深度网络引导的证明搜索

Sarah Loos[1], Geoffrey Irving[1], Christian Szegedy[1], and Cezary Kaliszyk[2]
[1]谷歌研究
{smloos|geoffreyi|szegedy}@google.com
[2]因斯布鲁克大学，奥地利
cezary.kaliszyk@uibk.ac.at

摘要

深度学习技术是近年来若干重大人工智能进展的核心，包括物体识别和检测、图像说明、机器翻译、语音识别和合成，以及下围棋。

自动一阶定理证明器可以帮助数学定理的形式化和验证，并在程序分析、理论推理、安全、插值和系统验证中发挥关键作用。

在此，我们建议在定理检验器的证明搜索中采用基于深度学习的指导方法E.我们在现有的米扎尔语句的ATP证明的痕迹上训练和比较了几个深度神经网络模型，并在证明搜索过程中使用它们来选择处理的条款。我们给出了实验证据，通过一个混合的、两阶段的方法，基于深度学习的指导可以大大减少证明搜索步骤的平均数量，同时增加证明定理的数量。

使用一些利用深度神经网络的证明指导策略，我们找到了7.36%的米扎尔数学库定理的一阶逻辑翻译的一阶证明，这些翻译以前没有ATP生成的证明。这使得语料库中具有ATP生成的证明的语句比例从56%增加到59%。

1 简介

1.1 激励

在过去的二十年里，各种大型的计算机可理解的推理知识库已经被开发出来（Harrison等人，2014）。除了公理、定义和猜想之外，这类语料库还包括在选定的逻辑基础上得出的证明，其细节足以被机器检查。这要么以前提-结论对的形式给出（Sutcliffe, 2009），要么以程序和中间步骤的形式给出（Wenzel, 1999）。许多这些形式化证明的开发需要几十个人的时间，其规模以数万条人名定理计算，完整的证明包含数十亿个低级推理步骤。

这些形式化证明库对基于人工智能的方法也很有趣，其任务包括概念匹配、理论探索和结构形成（Autexier & Hutter，2015）。此外，人工智能方法可以通过自动推理来增强：高效的一阶*自动定理证明器*（ATPs）（Kovács & Voronkov，2013）的开发进展使得它们不仅可以作为重做形式证明的工具，还可以找到遗漏的步骤（Urban，2006）。再加上从交互式系统的丰富逻辑到ATP的简单逻辑的证明转换，这成为某些交互式证明器中常用的工具（Blanchette等人，2016）。许多涵盖数学和计算机科学的重要证明发展都是利用这种技术创造的。例如开普勒猜想的形式化证明（Hales等人，2015），或者seL4操作系统内

核的正确性证明（Klein等人，2010）。

尽管所采用的证明计算方法是完整的，但现代ATP在大量事实库的情况下表现不佳。为此，基于人工智能的启发式和学习技术被用来从外部预选公文（Kühlwein等人，2012）。即使是外部选择，Alama等人（2012）的研究表明，ATP只能在Mizar数学库（Grabowski等人，2015）的证明中找到最多20个人类步骤的证明，而该库包含了许多有数百个步骤的证明。与外部法则选择相比，在内部引导ATP具有更好的潜力，因为完整的证明状态是已知的。对于tableaux微积分，*机器学习连接证明者*（MaLeCoP）（Urban等人，2011）表明，使用机器学习来选择下一步，可以将证明中的推理数量平均减少20倍。由于目前最有竞争力的ATP不是基于tableau，而是依赖于叠加计算，在本文中，我们调查了使用深度神经网络指导最先进的自动验证器E（Schulz，2013）[1]。

深度卷积神经网络（LeCun等人，1998）是过去几年中几个最新人工智能突破的核心。深度卷积网络在语音识别（Hinton等人，2012a）和自然语言处理（Kim，2014）中发挥了作用。由于深度卷积神经网络的介入，物体识别在大型基准上的表现已经达到了人类水平（Krizhevsky等人，2012；Szegedy等人，2015；He等人，2015）。DeepMind的AlphaGo（Silver等人，2016）通过利用评估棋盘位置的深度卷积神经网络，在下围棋时表现出超人的表现。深度分层卷积网络在语音和声音生成方面有很大的改进（van den Oord等人，2016）。

深度神经架构的广泛适用性表明它们在指导数学定理证明的组合搜索方面也有潜在的用处。

1.2 贡献

我们首次评估了深度网络模型作为自动定理检验器内部的证明指导方法。我们描述了在一个依靠快速探索的系统中成功整合相对缓慢的神经网络模型所需要解决的技术问题。

实验结果是在一个从Mizar数学库（Grabowski等人，2010）翻译成一阶逻辑（Urban，2006）的大型数学语句语料库上给出的，该语料库涵盖了基础数学的重要部分，从离散数学和数学逻辑到微积分和代数。

由于神经网络推理的相对昂贵的性质，在证明搜索过程中应用深度网络模型是一个挑战。在一个条款被神经网络评估的过程中（在CPU上），验证人可能会执行几百或几千个叠加步骤。这意味着，通过神经网络的证明指导必须提供非常好的建议，才能对证明过程产生积极的影响。在本文中，我们描述了两个重要的技术细节，它们对于使用神经网络指导获得更好的结果至关重要。

首先，与单独使用神经网络来挑选下一个被处理的句子相比，网络引导和硬编码启发式方法的交错使用仍然有助于提高性能。

第二，我们使用了一个两阶段的方法，即在网络指导阶段之后，还有一个只使用快速启发式方法的阶段。如果没有这个第二阶段，网络指导方法往往只处理快速启发式搜索的1%的条款。虽然网络指导阶段很慢，但它的高质量为搜索过程的其余部分提供了一个更好的起点。

[1] 1.9.1pre014版本

从本质上讲，如果我们用一半的时间来引导搜索，用剩下的时间从这个起点开始完成，我们就会比低质量的条款选择有明显的改进，因为低质量的条款选择在搜索过程中迷失方向的几率较大，会遗漏对完成证明至关重要的关键条款的选择。

2 相关工作

Suttner & Ertel (1990)提出了一种基于多层感知器的证明搜索指导算法，该算法建立在手工设计的特征之上。Denzinger等人（1999）调查了基于学习和知识库的方法来指导自动定理证明器。Schulz(2000)提出了一种基于*k-NN*的学习方法，在选择下一个要处理的条款的规范化形式上手工制作了一些特征。最近，对于高阶逻辑，有人提出了基于天真贝叶斯和粒子群的方法，用于Satallax中给定条款算法的内部指导（Färber & Brown, 2016）。FEMaLeCoP使用天真贝叶斯来指导其表格证明搜索（Kaliszyk & Urban, 2015b）。对于定理检验器E来说，为了更好地选择处理过的条款，提出了一个手工设计的相似性函数，其中有几个学习过的权重。所有这些方法都依赖于精心设计的特征，而我们的深度学习方法则从文本表示或证明条款的语法树中进行端到端的学习。

最近，已经有一些尝试将深度学习应用到定理证明和一般推理中。卷积神经网络被成功地用于Mizar的前提选择（Alemi等人，2016），我们在这里使用类似的模型进行证明指导。Whalen（2016）提出了一个完整的基于神经网络的定理检验器架构，利用GRU（Chung等人，2015）网络来指导MetaMath系统（Megill，2007）的tableau式证明过程的证明搜索。Rocktäschel & Riedel（2016）在一些玩具问题上测试了一个全差分定理验证器。Rock- täschel等人(2015)提出了通过深度学习进行缺陷分析。

