arXiv:2304.10486v1 [cs.LO] 1 Mar 2023

CoProver：用于证明构造的推荐系统

Eric Yeh*、Briland Hitaj、Sam Owre、Maena Quemener 和 Natarajan Shankar

SRI International，美国加利福尼亚州门洛帕克 94025
{eric.yeh,briland.hitaj,sam.owre,maena.quemener,natarajan.shankar}@sri.com

摘要。 交互式定理证明器（ITP）是形式化方法专家工具箱中不可或缺的工具，为证明的构造和（形式化）验证提供平台。证明的复杂性，加上成功完成这一过程通常所需的专业知识水平，往往会阻碍交互式定理证明器的普及。近期的一系列研究探讨了将基于交互式定理证明器用户活动轨迹训练的机器学习模型作为实现完全自动化的可行途径。尽管这是一个有价值的研究方向，但许多问题仍需人工监督才能彻底解决，因此，运用学习方法为用户提供有用的建议可能更有成效。沿着用户辅助这一思路，我们推出了 CoProver，这是一种基于 Transformer 的证明推荐系统，能够从证明构造过程中的过往操作中学习，同时探索交互式定理证明器中存储的既往证明知识。CoProver 采用神经网络学习得到的序列基于序列的编码来表示序列，捕捉术语之间的长距离关系及其中隐藏的线索。我们将 CoProver 与原型验证系统（PVS）结合，并在两个关键领域评估其性能，即：(1) 下一步证明操作推荐；以及 (2) 给定理论库时的相关引理检索。我们采用分别源自推荐系统和信息检索领域的一系列成熟指标评估 CoProver。结果表明，CoProver 在该领域的推荐任务上成功超越了此前最先进的方法。最后，我们讨论了 CoProver（及类似方法）的可行未来研究方向，例如论证预测、证明摘要等。

1 引言

交互式定理证明（ITP）是数学、计算及其他多个领域中形式化证明的一项成熟技术。尽管交互式定理证明工具

* 本研究的部分内容是在国防高级研究计划局（DARPA）的资助下完成的。本文所载的观点和结论仅代表作者，不应被解读为代表美国政府或 DARPA 的明示或默示的官方政策。

提供了强大的自动化功能和定制能力，但手动引导定理证明器直至证毕仍是一项艰巨的任务。对于经验不足的用户而言，这一挑战意味着要构造数学上优雅的形式化表达、找出合适的证明命令，并诊断证明尝试失败的根本原因。对于专家用户而言，挑战则在于浏览大量形式化内容，从中找出有用的定义和恰当的引理。新手和专家用户都能从证明命令推荐和引理检索中受益，这些功能可以引导证明构造。

本项目旨在通过将 CoProver 引入为一种证明推荐系统，扩展证明技术的应用规模。该系统能够从库、证明上下文和证明目标中识别合适的线索，为交互式定理证明用户提供建议。本项目基于原型验证系统（PVS）这一先进的证明助手所积累的证明技术和证明语料 [34], ，重点针对交互式定理证明中的两项关键任务提供建议：推荐 PVS 命令和引理。第一项任务是根据当前证明状态，推荐专家用户可能采取的命令。可用命令可能多达 100 多条，因此推荐专家可能采用的步骤会很有帮助，尤其是对新手而言。第二项任务是从引理库中找出可纳入证明、并有助于推进证明的引理。目前，系统只考虑用户导入的理论中的引理，而引理的选择依赖于用户对候选理论及其引理的熟悉程度。对于 PVS 通常用于解决的问题，往往有数百个理论，其中包含数千条可供考量的引理。在如此大的规模下，即使是拥有数十年经验的专家用户，也可能不知道所有与其证明相关的引理（甚至不知道它们的名称）。因此，若能建立一种可大规模自动识别相关引理的机制，将十分有益。

为了开发这两种能力，我们利用 NASA 的 PVS 库 [1] (PVSLib) 中的专家证明轨迹。该库是一个大型集合，包含大量经过精心打磨、围绕安全关键应用展开的形式化开发成果；我们还利用 PVS Prelude，这是内置于 PVS 的理论集合。我们旨在通过训练系统模仿用户在这些已完成证明中的决策，捕捉开发者的专业知识和直觉。

我们方法的关键在于运用最新的机器学习技术，以捕捉比以往更长范围内的序列信息。事实上，我们展示了如何将这些方法应用于公式的简单基于序列的编码，从而比以往的序列编码技术更好地捕捉包含命令和库的证明状态之间的关系。

我们首先在第 2 节概述 CoProver 系统，介绍其核心推荐任务。我们将介绍每种方法，并与先前的研究进行对比。第 3 节将介绍如何将序列和

[1] https://shemesh.larc.nasa.gov/fm/pvs/PVS-库/

分发声明 A – 已批准公开发布，无限制分发

状态特征化为公共表示，以便为命令预测和引理检索功能提供输入。第 4 节和第 5 节将具体介绍这些功能的实现方式，第 6 节则通过实验说明其效果。第 7 节介绍这些功能在 PVS 上的部署及其初步用户评估，第 9 节最后讨论未来的发展方向。

