软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(8):2839−2850 [doi: 10.13328/j.cnki.jos.006601]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于消息传递关系网络的布尔可满足性预测*

包冬庆[1], 葛 宁[1,3], 翟树茂[1], 张 莉[1,2,3]

[1](北京航空航天大学 软件学院, 北京 100191)

[2](北京航空航天大学 计算机学院, 北京 100191)

[3](软件开发环境国家重点实验室(北京航空航天大学), 北京 100191)

通信作者: 张莉, E-mail: lily@buaa.edu.cn

摘 要: 布尔可满足性求解能够验证的问题规模通常受限, 因此, 如何高精度地预测其可满足性既是重要的研究问题, 也是一项具有挑战性的工作. 相关研究工作一般使用由文字节点和子句节点组成的图来表示布尔可满足性问题的结构, 但是这种表征方法缺少了变量、子句之间的重要关系信息. 在所提方法中, 通过将原始布尔可满足性问题实例表征为多关系异构图的方式来表达变量和句子之间的关系, 并设计使用消息传递关系网络模型来捕获实例的关系信息, 提取了更多的结构特征. 结果表明: 该模型在预测精度、泛化能力和资源需求等方面均优于现有模型, 对所选数据集的平均预测精度为 81%. 该模型在小规模问题(变量数为 100)上训练, 在大规模数据集上预测的平均预测精度达到了 80.8%. 同时, 该模型对随机生成的非均匀随机问题的预测精度达到 99%, 这意味着它学习了预测可满足性的重要特征. 此外, 模型预测所花费的时间随着问题规模的增大也只是成线性增长. 总结而言, 基于关系消息传递网络提出了一个预测精度更高、泛化能力更好的布尔可满足性预测方法.

关键词: 布尔可满足性问题; 消息传递网络; 结构特征; 可满足性预测; 多关系异构图

中图法分类号: TP311

中文引用格式: 包冬庆, 葛宁, 翟树茂, 张莉. 基于消息传递关系网络的布尔可满足性预测. 软件学报, 2022, 33(8): 2839–2850. http://www.jos.org.cn/1000-9825/6601.htm

英文引用格式: Bao DQ, Ge N, Zhai SM, Zhang L. Predicting Propositional Satisfiability via Message Passing Relation Network. Ruan Jian Xue Bao/Journal of Software, 2022, 33(8): 2839–2850 (in Chinese). http://www.jos.org.cn/1000-9825/6601.htm

Predicting Propositional Satisfiability via Message Passing Relation Network

BAO Dong-Qing[1], GE Ning[1,3], ZHAI Shu-Mao[1], ZHANG Li[1,2,3]

[1](College of Software, Beihang University, Beijing 100191, China)

[2](School of Computer Science and Engineering, Beihang University, Beijing 100191, China)

[3](State Key Laboratory of Software Development Environment (Beihang University), Beijing 100191, China)

Abstract: The scale of problems that can be verified by Boolean satisfiability (SAT) solving is usually limited. Therefore, how to predict the satisfiability of SAT problems with high accuracy is an important research problem and also a challenging task. Previous works used graphs consisting of literal nodes and clause nodes to represent the structure of SAT problems. The important relation information between variables and clauses is missing. Raw SAT instances are encoded to multi-relational heterogeneous graphs and a message passing relation (MPR) network model is used to capture more structure features of an SAT instance. It is showed that the MPR network model could outperform previous work in terms of prediction accuracy, generalization ability, and resource requirement. An average prediction accuracy of 81% is achieved on all datasets. The model trained on small-scale problems (the number of variables is 100) achieves an

* 基金项目: 国家自然科学基金(61902011, 61690202); 北京航空航天大学软件开发环境国家重点实验室开放课题(SKLSDE-2021ZX-01)

本文由“形式化方法与应用”专题特约编辑陈立前副教授、孙猛教授推荐.

收稿时间: 2021-09-05; 修改时间: 2021-10-14; 采用时间: 2022-01-10; jos 在线出版时间: 2022-01-28

average prediction accuracy of 80.8% on larger-scale datasets. Prominently, this model gets 99% prediction accuracy on the randomly generated non-uniform random SAT problems, which means it has learned important features to predict satisfiability. Moreover, the running time for prediction increases linearly with the size of the problem. In conclusion, the proposed method is of higher prediction accuracy and better generalization ability based on a relational messaging network to predict propositional satisfiability.

