施事者代码。

对于链接到维基百科的命名实体，处理过程也类似。获取某位具名行为体的维基百科页面后，我们会提取页面中描述该行为体身份的内容。对于维基百科页面侧边栏中带有信息框的公职人员，我们会提取与新闻报道日期相对应的职务名称。[19] 对于其他类型的行为体，例如组织、非公职人员或居民点，我们会从页面的单句摘要、信息框类型，或在不得已时从引言段落中获取信息。获得该行为体的职务名称或简短描述后，我们会进行上述相同的神经相似度比较，并返回与维基百科上该人物或组织描述最匹配的代码。

3 与 BFRS 巴基斯坦政治暴力数据集的比较

现在，我们来讨论如何使用不同于我们用于编码 Polecat 数据集的 Plover 本体的事件本体，通过我们的处理流程生成定制事件数据。许多研究人员，尤其是从事跨国研究或国际关系研究的研究人员，希望获得像 Halterman 等人（2023）介绍的那种开箱即用的全球事件数据集；但其他研究人员有特定需求，使用定制事件数据集最能满足这些需求。我们的无词典处理流程的一大优势在于，研究人员可以轻松修改该处理流程，以生成定制事件数据集，包括采用不同于 Plover 的编码本体的数据集。基于词典的编码器几乎无法完成这类修改，因为必须从头编写整部词典。

[19]处理行为体随时间变化的角色，是面向历史文本的事件编码器的一项关键组成部分。

BFRS 数据集是一个关于巴基斯坦政治暴力的人工编码事件数据集，涵盖 1988 年至 2011 年 11 月（Bueno de Mesquita 等人，2015年）。这 29,970 起事件是根据《黎明报》人工编码的。该报是巴基斯坦主要的英文报纸；数据集包含 12 种事件类型（恐怖主义、骚乱、无人机袭击、常规袭击、游击袭击等），以及有关责任方、所报告原因、是否有警察或军队参与、死亡或受伤人数，以及事件日期和地点的信息。

BFRS数据集是一个很好的案例，可用于研究本文介绍的工具是否适用于新推出的全球 POLECAT 数据之外的其他数据集。BFRS数据集对事件的概念化方式与我们采用的方式不同。若干事件在定义中已包含施事者或接收者（例如，针对安全部队的游击袭击、安全部队对非国家战斗人员的袭击等），同时还会另行报告“责任方”。“责任方”信息采用的分类方案与 Plover 方案不同。该数据集的内容也超出了 Plover 本体的范围，还纳入了事件的其他信息，例如死亡或受伤人数，以及事件是否成功实施或被拦截。我们展示了如何轻松修改处理流程，以生成符合 BFRS 格式的数据。

由于难以获得涵盖整个 BFRS 数据集时期的《黎明报》原始新闻文本，我们选择将标注人员撰写的事件简述作为文档，而非使用原始新闻文本。这样做大大简化了我们的任务，因为简述包含了事件的详细信息，且没有原始报道中可能出现的无关信息。这意味着我们的模型评估结果会高估其性能。将模型应用于目前可获得的部分原始文本，是验证我们方法的重要后续步骤。

3.1 事件分类

为模拟人工编码过程，我们从BFRS数据集中随机抽取1,000个已标注事件（约占总数的3%）。随机样本中，某些罕见事件类别的示例相对较少

，例如暴力威胁和警察／军方选择性暴力；但这与人工标注文档的过程相符，因为抽取罕见类别进行标注是一个难题（Miller、Linder 和 Mebane（2020））。我们使用这1,000个抽样事件训练了一个高效分类器，该分类器对现有的Transformer模型进行微调。具体而言，我们训练了一个SetFit模型（Tunstall等人（2022）），用来微调句子Transformer模型paraphrase-mpnet-base-v2（Reimers和Gurevych（2019））。[20]对于训练语料库中的每篇文档，SetFit都会抽取同类和不同类的文档，并使用对比学习目标生成嵌入，使同类句子的嵌入彼此相似，而与其他类别的嵌入不同。SetFit的高效学习能力使其仅使用数千万参数的模型和几十个示例，就能达到与使用数十亿参数、在数千个示例上训练的模型几乎相同的性能。该模型在图形处理器上训练约耗时5分钟，性能比简单的嵌入回归模型高出约3个百分点。图4展示了我们从BFRS验证集的事件描述中识别BFRS事件类型的能力。

对于大多数事件，分类F1分数尚可（高于0.8）。对于其他一些类别，包括安全部队遭受的游击袭击、暴力威胁以及“其他”类别，则需要进一步标注。根据开发NGEC事件编码器的经验，我们建议采用主动学习方法来决定标注哪些文档：使用初始模型找出该模型最不确定的示例（Miller、Linder 和 Mebane（2020））。

3.2 BFRS属性识别

虽然BFRS事件数据集并未按照PLOVER本体的方式记录行为体和受事者，但仍会报告事件的“责任方”和地点。我们可以针对BFRS事件定义提出新的问题，使用现有的问答模型获取地点和责任方的信息。编写新的

[20]我们在上述行为体分类步骤中使用了类似的句子嵌入模型。

事件类型	精确率	召回率	F1分数	N
常规 袭击 on 军事/准军事/警察/情报	0.46	0.40	0.43	77
恐怖主义	0.77	0.76	0.77	406
暴力政治示威	0.83	0.83	0.83	247
军事/准军事/警察 袭击 on 非国家战斗人员	0.62	0.87	0.73	69
暗杀	0.86	0.81	0.84	616
暗杀（无人机袭击）	1.00	0.86	0.92	14
其他	0.21	0.30	0.25	129
riot	0.83	0.76	0.79	193
暴力威胁	0.18	0.25	0.21	12
对国家的袭击	1.00	0.83	0.91	81
对军事/准军事目标的游击袭击/警察/情报	0.48	0.57	0.52	107
军事/准军事/警察选择性暴-力	0.62	0.29	0.39	45
准确率			0.73	1996
宏平均	0.66	0.63	0.63	1996
加权平均	0.75	0.73	0.73	1996

表4： 我们的 *BFRS* 事件分类器在 *1,000* 篇随机抽样的文档上训练后的验证集准确率（ *N=1,996*）。该分类器使用 *SetFit* 对 *paraphrase-mpnet-base-v2* 进行微调。

为这12种事件类型拟定问题，对于熟悉次国家冲突的研究人员来说相对容易，耗时不到一小时。由于我们原有的问答属性模型是在多样化的文本和事件类型上训练的，因此即使面对新的BFRS事件类型，它也表现良好。表5展示了问答属性模型对取自BFRS数据集的四段事件描述所生成的结果。请注意，该模型能够很好地处理被动结构，但有时会遗漏有关行为体的一些细节，例如曾将更宽泛的“持枪人员”选为答案，而不是更准确的“俾路支解放军”。

3.3 BFRS行为体分类

由于BFRS数据集对行为体的分类方式与PLOVER本体不同，我们不能直接沿用现有的通用角色和职务名称文件。不过，创建一个将行为体描述映射到15种行为体类别的新文件同样很简单。我们从BFRS编码手册中提取术语和示例，创建新的行为体分类文件。

3.4 BFRS 地理定位

接下来，我们评估从文本中恢复事件发生地点的能力。BFRS数据集记录了每个事件的地点、乡镇（次级行政区）、县和省。由于BFRS数据集与GeoNames地名录之间存在替代拼写，若不经人工核验，我们就无法直接比较地点名称，因此很难评估最细粒度地点的准确率。作为替代方案，我们评估省级地理编码，结果见表6。由于BFRS数据集中的事件描述并不总是包含原始文章中的地点信息，因此许多事件无法进行地理定位。在29,969个事件中，我们排除了描述中未检测到地点的17,338个事件。如表6所示，地理定位过程大多数时候是准确的，但在省级层面仍会出错。

事件类型	事件描述
常规对安全部队的袭击	三人，其中一名**安全人员**，在.....时受伤
	持枪人员骑着摩托车向位于**库兹-达尔镇**。俾路支解放军发言人 Mirack Baloch 声称对此次袭击负责。
无人机袭击	七名**涉嫌武装分子**被打死，另有四人受伤，当时一架**美国无人机**在**南部安古尔阿达附近发射了导弹**
	瓦济里斯坦
暴力的政治的示威	**当地高校的学生** 发起了一场抗议示威
	而宗教政党又举行了一次示威。 两者均为 com-
	合并为一个，位于**Jogi Chowk**。抗议者要求**店主**去关闭他们的商店。Kariana Bazaar 的一位店主进行了抵抗，而抗议者损坏了他的商店。他们还点燃了轮胎，并拆下了张贴在街道上的电影广告牌。
威胁 of violence	一封信被寄给了 **店主** 在 **查尔萨达的加富尔市场**
	集市据称由**当地塔利班**警告他们不要与之交易
	与女性顾客。
游击式袭击	**三人**遇难，另有二十多人受伤，其中包括
on 安全性	警察，在一名**自杀式炸弹袭击者**引爆一辆皮卡车时受伤
力量	警察局附近的卡车，位于**拉基马尔瓦特的达拉佩祖镇**。一警员在车辆向警察飞驰而来时向其开火。
	车站 ，轰炸机未能抵达车站的墙壁，便在其前方爆炸 。超过三十家商店被夷为平地，附近的房屋也遭到损坏。警察局的墙壁和清真寺也遭到损坏。