2 CoProver 概述



图 1：CoProver 系统。序列经过特征化后，其表示用于提供命令推荐并检索相关引理。

图 1 展示了 CoProver 系统。特征化步骤将序列的结构化逻辑公式及构成证明状态的其他信息（如之前的命令）转换为标记序列，供 Transformer 模型后续处理。该序列被提供给命令预测和引理检索模块。命令预测根据相似状态，识别专家用户在成功证明过程中可能采取的下一条命令。需要导入引理时，引理检索会检查当前状态，并根据人类选择引理以推进证明的历史记录，从指定的库中推荐最相关的引理。

命令预测和引理检索都使用 RoBERTa[28] 对证明状态序列词元进行编码。RoBERTa 是一种基于 Transformer 的神经语言模型 [42], ，能够学习术语的长程序列，这些序列有助于执行下游任务。与 n元语法或其他马尔可夫窗口方法不同，Transformer 使用自注意力机制，使词元能够利用更广泛的相邻词元来生成适用于下游任务的表示。词元级表示是实值向量，可针对多种任务进行调整；它们已用于多个任务，并取得了最先进的性能，例如大语言模型、文本分类和视觉理解。在我们的案例中，基于 RoBERTa 对证明状态进行编码，并将其作为多项分类器的输入，以预测专家用户接下来会采取什么命令。分类器产生的反向传播误差用于调整这些表示，或对其进行微调，使其更适合分类任务。

分发声明 A – 已获批准公开发布，无限制分发

引理检索旨在从现有理论库中找出可能相关的引理，以便将其纳入证明，推动证明取得进展。这项任务之所以具有挑战性，是因为可能有用的引理或许存在于某个理论中，而用户并不知道该理论，或者可能不记得它的名称，因此不会将其导入证明。这种应用场景与信息检索（IR）中的文档检索问题密切相关 [31]，其目标是确定集合中哪些文档与搜索查询最相关。一个特别的挑战在于，信息检索模型依赖词项重叠和词项稀有度启发式来判断相关性。尽管这些启发式所依据的假设适用于自然文档和查询，但它们是否适用于定理证明仍有疑问。为此，CoProver 使用用户引理导入决策构成的训练集，对 RoBERTa 得出的证明状态表示进行微调，从而学习证明状态与引理符号的组合，以便从给定的引理数据库中识别相关引理。

3 数据生成

3.1 数据集

对于命令预测和引理检索，我们都使用了 PVSLib 库中的证明序列。PVSLib 是一个大型形式化开发库，包含涵盖数学和工程学多个领域的定理证明。PVSLib 总共包含 184,335 个证明步骤。值得注意的是，这些都是已经完成并经过润色的证明，因此，经过回溯的步骤序列已被精简，仅保留成功的证明命令序列和导入的引理。

3.2 特征化

为了将逻辑公式转换为适合机器学习的表示形式，我们将结构化表示转换为标记序列。虽然原始 PVS 是一种结构良好的形式语言，但出于多种原因，仍需进行特征化。首先，我们使用 RoBERTa 对 PVS 证明状态和引理进行编码。由于 Transformer 模型的输入宽度固定，采用一种更精简的编码方式，在保留原始语义的同时去除模板化内容，可以让更长的公式无需截断即可输入。许多公式使用的变量名和常量名特定于某个证明或理论；如果模型学习这些名称，其泛化能力就会下降。

分发声明 A – 已获准公开发布，无限制分发

鉴于此，函数和运算符的所有符号名称均按原样保留。常量名和变量名则替换为通用占位符。整数值予以保留。括号等语法结构不纳入特征化形式，因为上述元素的排列顺序大致能够反映原始形式的语法结构。

图 2 展示了这一过程的输入和输出示例。在这里，全称量词和蕴含算子等符号予以保留，而变量 f、high 和 low 则替换为其类型 nat，表示自然数。



图 2：特征化将 PVS 中的公式（左）转换为更适合机器学习的令牌流（右）。

沿用此前基于 Transformer 的编码研究，我们使用特殊令牌<ant>、 <cons>和 <hid>，标记序列中对应于前件和后件公式的词元区域。最后一个令牌对应隐藏公式，这是 PVS 的一项功能，允许用户将某些公式保留起来，使其不受命令操作。

对于命令预测，我们在所有令牌流前加上一个 cmd1 特殊令牌。它充当此任务的偏置项，也可用于对 Transformer 模型进行条件化，以支持多任务学习。图 3 左侧展示了一个特征化序列示例，其中没有前件，只有一个后件。

我们当前的设置采用马尔可夫假设，即只有当前状态足以做出预测。至少从经验来看，了解执行过哪些命令有助于判断接下来会采取哪些步骤，因此，将先前的命令纳入其中可以捕捉到一些非马尔可夫信息。具体做法是在状态表示中，cmd1 标记之后加入用户先前发出的三个命令。

在引理检索实验中，我们修改了上述过程，使常量和变量可以替换为其类型名称。这样做是为了支持按类型匹配，因为导入引理的参数也需要按类型匹配。高阶类型和自定义类型目前用占位符表示。如何处理高阶类型匹配以及基于谓词的方法等高级类型操作，将留待未来研究。

分发声明 A – 已批准公开发布，无限制分发

我们使用 Huggingface Transformers 库中的字节对编码（BPE） [40,10]来训练一个针对特征化公式（序列和引理）定制的词元词汇表 [45] 。BPE 将单词编码为字节对序列，而不是单个词元。这样可以缩小所需词汇表的规模：罕见词可以用字节对编码，无需记忆；常见词则可以整体编码，以提高效率。这也使模型能够容纳新术语，因为它们可以由合适的字节对序列组合而成。