3 定理证明的预演

3.1 一阶逻辑验证器E

我们的目标是基于饱和度的一阶逻辑定理检验器E（Schulz，2013）。E将所有的输入预处理成*条款正常形式*（CNF），并以这种形式进行证明搜索。如果一个一阶公式是一个子句的结合体，其中每个子句是一个字词的二择一，每个字词是一个k-*ary*谓词符号对*k*个术语的应用（可能是否定的），并且术语是递归的变量或应用于术语的函数，那么它就是CNF。

E的证明搜索是由各种论据参数化的，其中最重要的是我们在这项工作中关注的*条款选择启发式*。在基于饱和度的定理证明中，有两组条款被操作：一组是初始化为问题的条款

形式的*未处理条款*，另一组是初始化为空的*处理条款*。在每一步，启发式选择一个未处理的子句，产生它的所有后果（将它们添加到未处理的子句集合中），然后将该子句移到已处理的子句集合中。启发式处理子句的顺序在很大程度上影响了找到一个证明所需的时间。E的自动模式检查问题并选择一个子句搜索启发式和其他可能表现良好的参数。

E中最成功的启发式方法是混合启发式方法，它通过轮流的方式，以不同的标准来选择下一个子句。例如，一个简单的混合启发式方法

可能会在以先进先出的顺序选择子句和从未处理的子句集中选择最短的子句之间交替进行。这两种启发式方法单独使用时可能会错过选择重要的条款，但结合使用时就会有明显的改善。E验证器通过对所有未处理的子句进行单独的排名来实现混合启发式方法，以满足每个选择标准。然后，它从每个排名中处理最高的条款，允许选择标准的任意交错。

在本文中，我们证明了在固定的时间和内存限制下，当我们将基于机器学习的分类加入到子句选择启发式中时，我们可以证明更多的语句。

3.2 米扎尔首发问题

Mizar数学库（MML）（Grabowski等人，2015）是在Mizar系统（Bancerek等人，2015）之上开发的形式化数学库。MML是当今最大的形式化证明库之一，涵盖了大多数数学的基本领域和一些计算机科学的证明。Mizar的基础已经被Urban（2006）翻译成一阶逻辑，使得MML成为机器学习与自动推理相结合的实验的第一个目标。Kaliszyk & Urban (2015a)在MML 4.181.1147版本上对AI-ATP方法进行了最广泛的评估。翻译成一阶逻辑的57,882条Mizar定理的集合已经与大约90,000个其他公式（主要用于表达Mizar的类型系统）一起按时间顺序排列。

本文中用于指导E验证器的一阶问题都是在Kaliszyk & Urban（2015a）中发现的不同证明。这包括基于ATP可以重做的人类依赖关系的证明和通过在预测的依赖关系上运行ATP发现的证明。在FOL中的57,882个Mizar定理中，有32,521个（约56%）至少有一个ATP证明，总共有91,877个不同的ATP证明。

我们使用这91,877个证明来训练和评估我们的神经网络。我们通过猜想将它们随机分配到一个训练集和一个验证集，使用90%-10%的分割。通过按猜想分割证明，我们避免了对我们的评估集的污染。如果一个猜想有
多个证明，那么所有这些证明都被分配到训练集，或者都被分配到评估集。训练集包含29,325个猜想的82,718个独特证明。剩下的3196个猜想和它们的9159个证明被分配到评估集。当我们在这个评估集上评估条款选择启发法时，我们把它们称为*容易的语句*，因为它们之前已经被一些ATP方法证明了。我们把没有找到ATP证明的25,361个猜想称为*困难的陈述*。这些定理加在一起构成了FOL中的57,882个Mizar定理。

4 深度网络

在这里，我们描述了数据收集、网络架构和我们的深度学习模型的训练方法，以纳入Schulz（2013）的自动定理检验器E的证明搜索程序。

我们重新使用了在Alemi等人（2016）的早期相关前提选择工作中被证明是成功的几个架构和训练选择。我们的模型有两个输入：一个待证明的否定猜想和一个未处理的条款，都是CNF。使用嵌入网络将每个输入简化为一个固定大小的向量，然后一个组合器网络将两个嵌入向量合并为一个分数，作为条款选择的启发式方法。理想情况下，分数取决于当前处理过的条款集，但我们将输入限制为否定的猜想

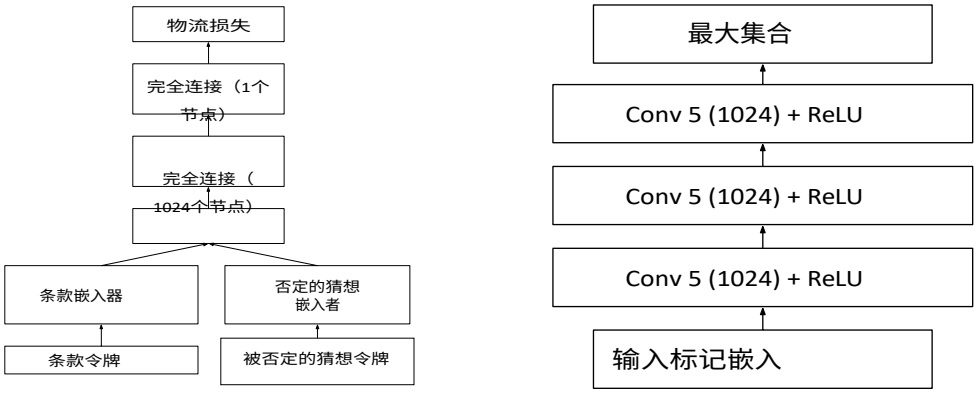


图1：左图：带有条款和否定猜想输入的整体网络。右图：递归嵌入模型。其他实验使用WaveNet或递归嵌入网络。

和条款，以达到简单和快速的目的。因此，整体架构是

$$v_c = f_{emb}(\text{条款}, w)_c$$
$$v_{nc} = f_{emb}(\text{negated-conjecture}, w_{nc})$$
$$p(\text{useful}|c, nc) = \sigma(g_{comb}(v_c, v_{nc}, w))_{comb}$$

其中f_{emb}和g_{comb}是嵌入和组合器网络，w_c，w_{nc}，w_{comb}是要学习的权重，$\sigma(z) = 1/(1 + e^{-z})$是sigmoid（图1，左）。请注意，我们使用相同的结构来嵌入否定猜想和句子，但所学到的内容不同。
砝码。

在训练时，如果该条款在证明中被使用，则使用逻辑损失法对输出概率进行训练，即p=1，否则p=0。在证明时间，选择具有最大概率的未处理的条款进行处理。

对于嵌入网络f_{emb}，我们已经尝试了三种架构：

1. 一个简单的卷积网络，有几个卷积层，然后是最大集合。
2. 一个WaveNet（van den Oord等人，2016）风格的网络，可以有效地模拟长距离的依赖关系。
3. 递归神经网络（Goller & Kuchler, 1996），其拓扑结构取决于要评估的公式的语法树。

在第4.3、4.4和4.5节中，在描述了第4.1节中的训练和评估数据后，我们对所评估的模型进行了详细描述，并比较了它们的预测性能。端到端系统只使用最快的模型和那些具有非常好的预测性能的模型进行评估。之后，我们使用最好的模型为Mizar语料库的 "硬子集 "生成新的ATP证明，目前的ATP方法已经失败。