表5： *BFRS*数据集中不同事件类型的示例描述，展示自动检测出的 施事者、 受事者和地点 属性。数据集的事件描述字段中很少报告日期。

	俾路支省	*KPK*	信德省	旁遮普省	阿扎德克什米尔	*GB*
俾路支省	**677**	82	64	143	8	13
KPK	136	**791**	84	169	13	6
信德省	542	481	**3083**	888	108	76
旁遮普省	189	550	325	**1898**	77	12
AJK	46	187	48	108	**131**	0
GB	3	3	3	7	0	**29**

表6： *BFRS*数据集中的事件的省级地理定位准确率。真实标签显示在行标签中，预测省份以斜体显示为列标题。在*29,969*个事件中，有*17,338*个事件的描述文本中未检测到地点，因此不纳入此表。

4 一些操作层面的考量

4.1 减少劳动与训练

NGEC 的专项工作始于2021年8月中旬，所有操作代码于2022年3月中旬前交付。相比之下，四个主要的基于词典的编码器为学术出版物生成了所用的数据——KEDS（约1990—1994年）、VRA-Coder（约1992—1996年）、TABARI（约2000—2004年）和ACCENT（约2011—2015年）——每个都需要大约四年的开发时间：通常先搭建可运行的编码平台，再花很长时间逐步完善与之配套的词典；除了 KEDS 之外，它们还都建立在早期项目定制的词典之上。除这些成功项目外，可能还有数量约为其两倍的项目未能成功部署新的编码系统，其中最近的一个是由美国国家科学基金会资助、为期三年、耗资200万美元的项目。[21]

NGEC 的高效主要归功于 Transformer革命，以及我们所需的大多数模型都可通过 Hugging Face（https://huggingface.co/）等文档完善的开源平台获取。Transformer模型在这一应用及许多其他应用中已证明具有出色的适应性和稳健性。例如，我们可以采用基于相同底层预训练模型的同一技术的不同变体，分别将事件作为分类问题处理，将实体作为问答问题处理。虽然在能力方面没有那么引人注目，但 Python 的 sklearn 等开放式机器学习套件让我们能够迅速开展实验，并随后实现支持向量机分类器。

我们还能够利用其他规模庞大的开源资源——尤其是维基百科和 GeoNames 数据库——来解决如何保持词典完整的问题。

[21]此外，也有一些成功的项目实际上属于扩展现有系统的混合方案：其中包括洛克希德的 JABARI——一种基于 Java 的 TABARI 扩展；开放事件数据联盟的 PETRARCH-1 和 PETRARCH-2 编码员——后者是由一名非常优秀的本科实习生在暑期独自完成的项目，利用了斯坦福大学的 CoreNLP解析器和 TABARI 词典；以及各种通常旨在争取美国国防高级研究计划局资助的尝试，例如 IBM 等大型软件公司对为其他用途开发的大规模自然语言处理系统进行修改，以用于事件编码，但这些方案都未得到广泛接受。

且保持最新，否则就需要受过训练的工作人员持续维护。这对于行为体和受体信息尤其重要，因为死亡率、选举、政府内部变动以及新行为体的出现都会使这些信息不断变化。

4.2 计算需求

本项目的计算需求相当高。特别是，由于所需的二元模型数量众多，使用大语言模型处理模式和**语境** 并不现实；即使只是估计事件类别的核心模型（这些模型基于多个估计模型），也需要在 Google Colaboratory 系统上使用图形处理器运行约 36 小时。没有图形处理器几乎无法完成，因为图形处理器能将速度提高 30 到 60 倍，不过最初的概念验证仍可使用普通个人计算机完成。属性步骤的计算成本远高于其他步骤，原因有二。第一，我们发现，与较大且运行较慢的 RoBERTa 模型相比，使用 DistilBERT 或 TinyBERT 这类更小、更高效的模型会造成不可接受的准确率损失。第二，如上所述，我们会分别运行问答模型三次，以利用先前的答案改进问题。我们发现，这是提高受体识别准确率的重要步骤，但会使耗时增加三倍以上。[22] [23]

这些较长的处理时间限制了可进行的实验数量，而每当添加基于大语言模型的组件以提升性能时，处理需求都会显著增加。因此，在条件允许的情况下，我们使用传统的机器学习分类器，例如支持向量机，其运行训练语料库所需的时间仅为几十分钟，而非几十小时。相比之下，基于词典的系统（如 TABARI、PETRARCH 和 ACCENT）虽然在处理大型语料库时也需要耗费一定的时间，但从未成为一个.....

[22]为了提高标注效率，我们只收集了关于模式和语境的二元标注（例如，报道是否具有“性别”语境？），而没有收集涵盖所有模式和语境的完整标注。因此，如果不采用复杂的技术来处理缺失标签，我们就无法训练多类别分类器。[23]由于我们在每一轮中都会提出多个问题，因此，随着我们将上一轮的回答纳入后续问题，可能会形成一种组合式流程，即每一轮提出的问题越来越多。

重要的限制。

我们认为有三种可能的方法可以缓解处理速度缓慢的问题。最简单的方法是利用图形处理器不断提升的速度，以及现成模型带来的效率提升。第二种方法是在处理流程中采用更好的软件工程实践。现代图形处理器能够高效地将多项任务批量处理，但属性模型的批次大小受限于我们针对每篇文章提出的问题数量。虽然复杂性会大幅增加，我们仍可以从多篇文章中收集问题并将其批量处理，从而有可能缩短计算时间。最后，获取更多训练数据后，我们就能使用更小、更高效的模型。我们受益于较大型语言模型能够通过迁移学习适应我们领域这一能力。不过，如果有更多训练数据，我们就能更充分地训练较小的语言模型，使其在我们的任务中表现出色。

不过，整体而言，我们的处理流程很容易并行运行。每篇文章的处理彼此独立，因此我们可以轻松地将处理任务分配到多个图形处理器或机器上。

5 Futu 基准测试与评估数据的未来工作

虽然事后看来这一点显而易见，但仍值得指出：基于词典的模型大多不透明，除非能够访问词典；而在事件编码以及 ACCENT 中涉及的许多行为体方面，我们并没有这类词典。相比之下，只要知道基本特征，就能有效地对任何基于示例的系统的模型进行逆向工程：不需要确切的训练案例，只需一组可与已知输出相匹配的相似输入（换言之，与基于词典的系统不同，不需要描述事件的确切 句子，只需一段看起来类似于大语言模型所用文本的文本：仅凭日期、行为体以及在事件记录中编码的位置，就很容易找到这些相似的故事文本）。政府 iDATA 中明确提供了数百万组这样的输入—输出对，其中包含源文本；而且，由于除少数情况外，通常很容易找到与特定事件相对应的存档文本，因此也可以轻易地从公开数据中推断出这些配对。尽管实现

神经网络的方法不计其数，但这些方法很可能会达到与我们相当或更高的准确率。

尽管如此，如果能有一套用于评估系统的标准基准案例，将会很有帮助。这些案例可提供指标，供学术计算机科学家用于评估新改进，并在取得成功时发表相关成果。计算机科学文献通常兼具时效性与开放获取的特点，并着重于在既定基准上超越先前的最先进水平，因此实际上可以免费提供训练有素的劳动力，探索对系统进行改进的可能性。迄今为止，我们对模型的“超参数”都只采用默认值——这些参数是无法从数据中估算出来的模型特征——所以它们大概还有改进的空间。此类改进通常幅度不大，准确率只提升几个百分点，但终究仍是进步，偶尔也会出现重大突破。计算机科学界也可能会尝试新的模型和方法，并提醒我们有哪些改进机会。

最后，计算语言学界也对这一主题感兴趣：2020年和2021年的语言资源与评测会议都举办了一个参与人数众多（$>$100 名参与者）的研讨会，主题是“从新闻来源自动提取社会政治事件”——也就是事件数据——并吸引了不少参与者，还举办了一些建模竞赛。

为了在系统的各个层面充分利用当前的机器学习环境，我们理想情况下需要一个大型且具有代表性的“金标准”[24] 验证数据集，其中包含完整的编码（不同于我们各个训练集中的逐组件选择），涵盖来自代表性语料库的大量案例。[25] 如果具备这样的数据集，就可以进行一系列额外的改进并采用更多指标，尤其包括：

[24]有一个青铜标准我们就很满意了。. .[25]这也不同于 TABARI 和 PETRARCH 系列编码器开发过程中使用的那个 [不]知名（但开源！）的《指环王》验证集：该验证集确保解析器编码器的各个组件能够持续正常工作，但并未系统地检查生成的数据。BBN 无疑也为 ACCENT 做过类似的验证，但与 ACCENT 动词模式一样，这并未开源。

- 针对训练案例中的正例/负例平衡等超参数进行实验

- 持续评估训练集中的异常案例

- 校准分类得分等级

- 估算（并降低）假阴性水平（请注意，甚至 ICEWS 的这一水平也尚不明确）。参见 Halterman 等（ 2021 ），了解估算事件数据召回率所面临困难的讨论。

更一般地说，如果有参照集，我们还可以轻松利用学术界和业界广泛提供的各种开源工具。

问题一如既往在于成本，因为至少需要人工标注来审核这些案例。不过，项目可以由多个机构共同开展（当然，还需进行近实时自动检查，以确保不会出现个别编码员或机构效应；若不加监控，这类效应几乎不可避免，而遗憾的是，多个广泛使用的人工编码数据集中都曾在事后才发现这类问题），也可以结合本科生和研究生编码员、专业编码员以及用户群体的反馈。在理想情况下，无论对编码结果作出肯定还是否定的评价，都不应被遗漏；也就是说，当用户说“这是我见过的最愚蠢的编码！”时，这是一条至关重要且相当重要的数据，我们希望将其收集起来，尽管要建立相应的系统并不容易。