与许多其他基于 Transformer 的编码器一样，RoBERTa 首先以自监督方式进行训练。给定无标签数据，可以通过能够确定真实标签的操作生成带标签的数据。掩码语言建模就是这样一种任务：通过遮蔽单词的身份，可以构建一个监督预测问题，然后让模型预测该单词 [7]。在大型语料库上进行这种自监督训练后，得到的表示能够捕捉该领域的分布信息，从而使下游组件的训练更加容易。

在我们的实验中，语言模型在数据集上训练了 1，100，000 步，使用了用于训练 RoBERTa 的默认自监督语言任务集。

4 命令预测

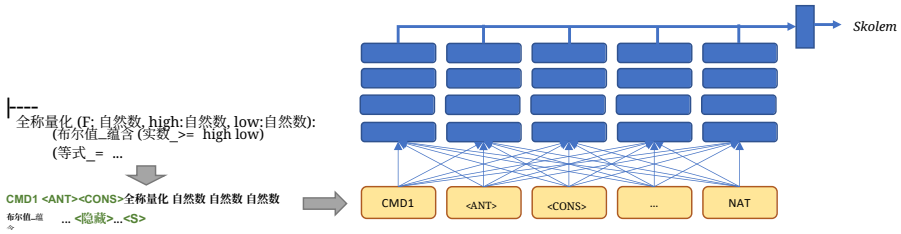

图 3：对序列进行特征化，然后使用重复的自注意力创建表示，以捕获用于预测下一个命令的信息的过程。

命令预测任务的目标是：根据证明中的当前步骤，识别专家用户接下来最可能执行的命令。为进行此预测，我们使用 Huggingface [45] 中实现的 T5 序列到序列训练框架 [36] 。这里，将特征化证明状态的 RoBERTa 编码作为输入，以输出专家用户在给定证明状态时所执行的命令。图 3 展示了如何进行命令预测。首先，序列被

特征化为标记序列，再通过反复应用自注意力，将其转换为适合分类的表示。随后，分类器对这些表示进行整合，并输出最可能的下一个命令。与大多数分类器一样，可以输出置信度最高的前 N 个假设，从而生成一组预测结果。

5 引理检索

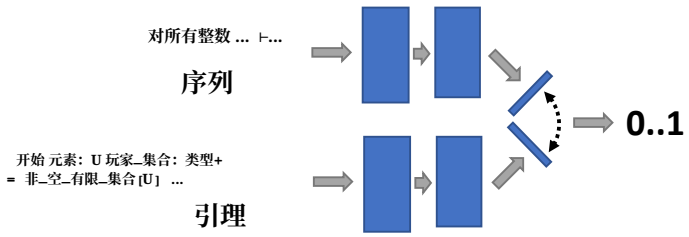

图 4：用于判断某个引理是否与给定序列相关的孪生网络架构。

如前所述，我们将引理检索视为信息检索问题，其中证明状态充当查询，用于在可用引理库中进行检索。鉴于 PVSLib 证明跟踪中包含大量用户执行的引理导入记录，我们可以利用这些记录训练一种神经信息检索方法 [32], ，使其从这些示例中学习相关性排序函数。CoProver 采用这种方法，因为用于确定查询与自然语言之间相关性的假设可能并不适用于匹配证明状态与引理。监督式神经网络架构可以学习用于评估引理与序列相关性的最佳特征组合。图 4 展示了如何对序列和引理进行编码与比较，以评估相关性。该架构遵循在 SBERT [37] 框架中实现的孪生网络 [5], 。给定两个输入，孪生网络会将它们编码为公共表示，通常通过余弦相似度或点积等张量运算比较输入表示，从而计算得分。监督训练使该架构能够学习有助于计算得分的特征组合及其权重。当使用余弦相似度等快速运算计算得分时，比较过程可以极为高效，因为大部分表示都可以预先计算。这种方法已用于为数据量庞大的任务学习排序函数，例如利用点击数据对查询和文档进行排序 [16]。在 CoProver 中，表示引理和序列的逻辑公式的标记序列会被传入 RoBERTa 编码器，以构建词元级表示。这些

表示经过平均，得到一个用于表征引理或序列内容的单一向量。最后，使用余弦相似度对引理向量和序列向量的相关性进行评分，其中 0 表示不相关，1 表示相关性最高。

6 实验与结果

我们详细介绍命令预测和引理检索任务的实验设置与结果。

6.1 命令预测

我们从完整的 PVSLib 证明库中抽取了 20，000 个证明步骤，以创建规模适中的训练集 [2]。随后，我们从这些配对中随机抽取 90% 作为训练数据，其余 10% 则作为留出测试集。