4.1 数据收集

如第3.2节所述，我们将至少有一个证明的32,521个Mizar定理以90%-10%的比例分成训练和验证集。我们的数据集中的证明源于各种一阶ATP。此外，即使是来自于以下方面的证明踪迹

E，预处理的配置（skolemization和clausification）对出现在句子中的符号和它们的形状有很大影响。因此，我们使用E的预处理程序的单一、一致的配置来复制证明，以避免我们的训练集和测试时看到的条款之间的不匹配，当我们用训练好的模型指导搜索时。具体来说，我们对使用附录中详述的Auto208配置生成的证明进行训练。

4.2 架构

我们考虑嵌入网络f_{emb}的三种架构：简单的卷积模型、WaveNet（van den Oord等人，2016）风格的模型和递归网络（Goller & Kuchler，1996）。所有实验都使用相同的组合网络g_{comb}，有一个1024单元的隐藏层：

$$g_{\text{comb}}(v_c, v_{nc}) = W_2\,\text{relu}(W_1\,\text{concat}(v_c, v_{nc}) + b_1) + b_2$$

其中$W_1 \in \mathbb{R}^{1024 \times 2\dim v}$，$W_2 \in \mathbb{R}^{1 \times 1024}$，$b_1 \in \mathbb{R}^{1024}$，$b_2 \in \mathbb{R}$。

在数据收集时，我们收集所有的符号（常数、函数名、变量名、逻辑运算和括号）到一个词汇表中，输入层使用查表法将每个词汇转换为1024维的嵌入向量。嵌入是随机初始化的，在训练过程中学习。我们在条款和否定猜想的嵌入之间共享相同的嵌入向量。卷积和WaveNet模型接收条款和否定猜想作为这些嵌入的序列；递归网络只对CNF树的叶子上的名字使用嵌入。与Alemi等人（2016）不同，我们避免了字符级的嵌入，因为证明搜索可能涉及非常大的条款，计算成本很大。

所有的模型都是用TensorFlow（Abadi等人，2015）和Adam（Kingma & Ba，2014）优化器训练的。

4.3 简单卷积模型

遵循Alemi等人（2016）的前提选择模型，我们的卷积模型（"CNN "为 "卷积神经网络"）具有相对较浅的深度，因为它们在该相关任务上给出了良好的结果。它们由三个一维卷积层堆叠而成，每个层的补丁大小为5（每个输出连接的输入数），步长为1，并有一个整流的线性激活。我们已经尝试了几个具有不同特征维度的模型。

4.4 波网模型

WaveNet（van den Oord等人，2016）是一种特殊类型的分层卷积网络，它采用了扩张卷积和剩余连接。扩张卷积允许长距离的特征依赖，只有适度的开销，因为较高的层具有几

何级数的扩张因子（见图2）。虽然van den Oord等人（2016）使用因果扩张卷积，但我们使用对称卷积，因为我们的任务是识别，而不是生成。我们的WaveNet嵌入网络由3个WaveNet块组成，其中每个块由7个
卷积层按d=1, 2, 4, ...扩展。, 64.我们使用门控激活$\tanh(x)\sigma(x)$
的van den Oord等人（2016），以及层和块的剩余连接（He等人、
2015).对于正则化，一些实验在输入层使用了20%的标记性剔除，并将其作为正则化。

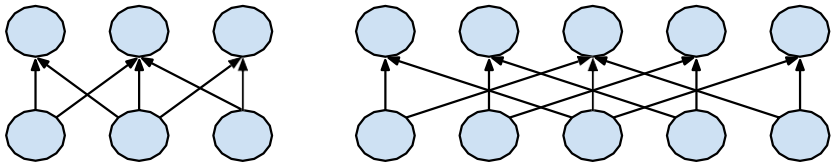


图2：扩张卷积和非扩张卷积的区别。左边是最小卷积网络的示意图，补丁大小为s=3，步长为1，而右边是扩张卷积，s=3，步长为1，扩张因子d=2。

在每个区块的输入处有20%的特征剔除（Hinton等人，2012b）。总的来说、

$$f_{emb}(x) = (B \circ B \circ B)(D_t(x, p))$$
$$B(x) = x + (L_{64} \circ - - - \circ L_2 \circ L_1)(D_f(x, p))L_d$$
$$(x) = x + \tanh(C_d(x))\ \sigma(C'(x))$$
$$C_d(x)_i = b + \sum_{j=1} w_j x_{i-d\ (j-)\ s/2\P)}$$

其中，$D_t(x, p)$将每个令牌嵌入设置为零，概率为p，$D_f(x, p)$将每个在个体特征的概率为零，C_d和C'是具有不同权重的卷积。和扩张因子d，而s = 3是卷积的补丁大小。这里x, B, L_d, C_d都是矢量序列，或2-D整体。我们的实验使用256和640维的向量。

4.5 递归神经网络

我们的第三个模型类别是递归神经网络（Socher等人，2011），它由一阶逻辑公式的解析树构建。神经网络的拓扑结构反映了解析树，并使用TensorFlow中的收集操作进行快速连接。

为了简化树，所有的函数应用都被夤缘化，从而得到以下类型的层：

- 应用节点，应用一个正好有两个孩子的函数。
- 或计算恰好两个孩子的不连接的嵌入的节点。
- 和计算恰好两个孩子的连接的嵌入的节点。这只用于嵌入被否定的猜想，因为证明条款不包含连词。
- 而不是计算单个子节点的否定的嵌入的节点。

每种类型的层的权重是在同一棵树上共享的。这意味着对于同一个公式，层的权重是共同学习的，因为它们往往包含同一类型的节点的多个实例。

在树的叶子上，我们有常数，可以代表不同算数的函数。这些符号有其相关的嵌入，这些嵌入是随机初始化的，与网络的其他部分一起学习。

这些内部节点中的每一个都由具有ReLU激活的全连接层或树状LSTM网络表示（Tai等人

，2015）。他们汇总固定大小的子节点，并输出一个具有预设输出嵌入大小长度的向量（或者对于树形LSTM来说，有两个向量

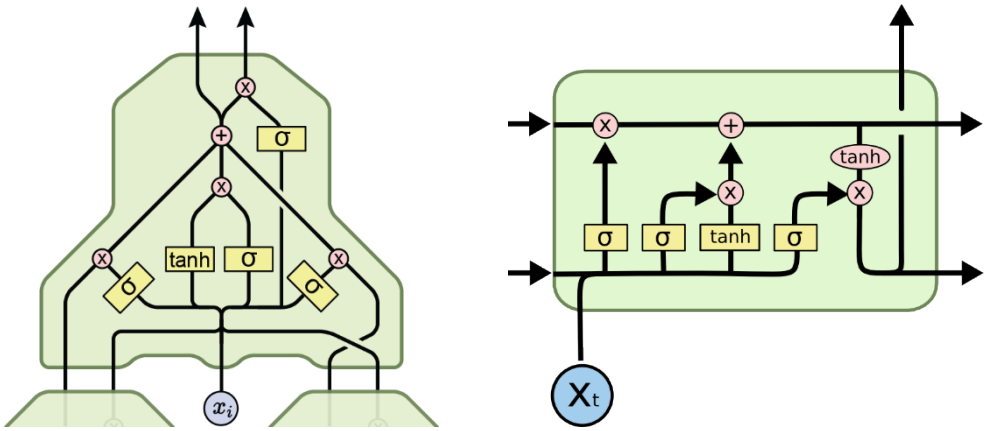


图3：一个有两个孩子的节点的树状LSTM块（左）与传统的序列LSTM（右）相比。对于内部节点，x_i 在第一层是空的，对于多层树状LSTM来说，等于下面一层的输出。图表由Chris Olah（2015）提供。