说到验证集，越大越好。对于“这要花多少钱？”这个问题，理想的回答是那个令人头疼的反问：“你有多少钱？”不过现实中，将现有分类器（或标准分类器集成）用于获取容易处理的部分（这类输出大多为肯定项，通常标注起来没那么费力），由多个研究机构和非政府组织分头协作，并收集用户反馈，可能只需相对适度的投入就能完成。参考数据需要不受知识产权限制——根据美国版权法，

由于研究适用“合理使用”例外，而且这些示例在原始语料库中所占比例微乎其微，因此这些数据很可能已经符合要求；不过，也可以像目前涉及人类受试者的公开研究数据中常见的做法那样进行“匿名化”转换，或者将这些案例输入基于大语言模型的系统，由其转换为合成案例，使其对训练产生的影响与原始案例无法区分（同样，我们可以用验证集来评估这一点）。这个参照集实际上将定义 Plover，未来多年都可供冲突研究、计算语言学和机器学习领域的研究人员使用：“理论来来去去，好的数据集却能流传至永远。”

6 结论

本文介绍了一种新型端到端事件编码器 NGEC，它突破了以往基于词典的编码器的局限。它充分利用了自然语言处理领域的最新进展，包括用于分类的 Transformer 模型、用于事件属性识别的问答模型，以及神经相似度技术。我们借助维基百科和 GeoNames 等现有资源，免去了维护大量行为体词典的需要。我们已部署该处理流程，生成了新的全球事件数据集 Polecat，旨在取代 ICEWS 数据集。

本文所述的各个步骤均为模块化设计，可单独使用，以满足研究人员的特定需求。例如，我们的主动学习标注方案和基于 Transformer 的分类器适用于希望进行文档分类的研究人员。我们的地理解析技术适用于需要从文档中抽取并消歧地名、但不进行完整事件抽取的研究人员，也适用于拥有地点半结构化数据、希望将其解析为坐标的研究人员。我们的维基百科查询流程用途广泛，适用于需要将政治行为体的多个提及统一解析为单一规范身份的研究人员。

开发该处理流程的一个主要目标是降低生成新的定制事件数据集的成本。研究人员通常对事件类型或施事者分类感兴趣。

现成数据集中不存在的情况，但手动编码事件数据或开发基于词典的事件编码器所涉及的成本，对大多数项目来说都高得令人望而却步。我们证明，我们的处理流程可以轻松改造，用于编码采用不同于我们所用 Plover 本体的事件本体的事件数据。我们希望，本文实现的技术和代码能够大幅降低创建新事件数据集的成本。

参考文献

Althaus, Scott, Buddy Peyton 和 Dan Shalmon。2022年。“用于验证事件数据的总误差方法。”《美国行为科学家》66(5):603–624。

Althaus, Scott L, Buddy Peyton 和Dan A Shalmon。2018年。“六种事件数据处理流程中的博科圣地活动时空动态。”*APSA*现代事件数据开发与分析小型会议。

Boschee, Elizabeth, Jennifer Lautenschlager, Sean O’Brien, Stephen M Shellman, James Starz 和 Michael D Ward。2015年。ICEWS编码事件数据。载于哈佛数据宇宙，第*9*版。Brathwaite, Robert 和 Baekkwan Park。2018年。“宗教暴力的测量与概念方法：运用自然语言处理生成宗教暴力事件数据。”《政治与宗教》第1–42页。

Bueno de Mesquita, Ethan, C Christine Fair, Jenna Jordan, Rasul Bakhsh Rais 和 Jacob N Shapiro。2015年。“衡量巴基斯坦的政治暴力：来自BFRS数据集的见解。”《冲突管理与和平科学》32(5):536–558。

Chopra, Sumit, Raia Hadsell 和 Yann LeCun。2005。以判别方式学习相似性度量，并将其应用于人脸验证。载于*2005 IEEE*计算机学会计算机视觉与模式识别会议 *(CVPR’ 05)*。第 1 卷，IEEE，第 539–546 页。

Devlin, J.、Ming-WeiChang、Kenton Lee 和 Kristina Toutanova。2019年。BERT：用于语言理解的深度双向变换器预训练。载于 *NAACL-HLT*。

Douglass, Rex W., Thomas Leo Scherer, J. Andrés Gannon, Erik Gartzke, Jon Lind- say, Shannon Carcelli, Jonathan Wilkenfeld, David M. Quinn, Catherine Aiken, Jose Miguel Cabezas Navarro, Neil Lund, Egle Murauskaite 和 Diana Partridge。2022年。“介绍 ICBe数据集：从国际危机叙事中进行具有极高召回率和精确率的事件抽取。”*https://arxiv.org/abs/2202.07081* 。

Du, Xinya 和 Claire Cardie。2020年。“通过回答（几乎）自然问题进行事件抽取。” *2020*年自然语言处理实证方法会议论文集（*EMNLP*）第 671–683 页。

FitzGerald、Nicholas、Julian Michael、Luheng He 和 Luke Zettlemoyer。2018年。《大规模 QA-SRL 解析》。*arXiv* 预印本 *arXiv:1805.05377*。

Hachey、Ben、Will Radford、Joel Nothman、Matthew Honnibal 和 James R Curran。2013年。《使用维基百科评估实体链接》。人工智能 194：130–150。

Halterman，Andrew。2019年*a*。文本中的政治事件地理定位。载于 计算语言学协会北美分会第*17*届年会（*NAACL*）第三届自然语言处理与计算社会科学研讨会论文集。第29–39页。

Halterman，Andrew。2019年*b*。《关联政治文本中的事件与地点》。 计算语言学协会北美分会（*NAACL*）第三届自然语言处理与计算社会科学研讨会论文集。

Halterman, Andrew. 2022年。“用于有监督文本分析的合成生成文本。” *Pol- Meth* 会议论文。

Halterman, Andrew。2023。《Mordecai3：神经地理解析器与事件地理定位器》。工作论文 。

Halterman、Andrew、Benjamin E Bagozzi、Andreas Beger、Philip A Schrodt 和 Grace I Scarborough。2023。“PLOVER 和 POLECAT：一种新的政治事件本体和数据集。”国际研究协会会议论文。

Halterman、Andrew、Jill Irvine、Christan Grant、KT Jabr 和 Yan Liang。2018年。为阿拉伯语文本创建自动化事件数据系统。载于 *2018*年旧金山 *ISA* 年会。

Halterman、Andrew、Katherine A Keith、SheikhMuhammad Sarwar 和 Brendan O'Connor。2021年。"事件问答的语料库级评估：涵盖2002年古吉拉特暴力事件的IndiaPoliceEvents语料库。"计算语言学协会研究成果。

Hoffart, Johannes, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater 和 Gerhard Weikum。2011。《文本中命名实体的稳健消歧》。载于*2011*年自然语言处理实证方法会议论文集，第782–792页。

Honnibal, Matthew 和 Ines Montani。2017。"spaCy 2：利用 Bloom嵌入、卷积神经网络和增量解析实现自然语言理解。"即将发表。

Kang, Jun Seok, Robert L Logan IV, Zewei Chu, Yang Chen, Dheeru Dua, Kevin Gimpel, Sameer Singh 和 Niranjan Balasubramanian。2019年。"PoMo：在语境中生成实体专属后置修饰语。"*arXiv* 预印本 *arXiv:1904.03111*。

King, Gary 和 Will Lowe。2003。"一种用于国际冲突数据的自动化信息抽取工具，其性能与人工编码员相当：一种稀有事件评估设计。"《国际组织》57(03):617–642。

Kuhn, Harold W. 1955. "指派问题的匈牙利算法。"《海军研究后勤季刊》2(1-2):83–97.

Liang, Yan, Khaled Jabr, Christan Grant, Jill Irvine 和 Andrew Halterman. 2018年. 跨语言政治事件编码的新技术。载于 *2018*年 *IEEE* 信息重用与集成国际会议（*IRI*）。IEEE，第 88–93 页。

Miller, Blake, Fridolin Linder 和 Walter R Mebane. 2020年. "文本标注的主动学习方法：主动学习方法性能的回顾与评估。"《政治分析》第 1–20 页。

Norris、Clayton、Philip Schrodt 和 John Beieler。2017。"PETRARCH2：另一个事件编码程序。"《开放源代码软件期刊》2(9)。

Northcutt、Curtis、Lu Jiang 和 Isaac Chuang。2021年。“置信学习：估计数据集标签中的不确定性。”人工智能研究期刊70:1373–1411。

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei 和 Ilya Sutskever。2019年。“语言模型是无监督多任务学习器。”第 9 页。

Radford, Benjamin J. 2019年。“用于政治事件编码的自动词典生成。”政治科学研究与方法第 1–15 页。

Rajpurkar, Pranav, Robin Jia 和 Percy Liang。2018年。“了解自己不知道什么：SQuAD 中无法回答的问题。” *arXiv* 预印本 *arXiv:1806.03822*。

Reimers, Nils 和 Iryna Gurevych。2019年。Sentence-BERT：使用孪生 BERT 网络的句子嵌入。载于*2019*年自然语言处理实证方法会议论文集。计算语言学协会。

Sanh, Victor, Lysandre Debut, Julien Chaumond 和 Thomas Wolf。2019年$_a$。“Distil-BERT：BERT 的蒸馏版本，更小、更快、更便宜、更轻量。” *arXiv* 预印本 *arXiv:1910.01108* 。

Sanh、Victor、Lysandre Debut、Julien Chaumond 和 Thomas Wolf。2019年$_b$。“DistilBERT，BERT 的蒸馏版本：更小、更快、更便宜、更轻量。”。**URL：** *https://arxiv.org/abs/1910.01108*