对于 CoProver，我们使用四张 NVIDIA GeForce RTX 3090 显卡，通过 PyTorch Lightning [3]实现的分布式数据并行训练（DDP） [26] ，在这些数据上训练了 10 个周期。由于 DDP 中每张显卡都包含各自的模型，我们选取了验证集误差最小的模型。作为比较基线，我们遵循了以往关于策略预测的研究 [11] ，并使用基于序列中观察到的符号的词频-逆文档频率（TF-IDF） [30,21]加权特征计数训练分类器。TF-IDF 综合考虑词项的出现频率及其相对于整个集合的区分能力。在这里，序列表示为一个向量，其中每个维度对应词汇表中的一个符号，并按 TF-IDF 加权。为确保比较公平，这些特征均来自 CoProver 使用的同一特征化令牌流。为增强此类方法的效果，我们针对基线尝试了多种分类器：线性支持向量分类器（Linear SVC）、使用径向基函数核（RBF）的支持向量机、使用多项式核（Poly）的支持向量机，以及 k近邻分类器（k-NN）。除非另有说明，我们使用了 Scikit-Learn[4] 的默认参数实现。对于 k近邻分类器，我们采用了距离加权变体 $n = 5$，遵循以往研究 [11]。

表1显示了各方法在有序列和命令历史以及没有命令历史的情况下，测试命令预测的准确率。我们

[2] 使用更大的样本进行的初步实验表明，使用较小数据集训练的系统在性能上并无差异
[3] https://www.pytorchlightning.ai/ [4] https://scikit-learn.org/

表1：采用和不采用命令历史信息时，各方法的命令预测测试准确率。

方法	准确率（命令历史） + 序列	准确率，仅序列	准确率，仅命令历史
线性支持向量分类器	$0.30 \pm 1.1 \times 10^{-2}$	$0.20 \pm 9.1 \times 10^{-3}$	$0.30 \pm 1.1 \times 10{-}2$
支持向量机（径向基函数）	$0.29 \pm 1.0 \times 10^{-2}$	$0.22 \pm 9.5 \times 10^{-3}$	$0.30 \pm 1.0 \times 10^{-2}$
支持向量机（多项式核）	$0.20 \pm 8.9 \times 10^{-3}$	$0.20 \pm 8.9 \times 10^{-3}$	$0.22 \pm 1.0 \times 10^{-2}$
k-NN	$0.28 \pm 1.0 \times 10^{-2}$	$0.19 \pm 8.6 \times 10^{-3}$	$0.27 \pm 9.6 \times 10^{-3}$
CoProver	$\mathbf{0.48 \pm 7.3 \times 10^{-3}}$	$0.28 \pm 9.8 \times 10^{-3}$	$0.21 \pm 9.3 \times 10^{-3}$

发现，使用完整序列和命令历史时，CoProver 的预测比基线方法更准确。值得注意的是，基线方法的大部分准确率都来自命令历史本身，而 CoProver 能够整合序列和命令历史，其表现显著优于次佳基线方法 k近邻。各方法的方差通过自助重采样估算 [8]，显著性则通过双样本 t-检验并使用 $\alpha = 0.001$ 确定。

表2：使用融入更多结构信息的特征时，基线方法的命令预测准确率（从左到右）。

方法 $\mathbf{n=1}$	$\mathbf{n=2}$	$\mathbf{n=3}$	
线性支持向量分类器	0.30	0.37	0.30
支持向量机（径向基函数）	0.29	0.32	0.33
支持向量机（多项式核）	0.20	0.18	0.19
k-NN	0.28	0.30	0.32

为了确定结构信息对这项任务的重要性，我们尝试了最大n-gram阶数逐渐增加的 TF-IDF 序列特征化。这里，n-gram 特征化由长度为 n 的所有符号序列构成。当 $n = 1$时，仅使用单个符号，因此不编码任何结构信息。相比之下，$n = 3$ 使用序列中最多三个符号的所有已观测组合，包含了更多的序列结构。表 2 按最大n-gram 阶数列出了各分类方法的准确率。除使用多项式核的支持向量机（支持向量机（多项式核））外，所有方法都受益于结构信息的增加。我们怀疑，该模型表现较差可能是因为多项式核引入了更多超参数，从而大大增加了在稀疏数据上过拟合的风险 [14]。

推荐系统通常会向用户呈现最相关的 N 条预测，而不是只呈现最优推荐。在许多领域，用户

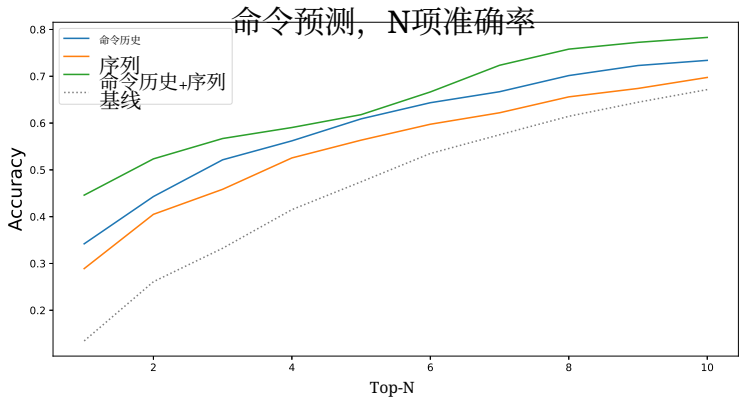

图 5：按方法划分的命令预测测试准确率，分别显示包含和不包含命令历史信息的结果。

可以轻松扫描一组候选项并选出最佳项。或者，在其他情况下，多个候选项提供的上下文也可能很有用。为了评估 CoProver 在这种模式下的表现，我们评估测试集的前 N 项准确率：如果正确的预测位于前 N 项预测中，就视为匹配。图 5 展示了 CoProver 系统在不同 N 大小下的准确率，其中仅使用命令历史（cmdhist）、序列（sequent）以及两者（cmdhist+sequent），N 的取值范围为 1 到 10。作为比较，我们还纳入了基线猜测：将训练集里最常见的前 N 条命令用作候选窗口。我们发现，结合序列和命令历史信息的准确率始终高于单独使用其中任一项，而所有方法的表现都优于基线。值得注意的是，超过 70% 的情况下，正确的命令都出现在前 7 项响应中。