LSTMs）。我们的树状LSTM对每个输入使用单独的遗忘增益，并包括非对角线遗忘门项（详见图3和Tai等人（2015））。

该网络为这三个（对于被否定的共轭物来说是四个）转换学习独立的权重矩阵。虽然计算证明步骤句子的嵌入和嵌入的方法相同，但它们的权重是不共享的：我们最终总共有七组权重需要学习。我们测试了常见的256和512的嵌入大小。更大的嵌入尺寸产生了更准确的模型。

5 实验结果

如第4节所述，在对深度神经网络进行训练后，我们用三种方法对其进行评估，每一种方法都比上一种方法更有说服力，而且计算成本更高。

第一个指标在第5.1节中提出，检查训练过的模型是否能够准确地预测一个条款是否在最终证明中使用。这个准确性测试是在训练数据中的一组保留的证明任务上进行的。

接下来，在第5.2节中，我们在Kaliszyk & Urban (2015a)的9,159个FOL证明任务的相同保留集上运行E验证器，使用训练好的网络来帮助指导条款选择。这些定理都至少有一个使用经典搜索启发式的ATP证明，而96.6%的
这些定理可以用E验证器内置的自动启发式方法之一来证明，所以有
在这个数据集上几乎没有改进的余地。然而，它允许我们进行理智的检查，即神经网络在

添加到自动启发式中时不会弊大于利。

最后的测试是最有趣的，但计算成本很高：这些深度网络引导的选择启发式方法是否允许我们证明那些还没有任何ATP证明的Mizar语句？为此，我们使用了在Kaliszyk & Urban（2015a）中没有找到证明的25361条Mizar定理的语料库。因为这是计算成本最高的任务，所以我们只使用在前面的任务中表现最好的网络来运行它。在第5.3节，我们提出在这些模型中，我们可以找到7.36%的 "*困难陈述*"的证明。

模型	嵌入尺寸	对50-50%分割的准确度
树状RNN-256×2	256	77.5%
树状RNN-512×1	256	78.1%
树-LSTM-256×2	256	77.0%
树-LSTM-256×3	256	77.0%
树-LSTM-512×2	256	77.9%
CNN-1024×3	256	80.3%
*CNN-1024×3	256	78.7%
CNN-1024×3	512	79.7%
CNN-1024×3	1024	79.8%
波网-256×3×7	256	79.9%
*WaveNet-256×3×7	256	79.9%
波网-1024×3×7	1024	81.0%
波网-640×3×7(20%)	640	**81.**5%
*WaveNet-640×3×7(20%)	640	79.9%

表1：预测一个处理过的句子最终是否需要证明的准确性。准确率是用各种递归深度神经网络模型对正反两方面的处理过的句子实例进行50-50%的测量。带星号（*）的模型是在一个数据集上训练的，该数据集还包括一个未处理过的句子样本作为负面例子。为了便于与其他模型进行直接比较，我们使用了相同的评估数据集，但这对标有（*）的例子略有偏颇。

5.1 准确度评估

表1显示了对猜想的保留集的已用和未用条款进行50-50%分割的准确性。这里*CNN*-N×*L*是一个卷积网，有*L*层，尺寸为*N*，而Trie-*Type*-N×*L*是一个树状RNN或LSTM，在输入树的每个节点有*L*层，尺寸为*N*。*WaveNet*-N ×B×*L*有*B*个块，有*L*层，维度为*N*。我们包括(*D%*)，表示在训练过程中使用D%的辍学率作为正则器。

有辍学现象的WaveNet 640的准确率最高，为81.5%，但许多CNN和WaveNet模型的表现明显优于其他模型。请注意，对一组例子进行模型评估所需的时间差别很大。鉴于我们限制的是整体运行时间而不是进行评估的数量，质量较高但速度较慢的模型在系统中的表现可能比稍差但速度快得多的模型更好。然而，有了专门的神经网络评估硬件，我们可以预期，与运行时间相比，模型的预测质量会越来越重要。

5.2 对具有现有ATP证明的声明进行实验

条款选择启发式是E验证器最重要的部分之一，因为它是证明搜索的主要动力。有了一个好的启发式，可以在相对较少的搜索步骤中找到一个证明。在本节中，我们使用第5.1节中准确度最高的模型作为E中的条款选择启发式。

现在给每个未处理的子句分配一个分数，即$p(\text{useful}|c,nc)$的训练值。换句话说，一个条款的概率是

[2] 由于其巨大的内存占用，我们无法在E验证器中对WaveNet-1024进行实验。

c，在最后的证明中使用，给定的否定猜想集是nc。E验证人使用这个分数对未处理的条款集进行排名，然后在每一步处理具有最佳排名的条款。[3] 由于每个条款被处理时都会产生许多新的未处理的条款，这个选择顺序是至关重要的。一个好的条款选择启发式方法可以在经过少量的搜索步骤后找到一个证明，与经过数年的计算时间后才找到一个证明之间产生区别。

虽然我们的最终目标是开发一个强大的启发式，足以证明那些还没有ATP证明的具有挑战性的语句，但在这个数据集上做实验的计算成本非常高。相反，我们保留了一个由9159条语句组成的保留集，我们没有将其作为训练数据使用。这些语句已经有了现有的ATP证明，而且计算量往往不大，所以我们可以用这个保留集来进行更多的实验，并直接与现有的自动策略产生的证明进行比较，正如附录中所讨论的，我们选择了这个数据集上表现最好的策略。

在这一节中，我们发现，即使用于推理，深度神经网络的计算成本也非常高，在与现有启发式方法结合使用时最为有效。我们还使用憋屈集来研究第4节中提出的模型架构中哪些是最有效的。

5.2.1 指导性设计

在我们的实验中，我们发现，当我们的计算量大的神经网络与现有的快速启发式方法配合使用时，它们的表现要好得多。在本节中，我们介绍了使用神经网络来指导现有启发式方法的不同方法。

1. 自动启发式是E中的标准方法（见附录）。这种方法已经是几个加权函数和调整参数的混合体，但不包括任何来自深度神经网络的推理。
2. 纯粹的神经网络启发式只使用训练好的神经网络对条款进行评分。这种方法很慢，因为它必须用昂贵的神经网络评估所有条款。它也不能像Auto函数那样利用基于不同启发式的排名，而且我们观察到它本身的表现并不理想。
3. 一种混合方法在基于深度网络的指导和快速自动启发式之间交替进行。这是用额外的神经网络建议的条款来增强标准的条款选择方法。虽然这个过程由于神经网络评估的缓慢，仍然可以做相对较少的证明搜索步骤，但仍然必须对所有未处理的条款进行评估，以获得一个等级排序。
4. 一种切换的方法，在编译的第一阶段使用深度网络引导（纯粹的或混合的）。当时间资源开始耗尽时，我们切换到自动，即传统的完全搜索阶段，它可以处理大量的条款，而没有深度网络引导的开销。