Schrodt, Philip A. 2009年。“TABARI：通过增强替换指令进行文本分析。” 政治科学系，堪萨斯大学，布莱克楼，版本 *0.7。3B3*第1–137页。

Solaimani、Mohiuddin、Sayeed Salam、Latifur Khan、Patrick T Brandt 和 Vito D’Orazio。2017年。RePAIR：从新闻网站实时推荐政治行为体。载于*2017 IEEE* 国际大数据会议（大数据）。IEEE，第1333–1340页。

Toft、Monica Duffy 和 Yuri M Zhukov。2015年。“伊斯兰主义者与民族主义者：俄罗斯北高加索地区的叛乱动机与反叛乱。”《美国政治学评论》109(02):222–238。

Tunstall, Lewis, NilsReimers、Unso Eun Seo Jo、Luke Bates、Daniel Korat、Moshe Wasserblat 和 Oren Pereg。2022年。“无需提示词的高效少样本学习。”*arXiv* 预印本 *arXiv:2209.11055* 。

Vaswani、Ashish、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。2017。注意力机制就是你所需要的一切。载于《神经信息处理系统进展》。第5998–6008页。

Wick, Marc a nd C Boutreux。2011。“GeoNames。”*GeoNames*地理数据库 。

A 用于 event、mode 和 context 分类器的准确率评估

mode	训练样本数	测试样本数	准确率	精确率	召回率	F1
指控-声称	41	11	0.9091	0.9167	1.0000	0.9565
指控-反对	35	9	0.8889	0.9000	1.0000	0.9474
指控-调查	36	9	0.8889	0.9000	1.0000	0.9474
袭击-绑架	15	4	0.7500	1.0000	0.8000	0.8889
袭击-暗杀	8	2	0.5000	0.6667	1.0000	0.8000
突击-殴打	80	20	0.9500	1.0000	0.9167	0.9565
突击-清剿	24	7	0.8571	0.8750	1.0000	0.9333
突击-人群控制	12	3	0.6667	0.7500	1.0000	0.8571
突击-摧毁	15	4	0.7500	0.8000	1.0000	0.8889
突击-处决	17	5	0.8000	0.8333	1.0000	0.9091
突击-爆炸物	43	11	0.8182	0.8182	1.0000	0.9000
突击-火器	22	6	0.8333	0.8571	1.0000	0.9231
突击-重型武器	8	3	0.6667	0.7500	1.0000	0.8571

（续）

mode	训练样本数	测试样本数	准确率	精确率	召回率	F1
原始攻击	118	30	0.9667	0.9444	1.0000	0.9714
性攻击	8	3	0.6667	0.7500	1.0000	0.8571
自杀式袭击	23	6	0.8333	1.0000	0.8000	0.8889
酷刑	13	4	0.7500	0.7500	1.0000	0.8571
强制逮捕	364	91	0.9121	0.9123	0.9455	0.9286
强制禁止	72	19	0.9474	1.0000	0.9091	0.9524
COERCE-审查	63	16	0.9375	0.9000	1.0000	0.9474
COERCE-宵禁	65	17	0.9412	0.9000	1.0000	0.9474
COERCE-驱逐出境	101	26	0.9615	0.9231	1.0000	0.9600
COERCE-戒严	81	21	0.8571	0.8235	1.0000	0.9032
COERCE-虚假信息	72	19	0.9474	1.0000	0.9286	0.9630
COERCE-限制	96	25	0.8800	0.8500	1.0000	0.9189
COERCE-没收	99	25	0.9200	0.8667	1.0000	0.9286
COERCE-扣留	64	16	0.9375	0.9167	1.0000	0.9565
CONSULT-多边	53	14	0.9286	1.0000	0.9167	0.9565

续表

mode	训练样本数	测试样本数	准确率	精确率	召回率	F1
咨询电话	13	4	0.7500	0.8000	1.0000	0.8889
咨询第三方	29	8	0.8750	0.8750	1.0000	0.9333
咨询访问	112	29	0.8966	0.9231	0.9600	0.9412
动员民兵	18	5	0.8000	0.8333	1.0000	0.9091
动员警察	78	20	0.8500	0.9286	0.8667	0.8966
动员部队	121	31	0.8387	0.8000	0.9412	0.8649
动员武器	45	12	0.9167	0.8889	1.0000	0.9412
抗议-抵制	68	18	0.8889	0.8571	1.0000	0.9231
抗议-示威	464	117	0.8889	0.8866	0.9773	0.9297
抗议-绝食	109	28	0.9643	0.9375	1.0000	0.9677
抗议-阻挠	170	43	0.9535	0.9583	0.9583	0.9583
抗议-暴乱	275	69	0.8841	0.8367	1.0000	0.9111
抗议-罢工	147	37	0.9459	1.0000	0.9200	0.9583
拒绝协助	69	18	0.8889	1.0000	0.8000	0.8889

续

mode	训练样本数	测试样本数	准确率	精确率	召回率	F1
拒绝变更	92	23	0.6522	0.6500	0.9286	0.7647
拒绝会面	67	17	0.8824	1.0000	0.8182	0.9000
拒绝让行	64	17	0.7059	0.7000	0.7778	0.7368
请求-协助	248	62	0.8065	0.8095	1.0000	0.8947
请求变更	89	23	0.6522	0.7333	0.7333	0.7333
请求会面	73	19	0.8421	0.7273	1.0000	0.8421
请求-让步	68	18	0.8889	0.8750	0.8750	0.8750
撤退-准入	69	18	0.8333	0.8125	1.0000	0.8966
撤退-停火	82	21	0.9524	1.0000	0.9333	0.9655
撤退-解除武装	80	20	0.9000	0.8750	1.0000	0.9333
撤退-释放	227	57	0.8947	0.8718	0.9714	0.9189
撤退-辞职	93	24	0.9583	0.9286	1.0000	0.9630
撤退-返回	20	5	0.8000	0.8333	1.0000	0.9091
撤退-撤离	104	27	0.9259	0.9000	1.0000	0.9474

续

mode	训练样本数	测试样本数	准确率	精确率	召回率	F1
制裁-定罪	161	41	0.9024	0.9474	0.8571	0.9000
制裁-中止	226	57	0.8246	0.8889	0.7742	0.8276
制裁-驱逐	187	47	0.9149	0.9231	0.9231	0.9231
制裁-撤回	152	38	0.8421	0.8400	0.9130	0.8750
威胁-逮捕	71	18	0.8333	0.8000	0.8889	0.8421
威胁-禁止	77	20	0.8500	0.9167	0.8462	0.8800
威胁驱逐	64	17	0.9412	1.0000	0.9000	0.9474
威胁关系	72	19	0.8947	0.8824	1.0000	0.9375
威胁限制	117	30	0.7333	0.7692	0.9091	0.8333
威胁领土	71	18	0.8889	0.9000	0.9000	0.9000
威胁暴力	70 91	23	0.6957	0.8125	0.7647	0.7879

表 A3：模式分类器的准确率数据。请注意，验证集取自同一组标注文章，规模相对较小，因此较高的准确率可能无法推广到真实数据。

AGREE				
Mk7	训练集大小:	388	评估集大小:	97
Mk9	训练集大小:	239	评估集大小:	60
	Acc	精确率	召回率	F1
Mk7 均值	0.7126	0.6408	0.7744	0.6950
Mk7 标准差	0.0320	0.0574	0.0797	0.0121
Mk9 均值	0.7167	0.7373	0.8816	0.7983
Mk9 标准差	0.0408	0.0548	0.0721	0.0136
CONSULT				
Mk7	训练集大小:	404	评估集大小:	101
Mk9	训练集大小:	337	评估集大小:	85
	Acc	精确率	召回率	F1
Mk7 均值	0.8998	0.8920	0.9100	0.9004
Mk7 标准差	0.0273	0.0435	0.0173	0.0253
Mk9 均值	0.8603	0.8307	0.9884	0.9015
Mk9 标准差	0.0592	0.0567	0.0129	0.0346
拒绝				
Mk7 均值	训练集大小:	373	评估集大小:	94
Mk9	训练集大小:	247	评估集大小:	62
	Acc	精确率	召回率	F1
Mk7 均值	0.5425	0.5092	0.6477	0.5690
Mk7 标准差	0.0213	0.0175	0.0612	0.0274
Mk9 均值	0.4597	0.5245	0.5571	0.5305
Mk9 标准差	0.0228	0.0244	0.1262	0.0604
THREATEN				
Mk7	训练集大小:	71	评估集大小:	18
Mk9	训练集大小:	40	评估集大小:	11
	Acc	精确率	召回率	F1
Mk7 均值	0.5417	0.5863	0.6000	0.5910
Mk7 标准差	0.0666	0.0551	0.0866	0.0625
Mk9 均值	0.4431	0.5456	0.7678	0.6345
Mk9 标准差	0.0301	0.0192	0.0994	0.0361
袭击				
Mk7	训练集大小:	390	评估集大小:	98
Mk9	训练集大小:	212	评估集大小:	53
	Acc	精确率	召回率	F1
Mk7 均值	0.7398	0.7081	0.7772	0.7383
Mk7 标准差	0.0542	0.0677	0.0541	0.0447
Mk9 均值	0.6250	0.6785	0.2881	0.3494
Mk9 标准差	0.0476	0.1330	0.2227	0.2177