6.2 引理检索

对于引理检索，我们从 PVSLib 证明步骤中筛选出会将引理导入证明的引理命令。我们将分析范围限定在显式导入：虽然还有其他若干命令也会导入引理，但它们是在执行操作时隐式完成导入，且可能产生额外副作用。在这 20,221 个步骤中，我们随机选取了 12,132 个用于训练引理检索模型，其余 8,089 个用于测试。这些证明引用了 PVSLib 和 PVS Prelude 中的库，共有 9,468 个候选引理。

该模型使用 PVSLib 中证明序列训练集里的引理命令进行训练。对于每次引理调用，我们都会记录证明中该位置的序列

以及所引用引理的名称。PVSLib 的证明中共有 20,221 对引理。这些数据包括用户发出导入引理命令时的序列状态，以及引理名称。我们将其随机划分为 12,132 个训练对和 8,089 个测试对。推荐基于 PVS 和 PVSLib 理论构成的组合库，其中共有 747 个理论和 9468 个可用引理。

在训练孪生网络时，训练调用中观测到的序列和引理的得分为 1。我们采样了数量相等的序列—引理对，并将其得分设为 0，理由是：随机选取的引理很可能与推进给定序列无关。

我们在测试对上评估了所得网络。我们使用平均倒数排名（MRR）来衡量每种方法的表现。MRR 是信息检索中常用的评估指标，用于评估不同方法能否将语料库中的相关文档排在较高位置 [31]。MRR 根据给定序列时按相关性排序的得分中真实标签引理的排名位置计算。对于一个给定的测试对，其中包含序列 s、选定的引理 l_{gt}和引理库 L 以及 $l_{gt} \in L$，我们使用孪生网络来评估引理 $l \in L$ 与该序列的相关性，即 $f_{rel}(s, l)$。我们据此对所有引理 L进行排名，

$$MRR = \frac{1}{N} \sum_{i}^{N} \frac{1}{r_i}$$

其中， r_i 是该 i^{th} 引理对中真实标签引理的排名。因此，平均倒数排名越高，表示评分方法越能将较高的相关性赋予真实标签文档。

方法	平均倒数排名
基线	0.0015
计数	0.0030
TF-IDF	0.043
CoProver	**0.51**

我们发现，在根据序列识别相关引理方面，CoProver 方法优于其他方法，包括先前工作所使用的 TF-IDF 词袋表示法。MRR 为 0.51 时，对应的平均排名为 1.98，这意味着相关引理在基于得分对所有引理进行排序时大约位于第 2 位。

7 部署

CoProver 目前以由 Web 应用程序支持的服务形式提供给 PVS。我们将命令预测和引理检索模型封装为基于 Flask[5] 框架的 Web 服务。若要请求命令预测或相关引理，调用组件会将当前序列转换为 JSON 格式的消息。假设会以 JSON 对象的形式返回，并直接呈现在 PVS 中。

我们正在准备这个由推荐系统增强的 PVS 版本，以开展大规模用户研究，评估这些功能对最终用户有多大用处。不过，内部开发人员发现，在展示排名前 5 的命令时，该功能通常能给出不错的建议。

本研究使用的代码和数据均已开源，可在 https://github.com/coproof 获取。

8 相关工作

过去十年间，将机器学习应用于自动演绎的研究取得了显著进展。Markus Rabe（Google Research）在 2021 年 CADE 会议上受邀作报告，对相关进展进行了总结 [35]。此前关于在定理证明中使用机器学习的研究，可根据机器学习的预测目标分类如下

1. 学习搜索启发式（E-prover [38],SAT/SMT 求解器 [3]）：系统 ENIGMA [18], MaLeCoP [41]和 FEMaLeCoP [22] 为基于表格法的证明器 LeanCoP [33] 添加了朴素贝叶斯分类器，用于选择子句。图神经网络已被用于预测某个子句集中的哪些子句属于不可满足核心集合 [39] ，并为 SMT 求解器提供引导 [1]。

2. 从事实库中选择前提（DeepMath [17],CoqHammer、HOLy-Hammer、HOList [2], Thor [19]）：一些证明助手会针对每个子目标，调用自动化证明工具（定理证明器和 SAT/SMT 求解器），并提供一组背景引理（即前提）。如果前提过多，这些自动化证明工具可能会失败。机器学习已被用于从背景库中找出最有希望选用的前提 [24]。与下文的策略选择一样，前提选择采用了多种学习技术，包括序列、树和图表示 [43]。