图4左侧显示了这些方法在所有情况下使用简单的CNN进行证明指导的直接比较。毫不奇怪，纯CNN本身的表现并不好；然而，当我们允许E验证人在CNN启发式和自动启发式之间交替使用时，混合方法在处理条款的下限方面优于自动和CNN启发式。尽管如此，由于混合方法拥有深度网络评估的所有开销，我们看到这种混合方法在处理了大约7500个条款后，由于缺乏资源而陷入了困境。

[3] E验证器使用最低即最好的排序，所以在实施中我们使用-*p*(有用|c, *nc*)。

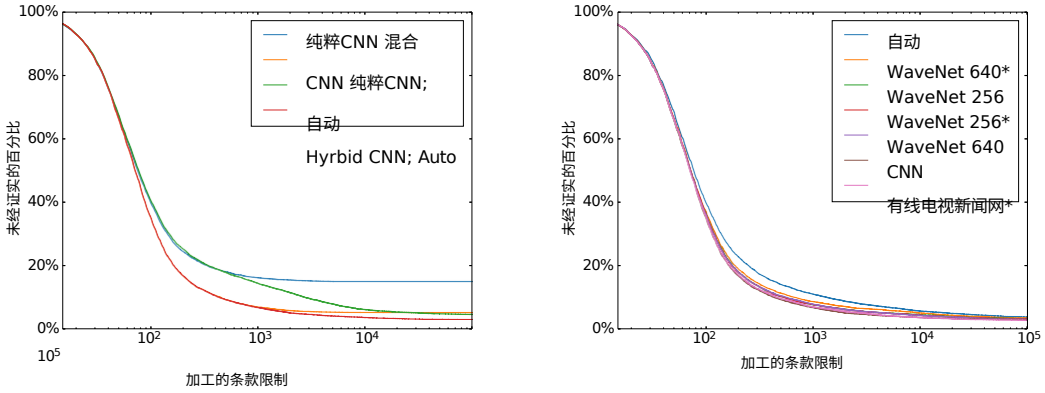


图4：使用不同的选择启发式在不同的处理子句限制下不成功的证明的百分比。在左图中，我们自始至终使用同一个网络（CNN-1024x3，有256个嵌入），但显示了与自动启发式的各种交互作用的效果。在右图中，我们使用混合、两阶段指导，但显示了不同神经网络的效果。更详细的数值显示在表2中。

我们探索的指导方法中最好的是切换方法，它使用自动启发式来避免在处理的条款数量相对较少时耗尽资源。这背后的直觉是，我们预计标准启发式方法太弱，无法避免组合爆炸，而神经网络引导在目前的硬件上仍然太慢。如果我们单独运行一个引导式证明搜索一段时间，它最终可能会选择证明的所有基本条款，但由于其速度较慢，它可能无法接近证明。我们的 "切换 "方法为网络引导方法的这一缺陷提供了补救办法。

在本文的其余部分，所有的证明指导方法都采用了两阶段的 "切换 "方法，首先运行混合网络指导阶段20分钟，然后再运行基于启发式方法的标准阶段10分钟。

5.2.2 模型在简易报表上的表现比较

在这一节中，我们介绍了四个WaveNet和两个CNN模型的性能，用于指导E验证器中的条款选择。这些是表1中准确度最高的模型，它们仍然足够小，可以装入内存。我们还包括所有三个在包括未处理的句子作为负面例子的数据集上训练的模型，这些模型用（*）表示。所有的实验都使用了1.9.1pre014版本的E定理验证器（Schulz, 2013），并进行了一些修改，以便能够与训练好的深度神经网络整合，作为句子选择启发式。

在图4右侧和表2中，我们展示了这些模型在不同处理条款限制下的性能。所有受益于深度神经网络指导的启发式方法都超过了自动基线，无论在找到证明之前处理的条款数量如何。两种CNN（嵌入大小为256）都做得很好，但我们注意到，用未处理的条款作为负

面例子训练的CNN*在处理条款的上限较高时开始做得更好，这可能更好地代表了在已经处理了几千个条款后被评估的条款。有点令人惊讶的是，在预测条款是否会被用于最终证明方面更准确的WaveNet模型（嵌入大小为256和640），在证明指导任务中表现得并不理想。这可能是因为WaveNet模型

模型	准确度	*PC*≤1,000	*PC*≤10,000	PC≤100,000	PC<∞
自动	不适用	89.0%	94.3%	96.2%	96.6%
*WaveNet 640	79.9%	91.4%	95.0%	96.5%	96.6%
波网256	79.9%	92.3%	95.5%	96.8%	96.8%
*WaveNet 256	79.9%	92.2%	95.7%	96.8%	96.8%
有线电视新闻网	80.3%	**93.**3%	**96.**4%	97.0%	97.1%
波网640	**81.**5%	92.2%	95.7%	97.0%	97.2%
*CNN	78.7%	93.0%	**96.**4%	**97.**2%	**97.**3%

表2：在各种模型结构下，在 "*简单陈述*"中证明的定理百分比。为方便起见，训练精度与表1重复。PC≤N列中的数值表示需要少于*N个*处理的声明的百分比。
句(PC)。最右边一栏 （PC<∞） 是在30分钟内被证明的语句的百分比。
分钟，内存限制为16G，对处理的条款没有限制。

在同样的资源条件下，CNN的资源消耗要大得多，因此不能像CNN那样评估许多条款。

虽然这些结果表明，来自深度神经网络的指导可以为传统的搜索启发式方法提供一些帮助，但仅自动启发式方法在这个数据集上的表现已经非常好了，所以改进是很小的。真正的测试是在 "*硬陈述*"的集合上，这些*陈述*以前没有ATP生成的证明。

5.3 硬性声明的实验

这里我们描述了我们在Mizar语料库中的25,361条定理上的结果，Kaliszyk和Urban （2015a）考虑的任何证明者、策略和前提选择方法都没有找到这些定理的证明。在本文的其余部分，我们将称这些为 "*硬陈述*"。

尽管所有的硬性语句都有人类的证明，但这个子集既不能被定理检验器E（Schulz，2013）证明，也不能被Vampire（Kovács & Voronkov，2013）在15分钟的超时限制和默认设置 （分别为自动和级联启发式）下使用从人类给出的证明中得到的前提证明。还要注意的是，传递给定理验证器的前提构成了对必要前提集的非常粗略的过度近似。通常情况下，大约10到20个依赖关系就足以证明定理，但后向包络中的依赖关系的数量可以达到数千。

网络引导下的证明搜索的主要瓶颈是对下一步要处理的条款排名的评估时间。鉴于我们没有为此使用特殊的硬件，我们的证明搜索完全被深度网络评估所支配。

如果开始时前提的数量非常多，由于每一步评价的条款数量较多，对深度网络引导的证明搜索的打击就会大得多。这就促使我们使用额外的前提选择作为一个重要的附加组件

。

尽管如此，我们首先提出了一个没有前提选择的基线，并表明相对快速的前提选择阶段对于有无深度网络指导的硬定理的良好表现至关重要。在第二小节中，我们打开前提选择，将各种证明指导模型与基线自动启发式进行比较。