Key words: Boolean satisfiability problem; message passing relation network; structure features; propositional satisfiability prediction; multirelational heterogeneous graph

布尔可满足性问题(SAT)用于确定是否存在满足给定布尔公式的解. 如果存在一组变量的赋值, 可以使得给定布尔公式的结果为真, 则该问题是可满足的; 否则是不可满足的. SAT 问题通常由 m 个合取子句组成的合取范式(CNF)来描述, 其中每一个子句由一组相互之间为析取关系的正文字或反文字组成. 所有的 SAT 问题都可以简化为 3-SAT 问题[1]. 一般来说, 处于相变时的 uniform random (UR)3-SAT 问题(子句数与变量数之比约等于 4.26)相比于其他问题求解起来要困难得多[2]. 实证研究表明: 在最坏的情况下, 使用完全基于回溯算法的 SAT 求解器求解 UR 3-SAT 问题, 所需时间随实例规模增加呈指数级增长[3].

SAT 求解能够验证的问题规模通常受限, 因此, 如何高精度地预测 SAT 问题的可满足性是一个重要研究问题, 也是一项具有挑战性的工作. 近年来, 一些研究工作转向了使用机器学习的方法来预测 SAT 问题的可满足性[4,5]. 一些研究者采用 SAT 问题的统计特征来加快 SAT 问题的求解速度[6−10], 并取得了不错的效果. 然而, SAT 问题的统计特征并不能完整地表达 SAT 问题的特征, 其中缺乏 SAT 问题的完整结构信息. 这在一定程度上导致模型难以很好地学习到 SAT 问题的共性特征, 降低了模型的学习能力和泛化能力. 此外, 计算 SAT 问题统计特征的过程通常十分消耗资源. 因此, 近期的研究工作开始将研究方向转换为使用 SAT 问题的结构特征来预测其可满足性[11−15].

建立模型的第 1 步是对 SAT 问题进行编码. SAT 问题很适合表征为图结构, 用图来表示 SAT 问题可以使模型提取更多的结构信息, 从而表现出良好的泛化能力. 现有的方法都将 SAT 问题编码为一个无向图, 每个文字都是一个节点, 每个子句也是一个节点, 如果文字出现在子句中, 则建立一条边[12−14]. 最后, 利用所有文字的特征来预测可满足性. 然而, 这种表征方法忽略了变量与子句之间的关系信息, 没有直接地在结构图中表现出来; 其次, 对于正反文字之间的关系, 虽然建立了边来表征两者之间有联系并进行了节点之间的信息交换, 但是并没有定义具体的关系函数, 也没有使用特定的关系向量来表征关系信息, 其采用的网络模型也没有考虑到对边上关系信息的提取. 因此, 我们认为(并将在稍后的实验中显示), 目前的表示仍然不足以完全表示 SAT 问题的结构.

一旦定义了 SAT 问题的表征方式, 下一步就是设计模型架构, 使模型学习如何解决 SAT 问题. 目前的工作使用消息传递(message passing, MP)神经网络来预测 SAT 问题的可满足性[13], 或通过使用可变矩阵神经网络来改进泛化能力[14], 并表明模型运行时间仅与 SAT 实例的规模成线性关系. 一些工作提到, 节点间的关系信息是图的重要组成部分, 但尚未提出一个模型架构来解决这个问题[16]. 因此, 关键问题是设计一种能够提取变量与子句之间的关系信息的网络模型.

本文旨在提高可满足性的预测精度, 创新性体现在 3 个方面.

- 首先, 本文使用由变量和子句节点组成的多关系异构图网络[16]来表征 SAT 问题的特征. 变量和子句节点之间的边表示变量出现在某一个子句中, 边上的信息表示变量在子句中是正文字还是反文字, 充分地反映了变量和子句之间的关系特征;
- 其次, 本文提出了一个消息传递关系(message passing relation, MPR)网络模型来提取 SAT 问题的结构特征. 在消息传递过程中, 节点和边上的信息都会被传递给相邻节点. 这意味着节点除了获取相邻节点的特征信息之外, 还可以获得相邻节点之间关系的特征信息. 因此, 采用 MPR 可以比其他模型提取更多的 SAT 问题结构特征[13,14];
- 第三, 在预测可满足性时, 本文并未使用所有的文字特征来预测可满足概率, 而是使用了所有子句的特征. 其原因是, SAT 问题是否满足取决于每个子句.