[5] https://flask.palletsprojects.com/

3. 步骤或策略选择（GPT-f、Holophrasm [44], CoqGym [47], HOL4RL [46], HOList [29], GamePad [15], Tactic-Toe [11], 证明合成 [9,20,25]）：交互式证明助手通过对目标应用策略来构建证明树，并生成零个或多个子目标。SEPIA 系统 [13]仅通过分析证明来预测 Coq 证明助手的策略。GPT-f 使用 GPT-3 Transformer 模型，在 Metamath 语料库中的目标/证明对（经合成证明扩充）上进行训练，从而根据目标预测证明。随后，系统将模型建议的证明步骤应用于证明树中的未解决子目标，以构建证明树。该系统能够为 Metamath 语料库中的 23 个定理找到更短的证明。CoqGym [47]使用规模大得多的训练语料库，其中包含来自各类 Coq 库的 71,000 个证明。TacticToe [12] 在 HOL4 库中的证明上进行训练，并将基于 k 近邻的策略预测与 A* 搜索相结合；在某些情况下，与使用自动化证明工具的其他方案相比，它能给出更清晰易懂、也更易维护的证明。IsarStep [27] 使用 Transformer 编码，在声明式 Isar 证明中识别中间公式。ML4PG [23]则运用 K 均值聚类等无监督学习技术，从高阶逻辑证明中提取有用的统计模式。

从用户活动轨迹中学习预测下一条命令，已在 TacticToe [11] 等工作中得到研究。然而，这些工作的重点在于系统完全自动化证明的能力，性能则通过无需人工协助即可完成的证明数量增幅来衡量。据我们所知，先前的工作尚未研究这些系统在预测用户为取得进展而采取的下一步操作方面表现如何。

此外，先前的工作还采用“词袋”方法对序列进行特征化，即将证明状态表示为序列中出现的词元直方图。尽管相邻词元对的直方图可以保留一定的序列性，但这种方法可能会削弱较长范围的结构信息，而这些信息对于预测等任务可能很有价值。相比之下，CoProver 利用神经语言建模技术，能够学习较长范围内词元的序列组合。

将引理检索视为信息检索问题的做法，已见于先前的研究 [4,11]。与大多数信息检索方法一样，这些研究使用在序列（查询）和引理（文档）中出现的术语的加权直方图来表征它们。尽管这种“词袋”方法消除了序列性，因而也舍弃了结构信息，但事实证明，查询与候选文档之间词汇用法的重合度，可以很好地近似搜索相关性。术语的相关性权重采用词频-逆文档频率（TF-IDF） [30,21], 计算，该方法综合考虑术语的出现频率及其在整个文档集中的区分能力。相似度比较通常使用一种计算高效的方法，

例如严格的点积或余弦相似度，从而能够针对大量候选文档高效地执行查询。

然而，词袋建模和 TF-IDF 加权方案基于一些假设，这些假设针对的是相关性在自然语言查询和文档中的表现方式。引理和序列中观察到的逻辑公式可能不会呈现相同的行为，尤其是在判断某个引理是否有助于推进序列中的证明进展时。事实上，神经信息检索侧重于使用监督式查询和文档对来学习相关性函数，而这些函数可能无法通过标准信息检索建模中的假设来捕捉。

9 结论

在这项工作中，我们展示了如何利用对证明状态进行简单的特征化来完成两项推荐任务：预测下一条命令和检索相关引理。对于命令预测，CoProver 的方法已被证明优于先前的方法，准确率显著更高。在推荐系统的应用场景中，展示排名靠前的 $3-5$ 条命令是合理的数量；在验证集上，这些候选窗口覆盖了原始预测中 50% $-$ 70% 的正确结果。与基于用户交互轨迹训练的系统一样，系统往往能找到许多原始用户未曾考虑过的解法。正如一位内部用户所评论的，在一个收敛证明中使用自动化锤子（grind 命令）出乎意料，却最终完成了证明。同样，与基于信息检索的基线方法相比，使用 CoProver 的特征化和神经学习方法可以显著提升引理检索的性能。

尽管取得了这些成果，我们仍注意到，神经学习机制未必学到了深层推理结构，更可能学到的是复杂的结构线索。事实上，一项对大语言模型的分析发现，它们是令人印象深刻的记忆机器，却无法进行算术运算 [6]。对命令预测任务中的注意力头进行粗略检查后发现，模型的注意力权重并不总是符合经验丰富的用户对于哪些因素应决定证明方向的直觉。

证明命令推荐的未来发展方向包括识别这些命令所使用的参数。这些参数主要由要使用的公式构成，但在某些情况下还需要更复杂的参数。将这一功能与解释机制结合起来尤其值得关注。或许最简单的解释功能是：在后台运行前 N 个命令，并显示结果，从而提供一种预览功能，使用户能够

分发声明 A – 已批准公开发布，无限制分发

看到各种结果的范围。当与衡量证明完成度的启发式方法结合时，这可能有助于培养直观的理解。

据我们所知，我们是首个将引理检索作为信息检索问题进行评估和考察的研究。以往关于引理检索的研究，重点在于所选引理能否推动自动求解器中的证明逐步完成。虽然这确实涵盖了多个引理可能与自动完成证明相关的情况，但对于交互式定理证明器所处理的问题，自动完成证明可能并不可行。不过，引入一个引理可以推动证明进展，因此也可视为相关，这与检索到合适的文档能帮助查询用户完成任务类似。

为此，我们展示了如何利用神经网络训练架构，根据序列判断人类用户在推进证明时会选择哪个引理。这种方法能够为引理提供更好的相关性排序，优于以往研究中讨论的表示和评分方法。