	无前提选择	有前提的选择
无指导的	145	458
有指导性的（混合）。	137	383

表3：用前提选择和（无开关）证明指导的不同组合证明的硬定理数量。请注意，即使我们的证明指导是部分的，它产生的结果仍然比没有深度网络指导的变体差。这是由于深度网络评估的缓慢性造成的。此表的唯一目的是强调硬性语句的前提选择的重要性。在其他实验中，我们专注于两阶段的 "切换 "方法，该方法以顺序的方式结合了引导和非引导的搜索，并且在没有切换的情况下比非引导的搜索和混合引导都要好。

5.3.1 选择前提的重要性

在这里，我们首先比较了四种不同的硬性声明的方法。这些方法包括有指导的和无指导的证明搜索，有前提选择和无前提选择。

对于前提选择，我们使用DeepMath论文（Alemi等人，2016）中的字符级别模型。该模型与我们的卷积证明指导模型非常相似，但在前提-结论对上进行了训练，该训练集是从56%的ATP证明的语句中随机选择的。一个主要的区别是，我们的证明指导卷积网络是词级的，为每个标记学习了一个嵌入，而前提选择网络将字符序列作为输入。这限制了质量，因为DeepMath论文表明，使用具有定义嵌入的词级模型可以获得最佳结果。为了简单起见，我们选择了质量较低的字符级前提选择模型。

在前提选择之后，我们运行四次证明尝试：首先按模型得分对前提进行排名，并选择前32、64、128和256个不同的前提（只要不超过原始前提数量）。我们使用选定的前提集运行E验证器，作为
只要它没有找到一个证明。当对任何前提的子集找到证明时，我们就停止搜索证明。

实验结果在表3中给出。我们给出的实验证据表明，转换的方法优于无指导的证明搜索。我们测试了两种不同的前提选择模型，以感受到结果的可变性和互补性。事实证明，不同的前提选择模型，即使具有相同的结构，也会引入很多变化，从而导致不同数量的定理被证明，但也会导致选择上的互补，并通过增加起始前提集的多样性来帮助证明搜索过程。

5.3.2 模型在硬报表上的表现比较

所有的证明指导方法都采用了两阶段的方法，首先运行混合网络指导阶段20分钟，然后是

基于标准启发式方法的阶段10分钟。

在这里，我们尝试了两种不同的前提选择模型 "DeepMath 1 "和 "DeepMath 2"，使用与（Alemi等人，2016）中描述的最佳字符级模型相同的字符级卷积架构。这样一来，我们可以评估结果在前提选择策略方面的稳定性。我们可以看到，尽管这两个模型的训练方式相同，质量也相当，但它们导致了明显不同的定理证明集，但

模型	DeepMath 1	DeepMath 2	1和2的联合
自动	578	581	674
*WaveNet 640	644	612	767
*WaveNet 256	692	712	864
波网640	629	685	997
*CNN	905	812	1,057
有线电视新闻网	839	935	1,101
共计（唯一的）	1,451	1,458	1,712

表4：用前提选择（DeepMath 1和2）和子句选择指导的不同组合证明的25361条硬定理中的定理数量。最后一列显示了在给定行中用任一前提选择步骤方法所证明的定理的联合。本表中的方法所证明的所有定理的联合体的大小为1,712（6.8%）。由深度网络引导的方法所证明的定理数量为1,665（6.6%）。

使用两种模型证明的定理数量非常相似，而使用较简单的CNN的 "切换 "策略在前提选择和证明指导方面的表现都很好。

实验结果在表4中给出。CNN模型使用简单的三层卷积网络对未处理的条款进行排名，而WaveNet模型则使用WaveNet架构，其评估速度明显较慢，但在保留集上产生了更好的代理元。请注意，基于简单卷积网络的方法证明了
比无指导的方法多出86%的定理（4.34%对2.33%的硬性陈述）。本文中通过任何方式（包括使用非开关神经引导）证明的语句总数为1,866条，占所有硬语句的7.36%。

6 总结

我们已经证明了通过用深度神经网络的排名来增强给定的条款选择方法来指导一阶逻辑证明搜索的可行性。通过适当设计的神经指导和手工制作的搜索策略的混合物，我们得到了一阶逻辑证明者性能的显著改善，特别是对于那些更难的、需要更深入搜索的定理。

由于神经网络评估的缓慢性，我们的方法只有在两阶段的方法中利用它才能导致成功证明的数量增加，其中深度网络引导阶段之后是更快的组合搜索，使用现有的手工制作的条款选择策略快速选择给定的条款。

此外，我们还发现，深度网络的预测精度和速度对提高证明力都很重要。例如，我们的WaveNet模型比简单的卷积网络在保留集上产生了更高的准确性，但是由于其计算成本高得多，在E验证器中作为条款评分器使用时，在同样的时间限制下，它仍然比便宜

但质量较低的卷积网络证明了更少的定理。

目前，我们的方法在计算上是昂贵的，我们允许每个证明30分钟，而以前的工作是15分钟。然而，除非与神经证明指导相结合，否则这个额外的时间对现有的启发式方法没有明显帮助（见附录）。这表明

神经引导代表了在限制搜索空间方面的重大改进。此外，我们预计，用于深度学习的专门硬件的进入将减轻大部分的计算开销。这将使性能更高的模型在未来发挥其优势，并获得巨大的效率和质量提升。

我们的方法只是将深度学习应用于指导组合搜索过程的第一步，可以说，与纯粹的句法输入格式一起工作，得出的特征集还不够强大，无法创建数学内容的语义相关表示。除了改善定理证明，我们的方法还有一个令人兴奋的潜力，即为研究公式在一组逻辑转换下的行为的系统产生更高质量的训练数据。这可以使学习表示公式的方式考虑到数学内容的语义而不仅仅是句法属性，并可以在证明搜索期间根据其行为做出决定。

参考文献

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Good- fellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore、 Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng.Tensor- Flow: Large-scale machine learning on heterogeneous systems, 2015.软件可从tensorflow.org获得。

Jesse Alama, Daniel Kühlwein, and Josef Urban.普通数学中的自动和人工证明： An Initial Comparison.In Nikolaj Bjørner and Andrei Voronkov (eds.), *LPAR*, volume 7180 of *LNCS*, pp.37-45.Springer, 2012.ISBN 978-3-642-28716-9。

Alexander A Alemi, Francois Chollet, Geoffrey Irving, Niklas Een, Christian Szegedy, and Josef Urban.用于前提选择的深层数学-深层序列模型。 In *Advances in Neural Information Processing Systems*, pp.2235-2243, 2016.

Serge Autexier和Dieter Hutter.大型理论中的结构形成。 In Manfred Kerber, Jacques Carette, Cezary Kaliszyk, Florian Rabe, and Volker Sorge (eds.), *Intelligent Com- puter Mathematics - International Conference, CICM 2015, Washington, DC, USA, July 13-17, 2015, Proceedings*, volume 9150 of *LNCS*, pp.155-170。 Springer, 2015.

Grzegorz Bancerek, Czesław Byliński, Adam Grabowski, Artur Korniłowicz, Roman Ma- tuszewski, Adam Naumowicz, Karol Pąk, and Josef Urban.Mizar： 最先进的技术和超越。 In Manfred Kerber, Jacques Carette, Cezary Kaliszyk, Florian Rabe, and Volker Sorge (eds.), *Intelligent Computer Mathematics - International Conference, CICM 2015, Washington, DC, USA, July 13-17, 2015, Proceedings*, volume 9150 of *LNCS*, pp.261-279.Springer, 2015. doi: 10.1007/978-3-319-20615-8.URL http://dx.doi.org/10.1007/

978-3-319-20615-8.