本文从预测的精度、泛化能力、资源需求和运行时间等 4 个方面对所提出的方法进行了评价. 为了方便与以往工作进行比较, 我们在评估中使用文献[14]的处于相位转移的 UR 3-SAT 数据集. 总体而言, 本文的方法在变量数为 100–600 的 SAT 问题数据集上的平均准确率达到了 81.1%, 而基线方法的最佳准确率为 79.0%; 在变量数为 600 的数据集上(求解时间为几个小时), 我们获得了 85%的准确率, 而基线方法的预测的准确率为 83.7%. 与文献[14]一样, 我们也在小规模的 SAT 问题数据集上进行训练, 然后将训练好的模型泛化到规模更大的数据集上. 我们在变量数为 100 的数据集上对 MPR 网络模型进行训练, 训练好的模型在 150–600 变量范围的 SAT 问题上内达到了 79.5%的平均预测精度, 而基线方法的平均预测精度只有 77.5%. 此外, 在 MPR 网络模型的训练阶段, 平均每个 SAT 问题只需要 100 MB GPU 显存, 而文献[13]的方法需要 500 MB GPU 显存, 文献[14]的方法需要 600 MB GPU 显存. 最后, 我们的方法在预测阶段的运行时间随问题规模的增长成线性增长趋势.

1 MPR 网络模型结构设计

预测 SAT 问题的可满足性本质上是一项图分类任务. 为了对图进行分类, 我们将 SAT 实例表征为一个多关系的异构图; 然后, 利用 MPR 网络对图中的结构特征进行提取, 并对提取的特征进行池化操作, 通过激活函数输出预测值. 对于模型的训练过程, 我们使用端到端的方法进行训练, 模型不但需要预测 SAT 问题的可满足性, 而且需要预测每个变量的赋值.

1.1 SAT实例图表征

为了将原始的包括了 m 个子句、n 个变量的 SAT 实例 X 编码为适当的向量表征, 现有工作将 X 编码为一个维度是 $m\times n\times 2$ 的子句-文字形式的张量[14]或由 m 个子句节点和 $n\times 2$ 个文字节点组成的图[13]. 事实上, 在 SAT 问题中, 文字的特征是根据变量的特征和变量与子句之间关系的特征来动态计算而来的, 而如果子句的特征确定之后, 也可以通过他们之间的文字关系反向计算出变量的特征. 因此, 本文将 SAT 实例表征为一个有向多关系异构图, 如图 1 所示, 本文将 X 表示为一个由 n 个变量节点和 m 个子句节点组成的图, 虚线表示变量在子句中为反文字, 实线表示变量在子句中为正文字, 图 1(a)变量节点 v_i 和子句节点 c_j 之间的边表示变量 i 的文字是否出现在子句 j; 图 1(b)中的边表示子句 j 是否包含变量 i; 否则, 在两个节点之间不定义边.

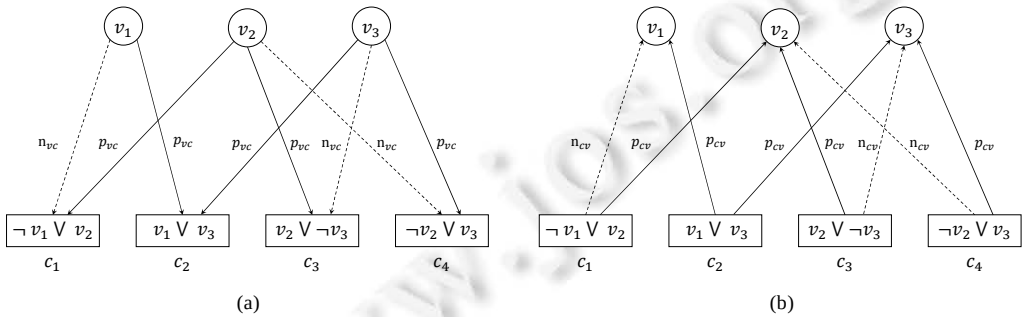


图 1 SAT 问题图表征方法

通过这种方式, 我们将原始 SAT 问题编码为一个多关系异构图, 该图由变量节点和子句节点组成和 4 种表示它们之间正或反关系的有向边组成, 可以将其形式化地描述为

$$G=\{(v_i,r_{ij},c_j)|i\in\{1\ldots n\},j\in\{1\ldots m\},r_{ij}\in R\} \tag{1}$$