表 A1： 文档级事件类别分类器在筛除无关训练文本并对训练案例进行人工审查之前（*Mk7*）和之后（*Mk9*）的准确率对比数据。

语境	训练样本数	测试样本数	准确率	精确率	召回率	F1
庇护	269	68	0.868	0.943	0.825	0.880
腐败	301	76	0.802	0.765	0.929	0.839
犯罪	258	65	0.831	0.774	0.857	0.814
网络	198	50	0.840	0.833	0.893	0.862
外交	190	48	0.729	0.833	0.758	0.794
灾害	196	50	0.840	0.842	0.762	0.800
经济	412	103	0.796	0.870	0.727	0.792
选举	306	77	0.727	0.745	0.796	0.769
环境	294	74	0.865	0.897	0.854	0.875
性别	204	52	0.769	0.926	0.714	0.807
健康	92	24	0.792	0.706	1.000	0.828
不平等	301	76	0.790	0.784	0.784	0.784
情报	233	59	0.780	0.788	0.813	0.800
法律	297	75	0.733	0.725	0.763	0.744
立法	170	43	0.767	0.818	0.750	0.783
lgbt	157	40	0.925	0.875	1.000	0.933
移民	170	43	0.767	0.708	0.850	0.773
军事	312	78	0.872	0.881	0.881	0.881
错误信息	86	22	0.727	0.684	1.000	0.813
维和	328	82	0.951	0.917	1.000	0.957
赔偿	113	29	0.759	0.800	0.842	0.820
镇压	312	79	0.810	0.857	0.800	0.828
技术	296	75	0.707	0.744	0.744	0.744
领土	79	20	0.800	0.733	1.000	0.846
恐怖主义	334	84	0.655	0.655	0.783	0.713

表 A2： 文档级语境分类器的准确率数值。请注意，验证集取自同一组标注文章，且规模相对较小，这意味着较高的准确率可能无法推广到真实数据。

B 使用主动学习和合成文本收集训练数据

迄今为止，每一种自动事件编码系统都依赖于人工编制的词典；这些词典用于处理文本，文本要么由程序内部解析（KEDS、TABARI、ACCENT），要么由开源解析器解析（PETRARCH 系列编码器）。编制这些词典需要投入大量专业人力，而且词典针对特定语言。

相比之下，NGEC 的各个子系统都会从一组训练示例中“学习”。目前这组示例约有数百个正例，不过增加示例数量会提升系统性能。具备语言知识和领域知识的人员就能轻松制作这些示例，无须掌握编码程序所用的词典规范语言（通常也无须具备计算语言学知识）。基于示例的编码还能确保类别定义随时间保持一致；而在一系列各自使用不同词典、用途单一的程序之间维持这种一致性，事实证明非常困难。过去，生成训练案例的过程相当耗时且繁琐；如今，Prodigy 标注系统等较新的方法可以大幅加快这一过程[26]，尽管繁琐的工作并未完全消失。

与编码器专用词典不同，即使底层的机器学习/自然语言处理技术发生变化，标注案例仍可重复使用；随着新技术在商业云平台上或以开源代码的形式出现，也可以几乎不费力地用它们来测试这些新技术。此外，大量且稳健的标注案例（除本体发生变化的情况外，可通过累积方式获得）使得分样本成为可能：用其中一部分案例训练模型，再用其余案例测试模型。基于词典的系统很少采用这种方法，即便采用，也只使用了少量样本。

最后，可以轻松评估标注案例本身的质量，以识别出以下几类案例群组：标注员表现不佳的案例群组，或其性质……

[26]https://prodi.gy/

数据发生了变化，通常是由于引入了新词汇（例如，“种族清洗”一词在 20 世纪 90 年代之前尚未被使用），或者我们发现对本体的解释存在不一致。这些问题十分普遍，事实上，它们是当代所有机器学习项目的核心问题。因此，人们已经开发出种类繁多的方法来解决这些问题，而这些方法通常以开源软件的形式实现。

Polecat/Plover 与所有基于Cameo的系统之间的一个关键区别在于，我们使用的是报道[27] ，而不是先前基于解析器词典的系统所编码的单个句子。这种方法的明显优势是，可以获取报道的大部分相关**语境**，不过，即便是完整的报道，尤其是来自地方而非国际来源的报道，往往也会假定读者掌握不少背景知识，例如邻国主要城市的名称以及重要政治人物的身份。因此，大多数报道都包含多个事件。复合短语也可能让单个句子包含多个事件，但这种情况相对少见。

由于涉及的文本更多，试图开发训练案例的人员需要承担更高的认知负荷，而且通常必须实际阅读和处理文本，例如解决共指问题。使用 Prodigy 等现代编码软件时，经验丰富的编码人员（以及运用主动学习的编码软件本身）可以非常快速地处理单个句子；而处理完整报道则要困难得多，也更令人疲惫。

尽管如此，为标注训练案例的人员提供的总体训练，远少于为基于解析器的系统开发词典的人员所需的训练；而在拥有成熟编码本体的系统中——我们在开发 Plover 的过程中逐步构建系统，因此当时并不具备这样的本体——所需的训练应该更少，尤其是那些对具备大学程度政治学知识的人员而言显得陌生的训练。

[27]更准确地说，是大段故事文本：第2.2.1节将对此进行讨论。

B.1 收集语境标注

如上所述，**语境** 对应于特定政治事件的“原因”部分。Plover 的**语境** 标签旨在让最终用户能够根据特定的总体主题搜索事件和／或筛选事件，例如环境、人权或犯罪相关事件：表 A2 展示了我们目前使用的编码集合。

在我们的标注工作中，大约八名人类专家标注员会判断一篇新闻报道的全文中是否包含某个给定的**语境**，以便日后用作训练数据。[28] 如果新闻报道中讨论了某个给定的**语境**，则将其视为“正例”；如果没有对某个给定的**语境**进行充分讨论，则将其视为“负例”。无论是人工标注任务还是后续的机器学习分类任务，一篇新闻报道都可以包含多个**语境**。在某些情况下，允许“忽略”某篇新闻报道，而不将其标注为某个**语境** 的正例或负例。[29] 新闻报道样本包括从 20 世纪 90 年代至今的文章，这些文章此前曾用于 ICEWS 事件抽取。

通过基于网络的 Prodigy 界面，以半随机的方式将单篇新闻报道分配给各位标注员。该界面一次向标注员展示一篇新闻报道，并依据一组特定于**语境**的关键词进行筛选。每个**语境** 类别的关键词均由同一位团队成员统一指定，依据的是 Halterman 等人（2023）补充附录中所列的段落长度**语境** 定义。随后，编码界面采用“主动学习”方法动态更新，为人工标注员提供相关性逐渐提高的新闻报道。具体而-

[28]出于预算考虑，每篇文章仅由一名标注员进行标注，这并非自然语言处理研究人员制作评估数据集时采用的最佳实践。不过，我们经常以小组形式审查标注人员的标签，并对其工作提供反馈。此外，这一流程也体现了我们的规程会随时间变化，因此早期训练案例与我们目前的做法并不完全兼容。最初，我们要求编码员标注报道**任意位置**出现的语境；但随着系统开始部署，我们意识到用户通常只能看到用于编码事件类别的 512 词元文本，如果所编码的**语境** 依据的是只能从数据库深处访问的文本，就可能造成困惑。因此，我们改为只根据 512 词元文本进行编码。[29]这种做法很少使用，主要适用于编码时含义过于模糊的报道，或与政治事件数据抽取完全无关的报道（例如各公司的股东／财务报告）。

言，标注员将收到的报道标记为某一**语境**的正例或负例后，系统会识别并向该标注员提供更多新闻报道；这些报道不仅依据最初提供的关键词筛选，还越来越多地依靠动态更新的监督机器学习模型来确定。系统会向标注员展示模型预测最不确定的文档，确保每份文档的标注都能为模型训练提供尽可能多的信息。通过这种方式，对于每个**语境**，专家标注员都能够高效标注最相关的新闻报道，以用于 Polecat 后续的监督机器学习（数据编码）阶段。

在针对**语境**定制的 Prodigy 界面上进行实际标注时，每位人工标注员都必须先完整阅读提供的新闻报道，再作出标注判断。随后，标注员需要指出该报道是否在文本某处讨论了指定的**语境**类别。此外，还采用了以下标准。首先，新闻报道若要获得肯定的**语境**标注，其主题不必与报道中感知到的事件明确相关。这样做是为了确保**语境**标注与事件数据标注和编码流程中的其他组成部分充分分离。同时，只有当**语境**构成报道的总体主题时，该报道才能获得肯定标签。其次，新闻报道可以涉及多个**语境**；在适用的情况下，某一**语境**的存在不应妨碍为该报道标注另一**语境**。最后，标注**语境**时，应侧重于对总体主题的讨论，而非提及具体的行为体和行动。这样可以确保后续对实体、事件类型和模式的标注与**语境**部分充分分离。

我们要求标注员为 Plover 的 37 个常见**语境**类别各标注至少 250 个正例（即新闻报道）[30] ；对于 13 个较少见的**语境**类别，则标注总计 1,000 个正例或负例**语境**案例。[31]。

[30]这通常确保每个**语境**类别的负标签数量显著多于这些正例。[31]这确保了这些较罕见的情况有 75 至 200 个正标签，另有 800 多个负标签。

B.2 收集事件类别的标注

事件类型标注在 Prodigy 上采用了与**语境**标注类似的标注过程，并使用了与上述相同的标注员。不过，与**语境**标注不同，事件类别标注需要标注文本文档中的特定子集（即文本片段），而不是仅对整篇新闻报道的全文进行标注。事件类别标注有两个目标。首先，对于给定类别，我们标注特定新闻报道是否记录了某一类型的事件。尤为重要的是，与以往的自动事件编码系统不同，标注员以文档而非句子为单位开展工作，以处理同一事件常常跨多个句子进行报道的情况。出现某一类型事件的实例（即报道）被视为该类别的“正例”，而未出现某一事件类别的实例（即报道）则被视为“负例”。与**语境**标注一样，在适当情况下也允许跳过新闻报道而不赋予标签。