Jasmin Christian Blanchette, Cezary Kaliszyk, Lawrence C. Paulson, and Josef Urban.Hammering towards QED.*J. Formalized Reasoning*, 9(1):101-148, 2016. doi: 10.6092/issn.1972-5787/4593.URL http://dx.doi.org/10.6092/issn.1972-5787/4593.

克里斯-奥拉。了解LSTM网络，2015年。URL https://colah.github.io/posts/ 2015-08-Understanding-LSTMs.

Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio.gated feedback recurrent neural networks. *arXiv preprint arXiv:1502.02367*, 2015.

Jörg Denzinger, Matthias Fuchs, Christoph Goller, and Stephan Schulz.从以前的证明经验中学习。技术报告AR99-4，慕尼黑工业大学信息学院，1999。

Michael Färber和Chad E. Brown。 Satallax的内部指导。 In Nicola Olivetti and Ashish Tiwari (eds.), *International Joint Conference on Automated Reasoning (IJCAR 2016)*, volume 9706 of *LNCS*, pp.349-361.Springer, 2016. doi: 10.1007/978-3-319-40229-1.

Christoph Goller 和 Andreas Kuchler.通过结构的反向传播学习依赖任务的分布式表征。In *Neural Networks, 1996., IEEE International Con- ference on*, volume 1, pp.347-352.IEEE, 1996.

Adam Grabowski, Artur Korniłowicz, and Adam Naumowicz.Mizar 简 述 .*J. Formalized Reasoning*, 3(2):153-245, 2010.

Adam Grabowski, Artur Korniłowicz, and Adam Naumowicz.米扎尔四十年--前言。*J.Autom.Reasoning*, 55(3):191-198, 2015. doi: 10.1007/s10817-015-9345-1. URL http://dx.doi.org/10.1007/s10817-015-9345-1.

Thomas C. Hales, Mark Adams, Gertrud Bauer, Dat Tat Dang, John Harrison, Truong Le Hoang, Cezary Kaliszyk, Victor Magron, Sean McLaughlin, Thang Tat Nguyen, Truong Quang Nguyen, Tobias Nipkow, Steven Obua, Joseph Pleso, Jason Rute, Alexey Solovyev, An Hoai Thi Ta, Trung Nam Tran, Diep Thi Trieu, Josef Urban, Ky Khac Vu, and Roland Zumkeller. 开 普 勒 猜 想 的 正 式 证 明 。 *CoRR*, abs/1501.02155, 2015.URL http://arxiv.org/abs/1501.02155。

John Harrison, Josef Urban, and Freek Wiedijk. 交互式定理证明的历史。 In Jörg H. Siekmann (ed.), *Computational Logic*, volume 9 of the *Handbook of the History of Logic*, pp.135 - 214.North-Holland, 2014. doi: http://dx.doi.org/10.1016/ B978-0-444-51624-4.50004-6.URL http://www.sciencedirect.com/science/article/ pii/B9780444516244500046。

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun.Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, and Brian Kingsbury.语音识别中用于声学建模的深度神经网络：四个研究小组的共同观点。*IEEE信号处理杂志*，29（6）：82-97，2012a。

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 通 过 防 止 特 征 检 测 器 的 共 同 适 应 来 改 进 神 经 网 络 。 *arXiv 预印本 arXiv:1207.0580*, 2012b.

Cezary Kaliszyk 和Josef Urban 。 MizAR 40 for Mizar 40.*J. Autom.Reasoning*, 55(3)： 245-256, 2015a. doi: 10.1007/s10817-015-9330-8.URL http://dx.doi.org/10.1007/ s10817-015-9330-8.

Cezary Kaliszyk and Josef Urban. FEMaLeCoP: Fairly efficient machine learning connec- tion prover.In Martin Davis, Ansgar Fehnker, Annabelle McIver, and Andrei Voronkov (eds.), *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015*, volume 9450 of *LNCS*, pp.88-96.Springer, 2015b.ISBN 978- 3-662-48898-0. doi: 10.1007/978-3-662-48899-7_7.URL http://dx.doi.org/10.1007/978-3-662-48899-7_7.

Yoon Kim.卷积神经网络用于句子分类。*arXiv预印本arXiv:1408.5882*，2014。

Diederik Kingma 和 Jimmy Ba.Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Gerwin Klein, June Andronick, Kevin Elphinstone, Gernot Heiser, David Cock, Philip Derrin, Dhammika Elkaduwe, Kai Engelhardt, Rafal Kolanski, Michael Norrish, Thomas Sewell, Harvey Tuch, and Simon Winwood. seL4：操作系统内核的形式验证。*Commun.acm*，53（6）：107-115，2010。

Laura Kovács 和 Andrei Voronkov.一阶定理证明和Vampire.In Natasha Sharygina and Helmut Veith (eds.), *CAV*, volume 8044 of *LNCS*, pp.1-35.Springer, 2013.ISBN 978-3-642-39798-1.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton.用深度卷积神经网络进行图像分类。In *Advances in neural information processing systems*, pp.1097-1105, 2012.

Daniel Kühlwein, Twan van Laarhoven, Evgeni Tsivtsivadze, Josef Urban, and Tom Heskes.大理论数学的前提选择技术的概述和评估。In Bernhard Gramlich, Dale Miller, and Uli Sattler (eds.), *IJCAR*, volume 7364 of *LNCS*, pp.Springer, 2012.ISBN 978-3-642-31364-6。

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner.基于梯度的学习应用于文档识别。*IEEE论文集*，86（11）：2278-2324，1998。

诺曼-梅吉尔*Metamath：A Computer Language for Pure Mathematics*.Lulu Press, Mor- risville, North Carolina, 2007.Isbn 978-1-4116-3724-5.URL http://us.metamath.org/downloads/metamath.pdf。

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu.Wavenet：原始音频的生成模型。*arXiv预印本arXiv:1609.03499*，2016。

Tim Rocktäschel 和 Sebastian Riedel.用神经定理证明器学习知识库推理。*第五届自动知识库构建研讨会（AKBC）*，2016年。

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočisky`, and Phil

Blun- som.*arXiv preprint arXiv:1509.06664*, 2015.

Stephan Schulz.*学习搜索控制知识的等价推理*，第230卷。
DISKI. Infix Akademische Verlagsgesellschaft, 2000. ISBN 978-3-89838-230-4.

斯蒂芬-舒尔茨系统描述：E 1.8.In Kenneth L. McMillan, Aart Middeldorp, and Andrei Voronkov (eds.), *Logic for Programming, Artificial Intelligence, and Reasoning - 19th International Conference, LPAR-19*, volume 8312 of *LNCS*, pp.Springer, 2013.ISBN 978-3-642-45220-8. doi: 10.1007/978-3-642-45221-5_49.url http://dx.doi.org/ 10.1007/978-3-642-45221-5_49。

Stephan Schulz.*E 1.9.1用户手册（初步版本）*，2016。URL http://www.lehre. dhbw-stuttgart.de/~sschulz/WORK/E_DOWNLOAD/V_1.9.1/eprover.pdf。

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanc- tot, et al. Mastering the game of go with deep neural networks and tree search.*自然》*，529（7587）：484-489，2016。

Richard Socher, Cliff C Lin, Chris Manning, and Andrew Y Ng.用递归神经网络解析自然场景和自然语言。在*第28届国际机器学习会议（ICML-11）论文集*，第129-136页，2011年。

Geoff Sutcliffe.TPTP问题库和相关基础设施*J. Autom.Reasoning*, 43(4):337-362, 2009. doi: 10.1007/s10817-009-9143-8.URL http://dx.doi.org/10.1007/ s10817-009-9143-8.