其中, v_i 表示变量 i, c_j 表示子句 j, $r_{ij}\in R$ 表示 v_i 与 c_j 之间的关系. 本文中定义的关系是正反文字两种关系加上变量句子节点的两种方向的边的组合, 即 $R=\{p_{cv},p_{vc},n_{cv},n_{vc}\}$, p_{cv} 表示子句节点指向变量节点的正文字的正关系, p_{vc} 表示变量节点指向子句节点的正文字的正关系, n_{cv} 表示子句节点指向变量节点的反文字的反关系, n_{vc} 表示变量节点指向子句节点的反文字的反关系.

在我们的多关系异构图中，节点数为 $m+n$，边的数量为 k，其中，k 等于 SAT 实例中文字的数量×2.

1.2 消息传递关系

将 SAT 实例表征为图之后，下一步是设计适当的模型来提取 SAT 图的结构特征．我们设计了一个基于消息传递关系方法的 MPR 网络[16]来提取结构特征，该网络可以提取节点以及节点之间边的信息．

最初的消息传递网络[17]并不考虑节点间边的关系信息，只通过使用相邻节点的特征来更新节点的状态信息，如下所示:

$$h_i^{(l+1)}=\sigma\left(\sum\nolimits_{m\in M_i} g_m(h_i^{(l)},h_j^{(l)})\right) \tag{2}$$

其中，$h_i^{(l)}$ 是迭代 l 时节点 v_i 的状态；M_i 是 v_i 相邻节点的集合；$g_m(\cdot,\cdot)$是一个变换函数(例如一个简单的线性函数)，使用相邻节点的信息更新 v_i 的状态；$\sigma(\cdot)$是激活函数．基于这个消息传递网络，Selsam 等人提出了 MP 网络预测布尔可满足性[13]．他们将消息传递的过程分为两个步骤：首先，文字节点的信息传播到相邻的子句节点，以更新子句节点的状态；然后，子句节点的信息传播到相邻的文字节点，并更新文字节点的状态．这个过程可以表示为

$$(C_i^{(t+1)},C_{i,h}^{(t+1)})\leftarrow g_c([C_i^{(t)},\sum\nolimits_{j\in N(C)} f_c(L_j^{(t)})]) \tag{3}$$

$$(L_i^{(t+1)},L_{i,h}^{(t+1)})\leftarrow g_l([L_i^{(t)},\sum\nolimits_{j\in N(L)} f_l(C_j^{(t)})]) \tag{4}$$

其中，$g_c(\cdot)$和 $g_l(\cdot)$分别表示用户提取子句和文字特征的 LSTM 网络[18]，矩阵 $C_i^{(t)}$ 和 $C_{i,h}^{(t)}$ 是子句表示和循环隐藏状态(文字节点类似)，$f_c(\cdot)$和 $f_l(\cdot)$是提取邻居节点特征的多层感知机(multi-layer perceptrons，MLP)，$N(\cdot)$返回子句节点或者文字节点相邻节点的索引．这个网络在消息传递的过程中，只传播了节点的特征，缺失了变量与子句之间的关系．因此，我们认为，MP 网络并不完全适用于多关系异构图．

在本文的方法中，对于模型中的消息传递，主要参考了文献[16]中提出的方法，其特点是非常适合于多关系异构图，使用了如下公式进行消息传递:

$$h_i^{(l+1)}=\sigma\left(\sum\nolimits_{r\in R}\sum\nolimits_{j\in N_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)}+W_0^{(l)}h_i^{(l)}\right) \tag{5}$$

其中，N_i^r 表示节点 v_i 的相邻节点处于关系 $r\in R$，$c_{i,r}$ 是正则化系数．通过将权重矩阵 w_r 赋给不同的关系 $r\in R$，区分节点之间的关系．