其次，对于包含某一特定类型事件的报道，标注员会在新闻报道文本中标出描述该事件的具体段落（说明谁在何时何地对谁做了什么）。随后，我们利用这些标注，通过基于 Transformer 的神经网络，在未标注的新闻报道文本中检测具体的事件跨度，并确定 Polecat 完整新闻语料库中这些检测到的事件跨度所属的事件类别。[32]

在 Prodigy 事件类别界面中，标注人员一次标注一个事件类别，并力求在每个类别中至少标注 250 个正例。[33] 标注人员为全部 16 个事件类别都达到了这一正标签阈值，因此通常确保每个事件类别至少有 1,000 个可比较的负标签。即使事件的具体施事者和／或受事者不明确，标注人员也会标记存在事件的实例。

[32]我们目前尚未使用文本片段级信息，但计划在未来利用这些信息识别同一事件类型的多个实例，并更好地指导问答/属性识别步骤。[33]THREATEN 除外，见表 A1 及 C.2 节中的讨论。

与**语境** 标注类似，这确保类别（及其整体上的存在或缺失）的标注与更广泛流程中的属性标注环节相互独立。在这一阶段，标注人员必须确保，他们对新闻报道中事件的（未）识别，不受自身识别事件参与者和受事者能力（不足）的影响。

B.3 收集模式标注

模式标注采用了与前文所述事件类型及**语境**标注活动类似的、基于 Prodigy 的方法。在此过程中，同一组专家标注员对新闻报道进行标注，判断其中是否存在各个模式。这一标注过程不同于前文所述的事件类别及**语境**标注过程：模式标注预先要求，待标注的单篇新闻报道已在上一阶段被确定包含某一特定类别。为确保此类案例的样本量充足，研究人员还生成了合成报道，将其纳入各相关事件类型的模式标注样本（详情见 Halterman（2022））。对于这组由真实报道和合成报道组成的材料，模式标注员逐篇标注，判断其中是否存在与该类别相关的各个模式，并将其标记为“正例”或“负例”。

每位标注员都被分配了 Plover 中 11 个已定义模式的事件类别之一。标注员在 Plover 手册中查看了所分配类别的相关模式，随后收到了一系列与该类别相关的新闻报道。对于每篇提供的新闻报道，标注员都需要评估一种特定模式；该模式通过提示显示在 Prodigy 模式标注界面的顶部，供标注员参考。根据所提供的信息，标注员需要标注所提供的新闻报道中第一个相关事件是否包含某一特定模式的证据。在此过程中，我们发现，有些模式数量特别多的事件类别最终需要更多（最多两到三名）

标注员。每种事件模式至少由标注员标注 50 个正例（即新闻报道），负标签案例则没有规定阈值。最终，所有相关模式都达到了这一目标。同一类别中某些模式的出现频率往往与其他模式不同，因此每种事件类型通常至少有一些模式获得了 200 个以上的正标签。

B.4 收集属性标注

属性标注涉及事件的施事者、接收者、地点和时间，采用与上述相同的 Prodigy 界面和编码人员完成。在这种情况下，这些标注以先前识别出的事件（以及相应的真实和合成新闻报道）为基础，要求标注员突出显示所选事件相关的施事者、接收者、地点和发生时间；并标记同一施事者和/或接收者在选定的标注报道中以不同方式多次出现的情况。该流程适用于全部 16 个 Plover事件类别，并确保每个类别至少有 100 篇实体标注报道。在处理流程的开发过程中，标注工作持续进行，以确保有足够的数据用于训练和评估问答模型。

B.5 使用自动化方法优化训练案例 ds

与基于词典的方法不同，只需将模型（理想情况下是采用多种方法的模型集成）应用于训练数据，并找出那些始终被标记为假阳性（即被编码为负例、却被模型判为正例的案例）或假阴性的案例，再将其剔除，就能轻松评估训练案例的内部一致性。经过人工审查，还能进一步发现类别定义中可能存在的歧义；这类歧义很少能从词典中明显看出。人工审查也有助于发现个别编码人员的性能问题（这是资源充足的项目中的标准流程，但在需要寻求志愿者协助时，有时难以实施）。

用于此项工作的软件，无论是定制软件还是开源软件（例如 CleanLab（Northcutt、Jiang 和 Chuang，2021年）），都很容易使用。我们已经让案例经过了一到两轮这样的处理，并将在系统不断完善的过程中继续进行。值得指出的是，可以 将这一流程实现为完全自动化的迭代过程，但这会造成锁定效应：系统会越来越聚焦于初始阶段中占据主导地位的真正例和真负例，而这些案例未必对应着对编码框架的最佳解读。事实上，最需要投入额外人工标注工作的，可能正是那些非典型的“边界案例”。例如，在 PROTEST类别中区分街头抗议与外交抗议、立法抗议。理想情况下，应按照第5节所述，基于一个规模大且可靠的验证集进行这一改进；如果有这样的验证集，系统就可以完全自动化，但目前我们还没有。

B.6 为罕见事件类型开发案例：合成案例

我们在某些类别（无论是事件、模式还是语境）中很快遇到了一个问题，即罕见事件类型：在我们的语料库中，潜在训练案例的产出率低于 5%，有时甚至远低于 1%。这种情况存在两个方面的问题：一方面，寻找足够数量的训练案例需要投入大量人力；另一方面，这也容易使人工标注员感到极度疲惫和沮丧，因为他们需要阅读数十个不相关的案例，才能偶尔找到一个正例。

我们尝试用两种方法解决这个问题，但都不尽如人意。首先，由于这项工作是美国政府项目的一部分，我们可以访问那些经过编码、对应正例的授权新闻报道文本（受许可限制，这些文本无法提供在公开的 ICEWS数据集中）。我们有数百万条此类文本，涵盖了 20 年的时间。此外，对于我们正在编码的许多事件和模式类别，详细的 Cameo 代码与 Plover 事件-模式-**语境**三元组之间存在直接的对应关系；这张翻译表已在本项目中得到广泛使用（并且

基于美国国家科学基金会资助的早期 Plover 工作），可在我们的 GitHub 网站上获取。

遗憾的是，这最终只被证明是部分解决方案。首先，编码到 Cameo 中的 iData 文本当然都只是单句，而训练案例需要完整的故事文本。此外，虽然我们可以访问 iData，但无法访问用于生成这些数据的完整原始 Factiva 及其他信息源；即使在 iData 中，我们也只有产生了正向分类结果的文本。其次，许多新的 Plover 模式和 **语境**都没有明确对应的 Cameo 项目——几乎所有语境都是如此——因此我们没有可用的材料。最终，通过基于支持向量机的分类器反复从 iData 中提取文本，有助于生成产出率显著更高的标注集，从而减轻标注人员的认知负担，但仅凭这种方法不足以生成有用的训练案例。

我们尝试的第二种方法是使用 GPT-2（Radford，2019年）生成合成案例，尤其针对那些没有 Cameo/iData 对应项的**语境**。同样，这种方法生成了一些有用的案例，但由于该技术的两个局限，只能算部分解决方案。首先，我们通常只能生成可信的、长度约为两句话的合成文本，远低于实际故事文本的 512 个词元长度。其次，文本生成器往往会集中生成少数几种有代表性的动词短语——例如，不知为何，它喜欢生成抗议者用棒球棒袭击警察的 ASSAULT 案例——同时只变换行为体；生成若干个案例后，这些文本便不再为训练集提供新信息，实际上还会使其产生偏差。不过，优化 GPT-2 的生成超参数以提高生成文本的质量，或使用 GPT-3（及其后续版本），都是未来标注工作的可行方向（Halterman，2022年）。

尽管存在这些限制，我们还是纳入了一些合成案例：这些案例的优势在于易于审阅，且产出率相当高（25%+）；更重要的是，完全不存在知识产权限制。借助当前（以及未来）的大语言模型技术——ChatGPT 及其迅速涌现的衍生模型——我们或许能够获得好得多的结果，而且尤其能够轻松获得与 512词元框架兼容的故事文本，

如果资源允许，我们无疑会在未来深入探讨这一点。

B.7 关于标注数据的注意事项

为确保透明度，我们谨指出，现有训练案例至少在三个方面仍有很大不足：

首先，由于预算有限，且相较于任务规模，最初可用的时间非常短，同时也无法获得大学研究环境中可用的低成本熟练劳动力，因此我们没有采用标准的大规模编码项目规程：先组建一组编码员并对其进行充分训练；在生成实际用于建模的数据之前，先进行多轮初始编码实验以完善编码手册（我们做了一部分）；通过这些初始步骤确定哪些编码员具备相当的熟练程度（值得一提的是，能力不足的人通常会自行退出）；然后在数月时间内，在密切监督下完成最终编码；最后，在这些准备工作全部完成后，如有需要，再对发现存在歧义的案例重新编码。KEDS/TABARI 项目采用的就是这种方法论，由学生编码员参与，历时约十五年；BBN/ACCENT 项目也采用了这种方法论，由专业编码员参与，历时约五年。而我们只有约四到六个月的兼职劳动力。

其次，尽管在项目开始时，Plover事件类别相对稳定，但模式和**语境** 并不稳定，尤其是**语境**。此外，尽管模式大多对应于Cameo子类别，但Cameo中几乎没有与**语境**相对应的内容，因此，无论是Cameo数据还是实现Cameo的PETRARCH 词典，都无法为我们提供太多帮助。