Christian Suttner and Wolfgang Ertel.自动获取搜索指导启发式方法。在*国际自动演绎会议*，第470-484页。Springer, 1990.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich.用卷积法深入研究。在*IEEE计算机视觉和模式识别会议论文集中*、pp.1-9, 2015.

Kai Sheng Tai, Richard Socher, and Christopher D Manning.*arXiv preprint arXiv:1503.00075*, 2015.

Josef Urban.MPTP 0.2: 设计、实现和初始实验*J. Autom.Reason-ing*, 37(1-2):21-43, 2006. doi: 10.1007/s10817-006-9032-3.URL http://dx.doi.org/10.1007/s10817-006-9032-3.

Josef Urban, Jiří Vyskočil, and Petr Štěpánek.MaLeCoP：机器学习连接验证器。In Kai Brünnler and George Metcalfe (eds.), *TABLEAUX*, volume 6793 of *LNCS*, pp.263-277.Springer, 2011.ISBN 978-3-642-22118-7.

Markus Wenzel.Isar - 可读形式证明文件的通用解释方法。In Yves Bertot, Gilles Dowek, André Hirschowitz, Christine Paulin-Mohring, and Laurent Théry (eds.), *Theorem Proving in Higher Order Logics, 12th International Conference, TPHOLs'99*, volume 1690 of *LNCS*, pp.167-184。Springer, 1999.

Daniel Whalen.Holophrasm: a neural automated theorem prover for higher-order logic. *arXiv*

深度网络引导的证明搜索
S.Loos, G. Irving, C. Szegedy, and C. Kaliszyk
preprint arXiv: 1608.02644, 2016.

附录

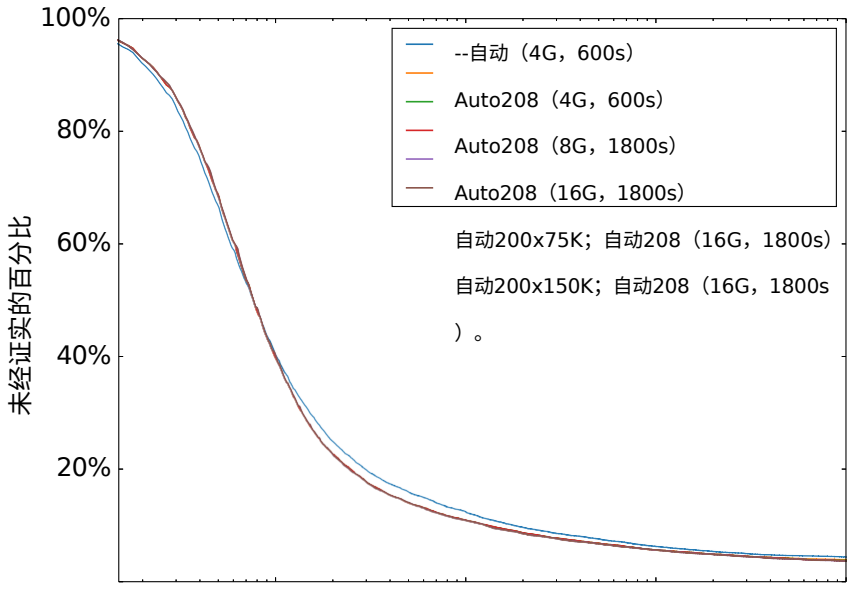
选择自动基线

在这里，我们比较了几种内置于E验证器的混合选择启发式方法。在本文中，我们使用在我们的保留集上表现最好的混合启发式方法作为基线。

首先，我们用--自动标志在我们的保留集上运行E验证器。当这个标志启用时，E验证器根据猜想的特征动态地选择一个（混合）选择启发式。它还会动态地选择术语排序和字面选择策略（Schulz，2016）。

我们发现，在用--auto标志生成的52.5%的证明中，使用了Auto208启发式。当我们在拥有4G内存和600秒超时的完整数据集上只使用Auto208启发式时，它比--自动标志证明了更多定理，如图5所示。

在第5.2.1节中，我们介绍了一种切换的方法，它首先运行混合启发式，然后随着定理检验器资源的耗尽，切换到Auto208选择启发式。为了确保我们获得的任何好处不是由于切换本身，我们还在此试验在两个内置启发式方法之间进行切换。在我们的保留集上，--auto标志在5.9%的证明中使用Auto200启发式，使其成为第二大使用的启发式。在表5中，我们显示
从Auto200切换到Auto208启发式对最终的影响非常小。
结果。"Auto200x150K; Auto208 "首先运行Auto200，处理多达15万个子句，然后运行Auto208处理剩余的子句（类似于处理7.5万个）。我们选择15万个已处理的子句，因为这是使用Auto208的不成功证明的子句的中位数。



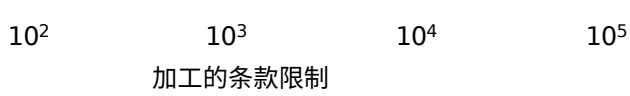


图5：使用E型验证器选择启发式方法在不同的处理子句限制下不成功的证明百分比。表5列出了没有处理条款限制的最终值。

启发式选择	记忆	超时	已探明的百分比
--自动	4G	600s	95.72%
自动208	4G	600s	96.12%
自动208	8G	1800s	96.52%
自动200x150K；自动208	16G	1800s	96.60%
自动200x75K；自动208	16G	1800s	96.62%
自动208	16G	1800s	96.64%

表5：比较自动基线对简单定理的影响。

这里包括了Auto208和Auto200启发式的全名和混合排序，对于它们，我们使用术语排序KBO6。在本文中，我们将Auto208作为我们的Auto基线进行比较，因为它是这些实验中表现最好的启发式方法。

```
"Auto208"
G_E___208_c18_f1_se_cs_sp_ps_s0y :=
1*ConjectureRelativeSymbolWeight(SimulateSOS,0.5,100,100,100,1.5,1.5,1),
4*ConjectureRelativeSymbolWeight(ConstPrio,0.1,100,100,100,1.5,1.5），1*FIFOWeight
（PreferProcessed），1*ConjectureRelativeSymbolWeight（
PreferNonGoals,0.5,100,100,100,1.5,1.5,1），4*Refinedweight（
SimulateSOS,3,2,2,1.5,2）。
```

```
"Auto200"
G_E___200_c45_f1_ae_cs_sp_pi_s0y :=
1*ConjectureRelativeSymbolWeight(SimulateSOS,0.5,100,100,100,1.5,1.5,1),
6*ConjectureRelativeSymbolWeight(ConstPrio,0.1,100,100,100,1.5,1.5），2*FIFOWeight
（PreferProcessed），1*ConjectureRelativeSymbolWeight（
PreferNonGoals,0.5,100,100,100,1.5,1.5,1），8*Refinedweight（
SimulateSOS,1,1,2,1.5,2）。
```