图 2 所示展示了消息传递的动态过程．本文提出的消息传递关系网络也将消息传递过程分割为两个步骤：首先，子句节点接收来自其相邻变量的消息并更新其特征(如图 2(a)所示)；然后，变量节点从其相邻子句接收消息并更新其特征(如图 2(b)所示)．

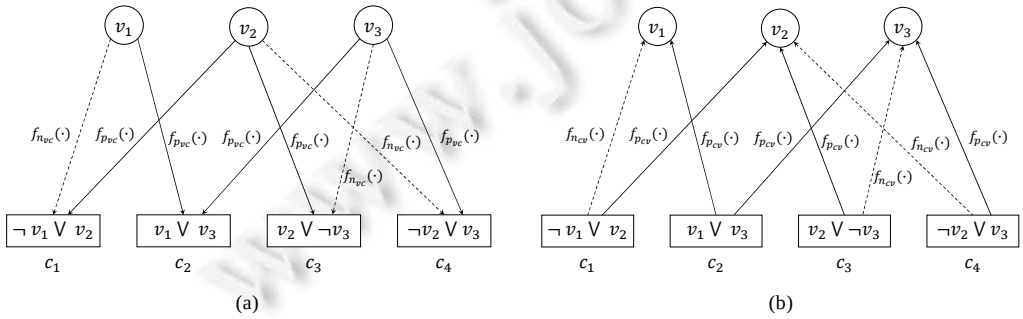


图 2 消息传递关系网络

此外，我们还引入了关系函数 $f_r(\cdot)$来提取变量与子句之间的正文字关系和变量与子句之间的反文字关系的信息，使得模型可以提取到变量和子句之间的关系特征．以下公式描述了用关系传递信息的过程:

$$(C_i^{(t+1)},C_{i,h}^{(t+1)}) \leftarrow g_c([C_i^{(t)},\sum_{r\in R}\sum_{j\in N_i^r} f_r(V_j^{(t)})]) \tag{6}$$

$$(V_i^{(t+1)},V_{i,h}^{(t+1)}) \leftarrow g_v([V_i^{(t)},\sum_{r\in R}\sum_{j\in N_i^r} f_r(C_j^{(t)})]) \tag{7}$$

其中, $g_c(\cdot)$和 $g_v(\cdot)$分别表示用户提取子句和文字特征的 LSTM 网络, $C_i^{(t)}$ 和 $V_i^{(t)}$ 表示在第 t 个迭代中子句 i 和变量 i 的特征, $C_{i,h}^{(t)}$ 和 $V_{i,h}^{(t)}$ 表示在第 t 个迭代中子句 i 和变量 i 的隐藏特征, R 是图中关系的集合, N_i^r 表示节点 i 相邻的处于关系 $r\in R$ 下的节点的集合. $f_r(\cdot)$是用于提取关系 $r\in R$ 特征的 MLP, 其中, $R=\{p_{cv},p_{vc},n_{cv},n_{vc}\}$.

在我们的模型中, 对应于图结构表征中定义的关系 $r\in R$, 分别采用了 4 个 MLP 进行信息提取. 其中: $f_{p_{cv}}(\cdot)$ 和 $f_{n_{cv}}(\cdot)$ 用于提取子句到变量之间的关系特征, 表示包含多个文字特征的子句特征经过关系函数将子句中所有的文字特征通过关系函数传递给变量; $f_{p_{vc}}(\cdot)$ 和 $f_{n_{vc}}(\cdot)$ 用于提取变量到子句之间的关系特征, 表示变量特征经过关系函数变换成为了子句中的一个文字的特征. 此外, 消息传递层只有一层, 但是拥有很多次的迭代. 在一次迭代中, 消息被反复传递, 类似于递归神经网络[19]. 在整个过程中, LSTM 和 MLP 的参数是共享的.