第三，这一问题之所以更加复杂，还在于，尽管某些模式和**语境**受到我们的赞助方关注，但由于我们所使用的数据集此前已通过 Cameo/ACCENT编码进行过滤，因此这些模式和语境在其中极为罕见（也就是说，我们没有未经筛选的新闻源）。

合成案例在一定程度上解决了这一问题。[34] 未来，我们几乎肯定会使用 ChatGPT 及其后续版本；它们远远优于我们一年前所能使用的方法——但也只能部分解决问题。

这些问题都没有妨碍我们实现一个经实践检验、值得信赖的系统。不过，如果重新来过，无论是聘请专业编码员，还是在大学环境中由训练有素的本科生和研究生编码员参与，并留出更长的时间，采用标准的编码员培训方案肯定是可行的。积极的一面是，基于示例的系统中错误的编码结果，比基于词典的系统中有问题的短语容易发现得多；随着时间和资源逐步到位，我们预计会着手处理这一问题。

C 预过滤与后过滤

- 在文本中，我们会删除一系列日期行、[35] 归属信息/作者信息、新闻稿组成部分（例如“栏目：”“照片：”）、编辑处理备注（例如“禁发至……” “拼写已更正……”）以及其他重复出现的套话

- 篇幅明显超过512词的报道：这类报道很少涉及当代事件，更可能是背景或历史讨论，或演讲稿的文字记录

- 非常短的故事文本：这类情况通常发生在某篇报道的解析过程中出现错误，导致最后只剩下标题，或标题的片段

[34]在2000—2003年堪萨斯大学开发Cameo期间，这里出现了一个有趣的相似之处：我们决定为每个编码至少收录三句来自路透社真实新闻报道并带有标注的例句；后来，由于在约20年的中东新闻报道中找不到足够明确无歧义的示例，我们最终删除了几个编码。遗憾的是，不同来源的日期行没有统一格式，这给试图识别地点的子系统带来了特别大的问题，因为日期行通常标注的是一座大城市，而该城市可能距离实际事件发生地数百甚至数千英里。非洲研究者中流传着一个或许并不真实、却广为人知的故事：20世纪60年代，《泰晤士报》的一位编辑给该报开普敦分社发电报，请他们报道西非的某场动乱，得到的回复却是：“你们自己报道吧：你们离那里更近。”至于ICEWS系统，它在这项任务上的表现确实很糟糕，因为地点解析是后来才补上的。

• 主要由数字组成的故事文本，通常是市场报告或体育比分。你可能会认为 Factiva 可以将它们过滤掉，但不知为何过滤效果并不完善。

• 涉及不含政治内容的犯罪的故事文本，通常是谋杀，或游客遭遇的不愉快事件。我们原以为 Factiva 已经剔除了这些内容；事实证明我们错了，而这些内容最初严重干扰了我们的胁迫和袭击行为编码。

对于那些可能涉及政治、也可能不涉及政治的犯罪，例如腐败，以及可能出于政治动机、也可能并非出于政治动机的袭击（或杀戮），该如何处理？这里显然存在灰色地带，而虚假指控又使情况更加复杂；其中，针对各种形式性不当行为的指控颇为常见，而不同文化对不可接受行为的具体模式看法各异。有时，抢劫就只是抢劫，因此最终总会有一些无法判定的案例。

• 涉及事故和自然灾害的故事文本，这些都不在我们的编码范围内。就假阳性编码而言，这些内容不像犯罪故事文本那样棘手，但仍会耗费资源。此外，这些内容也存在灰色地带，因为重大灾害往往会引发国际援助响应。

• 我们排除了完全属于美国国内事件的事件。

• 涉及否定的句子，例如政客表示自己不会做某事。这类情况颇为特别：由于人类倾向于高估新颖性，这种表达似乎很常见，也很重要。事实上，就事件而言，这类情况相当罕见——人们会使用“拒绝”或“指控”等否定动词，但很少使用简单否定；而且最常见的情形是，有人在谈论对手据称持有的立场（换言之，是把否定的新颖性当作一种修辞手法）。识别这类句子原本可能是个重大难题——事实上，否定也是计算语言学领域的一项重大技术挑战——但好在 spaCy 自然语言处理

系统已为我们处理了这项任务，而且在处理流程中只需几行代码。

- “复合新闻”是由标题、重大新闻、即将发生的事件等组成的列表，在单句分类器中相当有用，因为单篇报道会提供多个独立事件的信息。遗憾的是，在全文编码中，这类文本会被编码为大量可能互不相关的事件，因此问题不少；作为训练案例，它们的问题更为严重，因为提供的信号极其分散，而这恰恰是训练案例最不应有的特点。[36] 路透社和英国广播公司等主要国际来源对此类文本采用的格式相当标准，因此很容易将其剔除；但各种国家级来源或其他来源的情况就没那么简单了。在人工标注过程中，标注人员疲劳时也很容易漏掉这类文本：报道前约三分之二看起来完全正常，但剩下的三分之一中可能夹杂着几个无关的标题或摘要，标注员很容易忽略这一点。

C.1 校准与分类后筛选

如上所述，与基于词典的模式匹配相比，机器学习分类方法的一项关键优势在于，我们得到的是实数形式的分类分数，而不是非真即假的二元判定。原则上，分类分数——这是一种任意指标，并非概率——与分类置信度成正比。因此，在确定一组共识模型后，我们将其应用于大量代表性样本，并确定正向分类分数的分布，[37]随后尝试将截断点设在不同水平——从一开始就很明显，这些截断点需要设得相对较高，例如处于第70至第95百分位范围——然后[迅速] 审查输出，以评估我们在多大程度上排除了那些在人看来似乎不属于此类别的案例。

[36]我们在这里没有妥善指导人工标注员不要纳入这些内容；在训练集的一些早期版本中，它们造成了相当大的问题。现在大多数这类内容都已移除，但由于非标准化问题，仍有一些未被剔除。[37]我们还尝试了其他几种指标，例如正分数与负分数之间的差值和比率，但这些指标似乎没有带来明显的额外优势

当然，我们原以为阈值设得越高，结果就会越好。遗憾的是，事实并非如此：当然，按定义，随着百分位数上限提高，得到的案例会减少，但我们认为属于此类别的案例比例几乎没有变化。这是因为机器只是在越来越严格地执行它从 [明显不完美的] 训练案例中推断出的规则，而并未真正理解我们对于哪些内容属于每个 Plover类别的直觉。

一方面，我们可以将此简单地解释为训练案例不足所导致的结果；但经过进一步思考，我们认为原因可能不止于此。首先，在许多（或者说大多数？）机器学习方法中，提高分类阈值确实会将输出收窄到某种意义上最接近训练案例质心的内容，也就是说，只保留最匹配的结果。但这里并没有发生这种情况，原因可能在于大语言模型的特性：任何大语言模型所包含的绝大部分知识，都来自未经修改的母模型；在我们的案例中，这个母模型是distillBERT，它关注的是一般语言特征。而我们数量相当有限的训练案例 —— 充其量大约 50 万个词，相比之下，用于训练 BERT 的词数达 34 亿——只能对这些特征加以微调。

但更普遍地说，这些模型非常擅长捕捉某些规律；尽管这些规律在庞大的大语言模型向量空间中确实存在，我们人类却无法察觉，尤其是那些迫切希望这项技术奏效、恨不得越快越好的人。

一个很好的例子是，我们曾试图改进用于性别语境的**分类器**。[38] 这本身就是一个相对棘手的案例：对人工标注员来说，**语境** 相当明确，但在现有新闻记录中，相关事件却 极其 罕见，尤其是在剔除性侵刑事案件之后。因此，系统大幅过度分类，而且特别倾向于将涉及亚洲大国的事件归类其中。

不过，众所周知，直到不久之前，ICEWS 对亚洲的事件采样严重过度，因为亚洲及太平洋地区是美国国防高级研究计划局最初资助的 ICEWS 项目的重点，这一点也反映在用于选择训练案例的语料库中。当一个分类器[38]实际上这是一个支持向量机，但基于大语言模型进行向量化可能存在的问题仍然适用。

在这些案例上训练的分类器随后应用于当前语料库时，由于新闻搜索词已经修改，以减少亚洲选择偏差，系统“看到”亚洲案例时，它看起来与性别案例相似；而由于训练案例数量较少，性别案例本身往往相当模糊，因此系统便认为**语境**一定与**性别**有关。我们没有时间或资源正式检验这一假设，但它至少看起来是合理的。

那么，尤其是在时间紧迫的情况下，我们该如何获取关键词呢？对于事件，我们先从 PETRARCH 词典中的主要动词列表入手；这些词典是开源的。随后，我们用 spaCy 的 word2vec 相似度度量，将这些动词与正例训练案例中频繁出现的动词进行比较（先照常剔除停用词等），之后又做了一些调整。**语境**（模式已经运行得足够好）的处理方法类似，不过我们的基准是 Plover 手册中的**语境**头衔和**语境** 描述中的代表性词语。[39]针对**语境**的最后一步是通过一个程序进行交互式筛选：我们几个人想出各种候选术语，再用代表性语料库测试这些术语是否实际出现——几秒钟内便能得到反馈——最终汇集成一份人工整理的列表。

这最后一步大幅且系统地减少了正向类别和**语境** 分配，[40] 产生了两种影响。对于主要事件，这改变了事件分布：原先的分布大致均匀（也就是说，各类别中的事件数量大致相等，这可能是我们在训练集中始终采用 1:1 正负比例所致），现在则更接近 ICEWS 中的事件分布；这是我们现有数据中最接近参照集的分布。对于**语境** ，这种情况似乎总体上也发生了：后过滤消除了大量我们认为相对罕见的**语境** 分配，但在**语境**中，