需要注意的是，根据我们的数据设置，只有用户选取的引理才被视为与给定序列相关。这忽略了其他引理也可能同样有助于证明的可能性。然而，这是自然语言信息检索语料库中一个众所周知的问题；平均倒数排名等指标用于比较系统性能，而非作为独立的性能衡量标准。

该领域未来可能的研究工作可以着重分析促使序列与引理相匹配的相关结构元素。虽然最终的比较通过余弦相似度计算完成，但最高得分的特征匹配的性质可能难以辨别。特别是，引理和序列的公式之间可能没有明显的重叠，但相关的标记序列可能会映射到同一特征。

未来的研究工作中，我们正在探讨将 CoProver 基于 Transformer 的序列表示应用于指定见证项、证明修复以及制定证明进展度量等任务。

致谢 本材料基于美国国防高级研究计划局（DARPA）根据合同编号 HR00112290064 以及国家航空研究院提供支持的研究工作。本文中表达的任何观点、研究结果、结论或建议均属于作者个人意见，并不一定反映美国政府或 DARPA 的观点。

分发声明 A – 公开发布，无限制分发

参考文献

1. Balunovic, M., Bielik, P., Vechev, M.T.：学习求解 SMT 公式。载于：NeurIPS。第 10338–10349 页（2018） 2. Bansal, K., Loos, S., Rabe, M., Szegedy, C., Wilcox, S.：HOList：一个用于高阶逻辑定理证明机器学习的环境。载于：国际机器学习会议。第 454–463 页。PMLR（2019） 3. Biere, A., Heule, M., van Maaren, H., Walsh, T.（编）：可满足性手册。IOS Press（2009） 4. Blanchette, J.C., Greenaway, D., Kaliszyk, C., Kühlwein, D., Urban, J.：一种基于学习的 Isabelle/HOL 事实选择器。《自动推理期刊》**57**，219–244（2016） 5. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.：使用“连体”时延神经网络进行签名验证。神经信息处理系统进展 **6**（1993） 6. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.：语言模型是少样本学习者。CoRR **abs/2005.14165**（2020年），https://arxiv.org/abs/2005.14165 7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.：BERT：用于语言理解的深层双向 Transformer 预训练。载于：2019 年计算语言学协会北美分会年会论文集：人类语言技术，第 1 卷（长短篇论文）。第 4171–4186 页。计算语言学协会，美国明尼苏达州明尼阿波利斯（2019 年 6 月）。https://doi.org/10.18653/v1/N19-1423，https://aclanthology.org/N19-1423 8. Efron, B., Tibshirani, R.J.：自助法简介。统计与应用概率专论第 57 号，Chapman & Hall/CRC，美国佛罗里达州博卡拉顿（1993） 9. First, E., Brun, Y., Guha, A.：Tactok：具备语义感知能力的证明合成。《ACM 编程语言汇刊》**4**（OOPSLA），1–31（2020） 10. Gage, P.：一种新的数据压缩算法。《C Users J.》**12**（2），23–38（1994 年 2 月） 11. Gauthier, T., Kaliszyk, C., Urban, J., Kumar, R., Norrish, M.：学习使用策略进行证明。arXiv 预印本 arXiv:1804.00596（2018） 12. Gauthier, T., Kaliszyk, C., Urban, J., Kumar, R., Norrish, M.：TacticToe：学习使用策略进行证明。《自动推理期刊》**65**（2），257–286（2021） 13. Gransden, T., Walkinshaw, N., Raman, R.：SEPIA：使用推断自动机搜索证明。载于：国际自动演绎会议。第 246–255 页。施普林格（2015年） 14. Hsu, C.W., Chang, C.C., Lin, C.J.：支持向量分类实用指南。技术报告，国立台湾大学计算机科学系（2003），http://www.csie.ntu.edu.tw/~cjlin/papers.html 15. Huang, D., Dhariwal, P., Song, D., Sutskever, I.：GamePad：一个定理证明学习环境。arXiv 预印本 arXiv:1806.00608（2018）

16. Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.：利用点击数据学习用于网络搜索的深度结构化语义模型。载于：第22届ACM国际信息与知识管理会议论文集。第2333–2338页。CIKM '13，美国纽约州纽约：计算机协会（2013年）。https://doi.org/10.1145/2505515.2505665, https://doi.org/10.1145/2505515.2505665 17. Irving, G., Szegedy, C., Alemi, A.A., Eén, N., Chollet, F., Urban, J.：Deepmath：用于前提选择的深度序列模型。《神经信息处理系统进展》**29**，2235–2243（2016年）18. Jakubův, J., Urban, J.：Enigma：高效的基于学习的推理指导机。载于：《智能计算机数学：第10届国际会议》，CICM 2017，英国爱丁堡，2017年7月17日至21日，论文集10。第292–302页。施普林格（2017年）19. Jiang, A.Q., Li, W., Tworkowski, S., Czechowski, K., Odrzygóźdź, T., Miłoś, P., Wu, Y., Jamnik, M.：Thor：运用自动化证明工具整合语言模型与自动定理证明器（2022年）。https://doi.org/10.48550/ARXIV.2205.10893, https://arxiv.org/abs/2205.10893 20. Jiang, A.Q., Welleck, S., Zhou, J.P., Li, W., Liu, J., Jamnik, M., Lacroix, T., Wu, Y., Lample, G.：起草、构思并证明：利用非形式化证明引导形式化定理证明器（2022年）。https://doi.org/10.48550/ARXIV.2210.12283, https://arxiv.org/abs/2210.12283 21. Jones, K.S.：术语特异性的统计解释及其在检索中的应用。《文献学杂志》**28**，11–21（1972年）22. Kaliszyk, C., Urban, J.：Femalecop：高效的机器学习连接证明器。载于：《程序设计、人工智能与推理逻辑》。第88–96页。施普林格（2015年）23. Komendantskaya, E., Heras, J., Grov, G.：Proof General中的机器学习：接口的对接。arXiv预印本 arXiv:1212.3618（2012年）24. Kühlwein, D., Blanchette, J.C., Kaliszyk, C., Urban, J.：Mash：用于Sledgehammer的机器学习。载于：《交互式定理证明国际会议》。第35–50页。施普林格（2013年）25. Lample, G., Lachaux, M.A., Lavril, T., Martinet, X., Hayat, A., Ebner, G., Rodriguez, A., Lacroix, T.：用于神经定理证明的超树证明搜索（2022年）。https://doi.org/10.48550/ARXIV.2205.11491，https://arxiv.org/abs/2205.11491 26. Li, S., Zhao, Y., Varma, R., Salpekar, O., Noordhuis, P., Li, T., Paszke, A., Smith, J., Vaughan, B., Damania, P., Chintala, S.：PyTorch Distributed：加速数据并行训练的经验。《CoRR》**abs/2006.15704**（2020年），https://arxiv.org/abs/2006.15704 27. Li, W., Yu, L., Wu, Y., Paulson, L.C.：Isarstep：高阶数学推理基准。arXiv预印本 arXiv:2006.09265（2020年）28. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.：RoBERTa：一种经过稳健优化的BERT预训练方法（2019年）。https://doi.org/10.48550/ARXIV.1907.11692, https://arxiv.org/abs/1907.11692 29. Loos, S., Irving, G., Szegedy, C., Kaliszyk, C.：深度网络引导的证明搜索。arXiv预印本 arXiv:1701.06972（2017年）30. Luhn, H.P.：文献信息机械化编码与检索的统计方法。《IBM研究与开发杂志》**1**(4)，309–317（1957年）。https://doi.org/10.1147/rd.14.0309

31. Manning, C.D., Raghavan, P., Schütze, H.：《信息检索导论》。剑桥大学出版社，英国剑桥（2008年），
http://nlp.stanford.edu/IR-book/information-retrieval-book.html 32. Mitra, B., Craswell, N.：（2018年）33. Otten, J., Bibel, W.：leancop：基于精简连接的定理证明。《符号计算杂志》**36**(1-2)，139–161（2003年）34. Owre, S., Rushby, J.M., Shankar, N.：PVS：原型验证系统。载于：《自动演绎国际会议》。第748–752页。施普林格（1992年）35. Rabe, M.N., Szegedy, C.：迈向自动数学家。载于：《自动演绎国际会议》。第25–37页。施普林格，瑞士沙姆（2021年）36. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.：探索使用统一文本到文本Transformer进行迁移学习的极限。《CoRR》**abs/191**010683 3**710683** 2019 1910.( ), http://arxiv.org/abs/ . . Reimers, N., Gurevych, I.：Sentence-BERT：使用孪生BERT网络的句子嵌入。《CoRR》**abs/1908.10084**（2019年），
http://arxiv.org/abs/1908.10084 38. Schulz, S.：E——一个天才定理证明器。《人工智能通信杂志》**15**(2/3)，111–126（2002年）39. Selsam, D., Bjørner, N.：利用不可满足核心预测引导高性能SAT求解器。载于：《可满足性测试理论与应用国际会议》。第336–353页。施普林格（2019年）40. Sennrich, R., Haddow, B., Birch, A.：利用子词单元对稀有词进行神经机器翻译。载于：第54届计算语言学协会年会论文集（第1卷：长篇论文）。第1715–1725页。计算语言学协会，德国柏林（2016年8月）。
https://doi.org/10 1162 41 18653 1 16 1162, 16 . /v /P - https://aclanthology.org/P - . Urban, J., Vyskočil, J., Štěpánek, P.：Malecop机器学习连接证明器。载于：《自动推理分析表与相关方法国际会议》。第263–277页。施普林格（2011年）42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.：注意力就是你所需要的一切。《神经信息处理系统进展》**30**（2017年）43. Wang, M., Tang, Y., Wang, J., Deng, J.：通过深度图嵌入进行定理证明的前提选择。arXiv预印本 arXiv:1709.09994（2017年）44. Whalen, D.：Holophrasm：用于高阶逻辑的神经自动定理证明器。arXiv预印本 arXiv:1608.02644（2016年）45. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.：Transformers：最先进的自然语言处理。载于：2020年自然语言处理经验方法会议：系统演示论文集。第38–45页。计算语言学协会，线上（2020年10月6日）46 2020），
https://www.aclweb.org/anthology/.emnlp-demos. . Wu, M., Norrish, M., Walder, C., Dezfouli, A.：Tacticzero：通过深度强化学习从零开始学习证明定理。arXiv预印本 arXiv:2102.09756（2021年）47. Yang, K., Deng, J.：通过与证明助手交互来学习证明定理。载于：《机器学习国际会议》。第6984–6994页。PMLR（2019年）