1.3 池化操作与概率预测

在消息传播之后, 需要在图节点上应用池化操作以提取图特征. 在本文中考虑了两种池化方法: 一是采用 Average Pooling 的方式, 另一种是 Attention Pooling. Average Pooling 是将输入的图节点的特征平均之后作为图特征; Attention Pooling 是通过应用注意力机制给输入的节点分配不同的注意力得分, 有选择性地对节点的特征进行提取, 从而得到图特征[20]. 在之前的工作中, 基本都采取使用文字特征池化的结果来预测可满足的概率, 而在本文中, 文字特征并没有直接的编码, 本文中的文字特征是通过在变量特征上应用关系函数动态计算的, 所以我们无法直接利用文字特征池化的结果.

为了预测 SAT 问题的可满足概率, 本文决定采用子句特征池化的结果来预测可满足的概率. 原因有两点: 一是在消息传递的过程中, 由变量特征应用关系函数计算得到的文字特征信息会传递给子句节点, 子句节点因此也就包含了与之相关的文字的所有特征信息; 二是在理论上, 子句的是否可满足与 SAT 问题可满足本身也有直接的关系. 在本文中, 没有考虑使用变量特征的池化结果来预测概率. 因为变量在理论上并不具备判断 SAT 问题的可满足的信息, 之前的相关方法使用文字特征的一个理论依据是 SAT 问题的 CNF 表达式本身就是由一组文字构成的. 因此, 结合两种常用的池化方法得到了以下两种方法.

- 第 1 种方法是文献[13]的做法.

首先使用 MLP 网络基于节点特征输出问题可满足的置信度, 然后对整体节点输出的置信度取平均值,以表示最终模型预测的可满足的置信度. 在我们的方法中, 选择将所有子句的特征提供给 MLP 网络, 以返回标量输出 $y\in[0,1]$表示问题可满足的置信度; 然后, 我们将所有 m 个子句输出的可满足置信度取平均, 公式如下:

$$\hat{y}=\frac{1}{m}\sum_{i\in\{1\ldots m\}}\rho(C_i) \tag{8}$$

其中, $\rho(\cdot)$是 MLP, m 是子句数.

然而, 直接采取平均的方式本质上是认为每一个子句对 SAT 问题是否可满足的影响是一致的, 但是实际情况下却并不一定, 比如子句$(x\vee\neg x)$对 SAT 问题是否可满足根本不影响.

- 第 2 种方法是在图中的子句节点上应用注意力机制[20].

通过注意力机制对节点进行池化操作, 可以得到图的特征表示, 然后再使用 MLP 输出可满足的概率. 整个过程可以表示为如下公式:

$$\hat{y}=\rho\left(\sum_{i=1}^{m} softmax(f_{gate}(C_i))C_i\right) \tag{9}$$

其中, $f_{gate}(\cdot)$是输出注意力得分的函数. 注意力机制可以在训练过程中学习对不同子句应用不同的注意力得分, 这意味着每个节点的特征都有着不同的贡献度, 从而有可以实现有选择性地提取节点信息, 降低信息噪声.

文献[14]通过在训练阶段同时预测 SAT 问题的可满足以及每个变量为真的概率的方式, 明显提高了文献

[13]所提出的模型的预测精度. 因此, 本文同样采用同样的方法来训练模型, 输出层同时输出 SAT 问题的可满足概率以及每个变量为真的概率. 我们添加了一个全连接神经网络在消息传递层之后输出每个变量为真的概率, 公式如下:

$$\hat{v}_i = \phi(v_i) \tag{10}$$

其中, v_i 是 SAT 问题中变量 i 的特征, $\phi(\cdot)$是全连接神经网络.

1.4 目标函数

在训练阶段, 该模型的目标是最大限度地减少二分类损失函数 L_S 的交叉熵, L_S 公式如下:

$$L_S = \sum_{(X_i, y_i)\in D} -y_i \log(\sigma(\hat{y}_i)) - (1-y_i)\log(1-\sigma(\hat{y}_i)) \tag{11}$$

其中, $D=\{(X_i,y_i)|i\in\{1..n\}\}$表示数据集, X_i 和 $y_i\in[0,1]$表示 SAT 实例 i 及其可满足性标签, $\sigma(x)=\dfrac{1}{1+\mathrm{e}^{-x}}$.