[39]好吧，事实上，我们直到流程结束时才想到使用手册，但本该一开始就这么做。我们也应该使用二元词组，而不只是单词——spaCy 的相似度函数完全可以处理二元词组——但我们没时间了：各位不妨另辟蹊径。[40]偶尔，如果所有类别分配都被过滤掉，整篇报道就会被删除；但更常见的情况是，一篇报道被分配了八个类别，最后只剩下两个类别。只要报道有一个有效类别，即使没有**语境**，也会保留下来：这并不少见，因为**语境**并不打算面面俱到

例如，我们知道**军事** 很常见。

C.2 对 AGREE、CONSULT、拒绝、THREATEN 和袭击的过滤与人工编辑效果比较评估

2022年晚些时候，我们对事件类别分类做了两项调整：一是采用了第2.2.2节讨论的若干过滤器；二是对 ICEWS 中编码频率最高的三个类别（更准确地说，是它们对应的 Plover 类别）AGREE、CONSULT 和 袭击，以及分样本准确率低于我们 0.80 目标的两个类别 REJECT 和 THREATEN，进行了全面的训练语料库人工审查。表 A1 展示了各类别分样本统计量的初步比较，其中“Mk7”指较早的流程，“Mk9”指较新的流程。请注意，这些是分样本结果，而不是针对独立验证集的测试（见第5节）；对于袭击类别，这一点尤其重要。每个类别的结果基于 8 个估计模型（因此有 8 个不同的分样本）：在 Google Colab GPU环境中运行大约需要半小时。

首先请注意，所有人工审查都使训练数据的总量减少了约 30%。还要注意，Mk7 和 Mk9 中 THREATEN 类别的训练案例都非常少——如下文所述，我们的 Factiva 样本来自冲突相对较少的时期，因此很难找到此类行为的示例——这或许可以解释其准确率较低的原因。REJECT 类别的准确率较低仍有些令人费解，因为样本量本身很大，但这些示例的组合显然仍难以让大语言模型准确描述。

总体而言，结果，尤其是 F1指标（精确率与召回率的平衡），总体上变化不大：AGREE 和 THREATEN 中上升了不少，REJECT 中略有下降，CONSULT 中则没有变化。目前看来，精确率和召回率的变化似乎没有一致的规律。Mk7 到 Mk9 之间，准确率（“acc”）通常有所下降；但这一点不太重要，因为与实际数据相比，此处的样本经过人为平衡（正例和负例各占 50%）。

那么，这究竟是好是坏？这项耗时的审查值得做吗？从某种意义上说，结果可能是“好”：过滤器剔除了一些我们不希望编码的类别，这些类别可能因具有鲜明特征而人为抬高了各项指标；去除这些类别后，系统的表现基本保持稳定。此外，在较小的训练集下，分样本的表现也保持稳定。

表现异常的是袭击，其 F1值下降了一半以上（从 0.73 降至 0.34），尽管准确率仅下降了约 15%。原因在于召回率大幅下降，约减少了三分之二（从 0.77 降至 0.28）。（提醒一下，“召回率”指正类测试案例中被正确识别出来的比例。）这一显著差异几乎可以肯定主要是由于移除了街头犯罪案例，这类案例数量很多（请注意，训练案例数量减少了近 50% ），而且通常相对容易编码，因此这些案例抬高了 Mk7 的指标。与此同时，我们显然没有足够的“常规”袭击案例，这可能是因为现有训练案例取自一个相对平静的时期——在叙利亚、伊拉克和阿富汗的大部分暴力冲突之后，以及乌克兰—俄罗斯冲突之前。如果进一步完善该系统，所有这些危机都能为袭击类别提供训练案例，这些案例很可能会提升系统在 2010—2022年期间的表现；这一时期包含了相当多持续时间较长的大规模冲突。

这或许也是讨论人工编码事件数据局限性的好时机。委婉地说，这类数据远非完美。例如，Douglass等（2022年）介绍了新的ICBe数据集，该数据集基于Breecher-Wilkenfeld国际危机行为数据。ICBe拥有极为理想的待编码文本——由专家以英文撰写、经过彻底编辑的回顾性摘要——但其平均编码者间一致性（诚然，本体非常复杂）仍勉强超过60%，专家的平均一致性则只有约55%；唯一超过80%的是对某事件是否存在的判断。[41] 这些数字也与大约十年前发表的一项研究结果相符，该研究考察了乌普萨拉冲突数据项目的一个数据集；UCDP同样使用专业编码员，并进行回顾性编码（滞后一年）。在我们对已标注数据进行的内部测试中[41]这篇论文在arXiv上的V1版本中列出了编码者间可靠性数据，但在V2版本中已不再包含这些数据。

这些案例中，我们的分样本准确率通常超过80%，而且经常远超90%。[42] 因此，对于大多数Plover事件类别，NGEC的表现远胜人工编码；此外，它还能近乎实时地维护全球数据集，而人工编码根本无法做到这一点。

D 术语表

	术语表和首字母缩略词
ACCENT	由 BBN 团队开发的 Cameo 编码系统，该团队采用了 关于洛克希德在 2012 年前后生产的 ICEWS 数据；它是 源自名为 SERIF 的通用 BBN 自然语言处理系统
AI	人工智能，机器学习的早期称谓
BBN	一家大型国防承包商，擅长信号处理， 涉猎领域众多，并曾短暂负责生产 ICEWS 数据
BERT	由谷歌开发并发布的一种超大型神经网络 Transformer 模型 由谷歌提供。
Cameo	最初在美国国家科学基金会资助下开发的事件数据本体。 21世纪初用于学术研究，随后被……采用。 美国国防高级研究计划局 ICEWS 项目及后续研究
	续……

[42]在论文的未来版本中，我们将提供所有类别的这些指标：由于缺乏预见，我们没有保留最近几次估算运行的这些数据。

	术语表和首字母缩略词
ChatGPT	由 OpenAI 开发的大语言模型，目前（2023年2月）正席卷 科技圈
美国国防高级研究计划局	美国国防部国防高级研究 项目机构
distilBERT	一个更小的，更高效的 BERT 大语言模型子集
Factiva	一家主要的数据聚合商，为以下内容提供了源故事文本 ICEWS/iData
GitHub	广泛使用的开源软件云端代码库
GPU	图形处理器，一种专用但广泛可用的芯片 可将神经网络处理速度提升 30 至 60 倍
GeoNames	一个开放访问的全球地理数据库，包含超过一千一百 万个地名
GPT-3	构成 ChatGPT 部分基础的大语言模型
Hugging Face	一家开发和／或托管大量 开源 Transformer 软件和模型的<lemma
ICEWS	综合冲突预警系统，既是一个名称，也是一个ᵊᵉᶻᵊ 美国国防高级研究计划局的大型冲突预测项目（2008—2011年）专注于 亚洲，以及随后开发的全球性数据集来自 目前每周在 open- 上更新的技术。 访问 Dataverse 网站
	续……

	术语表和首字母缩略词
iData	PITF 生成的 ICEWS 数据政府版本 使用洛克希德和 BBN 软件
JSON	一种非常通用且如今广泛使用的数据交换格式 https: //www.json.org/json-en.html
KEDS	堪萨斯事件数据系统，首个自动生成的编码人员 数据到 20 世纪 90 年代初促成了经同行评审的学术出版- lication
Leidos	一家大型美国国防承包商目前负责该项目的 pro- ICEWS 和 iData 事件数据集的介绍
LLM	大语言模型：非常庞大的基于神经网络的模型 使用海量文本训练而成
ML	机器学习
洛克希德	一家美国大型国防承包商，其系统赢得了最初的 美国国防高级研究计划局 ICEWS 竞赛，因此连续数年 参与了ICEWS数据集的制作
NER	命名实体消解/识别，一项自然语言处理问题
NGEC	新一代事件编码器，本项目的工作名称
NLP	自然语言处理
NSF	美国国家科学基金会
	未完待续.....

	术语表和首字母缩略词
PETRARCH	一组于2010年代中期开发的实验性事件编码器使- 使用开源的斯坦福 CoreNLP 语言解析器和 TABARI 事件和行为体词典
PITF	政治不稳定工作组，一个长期运行（自1995年起）的多方协作项目。 接手 ICEWS 的机构间冲突分析项目，大约在 2020
Plover	我们用来取代 Cameo 的事件数据本体
Prodigy	Explosion AI 推出的一款用于快速标注文本数据的程序
Python	一种广泛用于自然语言处理的通用编程语言 工作以及更广泛的数据分析。 也是一种危害极大的入侵性 佛罗里达大沼泽地的爬行动物
QA	问答，一种我们用于实体识别的机器学习/自然语言处理技术—— 请勿与“质量保证”混淆，不过我们 也需要这样做。
roBERTa	一个更小的，BERT 大语言模型的一个更高效子集
spaCy	由 Explo- 开发的一套全面的开源自然语言处理软件 sion.人工智能。https://spacy.io/
	续。。。

	术语表和首字母缩略词
SVM	支持向量机，一种稳健且广泛使用的机器学习方法 用于分类。这是一系列对数据进行处理的算法，旨在 非常高维空间，而非实体机器。 https:// en.wikipedia.org/wiki/支持向量_机器
TABARI	在美国国家科学基金会支持下开发的 C++ 语言事件编码器 21世纪初的资助，并由洛克希德团队用于赢得 DARPA ICEWS预测竞赛
维基百科	一个开源的在线百科全书，由志愿- 来自世界各地的志愿者，并由维基媒体基金会主办。

表 A4： 术语表和首字母缩略词列表。