通过引入文献[14]的方法, 我们的目标函数定义修改为

$$L = \sum_{(X_i, y_i, v_i)\in D}\left(L_S(y_i, \hat{y}_i) + \frac{1}{\|v_i\|_0}\sum_j L_A(v_{i,j}, \hat{v}_{i,j})\right) \tag{12}$$

其中, $L_A(\cdot,\cdot)$表示二元分类的交叉熵损失; v_i 是问题 i 的一组赋值布尔向量, $\|v_i\|_0$ 是 v_i 的长度.对于不可满足的 SAT 问题, $\dfrac{1}{\|v_i\|_0}$ 的值为 0.

2 实验环境

2.1 数据集

在实验阶段, 我们使用由文献[14]共享的 UR 3-SAT 实例和自己生成的非 UR 3-SAT 数据集. UR 3-SAT 数据集由 11 组数据实例构成, 其变量数以 50 为梯度, 从 100 增加到 600, 每个 SAT 问题实例的子句数 $m=4.258n+58.26n^{-2/3}$, 其中, n 是变量数[21]. 每组数据集分别包含 5 000 个可满足和 5 000 个不可满足的实例. 在我们的实验中, 我们将数据随机分为训练、验证和测试集, 按照 8:1:1 的比例进行分割. 每个数据集中可满足和不可满足的实例保持 1:1, 维持样本均匀分布.

在工业应用中, 我们通常对非 UR 3-SAT 模型更感兴趣. 为了评估 MPR 网络模型是否可以提取非 UR 3-SAT 问题的结构特征, 我们使用 CNFgen 生成器生成了一组数据用于测试[22]. 生成命令是 cnfgen randkcnf *k n m*, 其中, k 是 3, n 和 m 分别是变量数和子句数. 该数据集由 7 组数据构成, 变量数以 100 为梯度, 从 100 增加到 799. 每组包含 100 个可满足实例和 100 个不可满足实例, 子句数与变量数之比为介于 1–10 之间的随机数.

2.2 模型训练

对于 MPR 网络模型的结构, 我们对变量节点和子句节点采用了 128 维的 embedding 向量, 用标准正态分布进行初始化. 在消息传递网络层, 对应关系集合 R, 我们分别使用一个 MLP 作为关系函数, 每个 MLP 有 3 个隐藏层. 此外, 我们用两个 LSTM 网络来更新变量节点和子句节点的特征, 消息传递网络中所有的隐藏层的神经单元都为 128 个. 在消息传递网络之后, 我们使用具有两个隐藏层和一个输出层的 MLP 来提取所有子句的特征, 输出每个子句可满足的概率, MLP 中的隐藏单位也是 128, 输出层维度为 2. 此外, 我们使用全连接神经网络输出每个子句节点的注意力分数、一个 128 维输入和 2 维输出的全连接神经网络提取变量节点的特征, 然后输出变量为真的概率. 每一个隐藏层的激活函数都是 *ReLu*[23].

训练过程中, 我们使用 Adam[24]作为优化器, 初始学习率为 0.001. 此外, 我们还引入了 $L2$ 正则化机制防止模型过拟合, 正则化系数为 10^{-8}. 在每个迭代中, 消息传播 9 次. 在训练过程中, 随机从训练集中抽取 32 个样本作为一个小批量. 最终迭代之后, 对于每个 epoch, 模型都在验证集上进行验证: 如果验证集上的预测精度连续 3 个 epoch 都没有增加, 学习率将调整到原值的一半; 如果准确率在连续 10 个 epoch 不增加, 就停止

训练, 并保存预测精度最高的模型参数.

2.3 基线方法

文献[14]将可变矩阵网络(exchangeable matrix architecture, Exch)与文献[5]中基于 SAT 问题统计特征的随机森林(random forest, RF)模型和文献[13]使用的 MP 网络进行比较. 其中, 可交换矩阵网络和 MP 网络模型都通过端到端学习进行训练. 本文主要与上述 3 种基线方法进行比较.

3 实验结果

3.1 UR 3-SAT数据集上的准确率

我们在每个数据集上都训练了 MPR 网络模型. 表 1 显示了所有数据集上预测的实验结果, 3 种深度网络模型都通过端到端学习进行训练.

使用人工统计的特征作为输入的模型(即随机森林)在预测准确率上比使用结构特征的模型差. MPR 网络的预测精度在 11 组数据上的准确率为 77%–85%, 平均预测准确率为 81.1%. MPR 网络模型的准确率大致上随着 SAT 实例规模的增大而增加. 对于 600 变量数据集, MPR 网络模型的预测精度达到 85%.

表 1 UR 3-SAT 预测准确率

变量数	RF	Exch	MP	MPR (ours)		
				All	Sat	Unsat
100	0.702	0.726	0.751	**0.777**	0.678	0.871
150	0.712	0.745	0.771	**0.780**	0.767	0.793
200	0.734	0.772	0.781	**0.782**	0.790	0.779
250	0.703	0.800	0.788	**0.811**	0.838	0.778
300	0.744	0.800	0.788	**0.825**	0.783	0.857
350	0.734	0.809	0.812	**0.824**	0.821	0.822
400	0.711	**0.790**	0.781	0.785	0.826	0.746
450	0.700	0.789	0.803	**0.824**	0.784	0.871
500	0.773	0.809	0.795	**0.840**	0.826	0.840
550	0.756	0.810	0.813	**0.826**	0.814	0.842
600	0.813	0.837	0.818	**0.850**	0.841	0.849
平均精度	0.734	0.790	0.791	**0.811**	0.797	0.823

在其中 10 组数据集中, 我们的预测准确率比对照组 Exch 模型提高了 1%–5%, 在变量数为 400 的数据集上仅偏低 0.5%. 在所有数据集上, 我们的预测精确度比 MP 网络模型高 1%–4%.

我们的平均预测准确率(81.1%)与 Exch 模型(79.0%)和 MP 网络模型(79.1%)相比有显著差异, 高出了 2%. 在所有数据集上, 我们的模型的预测精度方差是 0.000 6, Exch 模型是 0.000 9, MP 网络模型是 0.000 4. 三者的方差没有太大差距, 意味着本文提出的模型与基线方法相比同样稳定.

3.2 UR 3-SAT数据集上的泛化能力

模型的泛化能力表明一个模型是否可以在不同的数据集上提取 SAT 问题的共性结构特征. 如果模型在小规模数据集上训练后, 可以在较大规模的数据集上获得相当高的预测精度, 就能够大幅减少训练时间和计算资源. 我们在变量数为 100 的数据集上训练模型, 然后在其他更大规模的数据集上进行测试. 表 2 显示了不同数据集上模型泛化能力的对比结果.

结果表明: MPR 网络模型具有良好的泛化能力, 取得了比其他模型更好的预测精度. 在其中 6 组数据集上, MPR 网络的预测精度超过了 80%. 与 Exch 模型平均 77.5%的预测精度和 MP 网络模型平均 70.6%的预测精度相比, 在所有数据集上, MPR 达到了 79.5%的预测精度, 超过其余两个模型. 实验结果表明: 相比于 Exch 模型和 MP 网络模型, MPR 模型能够更充分地提取 SAT 问题的结构特征.

更进一步地, 我们采用 t-SNE 方法[25]对模型提取的变量数为 150 的数据集特征进行了可视化. 如图 3 所示: 在大部分的样本中, 模型对于可满足的问题和不可满足的问题特征可以进行区分, 部分可满足的样本的特征误判成了不可满足的, 因此在表 2 中的实验结果上, 模型在不可满足的数据集上的准确率比可满足数据

集上的准确率更高.

表 2 UR 3-SAT 泛化能力实验结果

变量数	RF-100	Exch-100	MP-100	MPR-100 (ours)		
				All	Sat	Unsat
150	0.695	0.758	0.734	**0.771**	0.683	0.861
200	0.695	0.759	0.704	**0.780**	0.677	0.888
250	0.654	0.776	0.722	**0.800**	0.697	0.901
300	0.711	0.780	0.729	**0.801**	0.668	0.925
350	0.705	0.791	0.725	**0.802**	0.657	0.942
400	0.681	0.756	0.711	**0.761**	0.865	0.657
450	0.692	0.778	0.699	**0.801**	0.664	0.940
500	0.716	0.777	0.686	**0.830**	0.723	0.938
550	0.722	0.768	0.669	**0.790**	0.626	0.953
600	0.739	0.809	0.683	**0.811**	0.677	0.937
平均精度	0.701	0.775	0.706	**0.795**	0.694	0.923

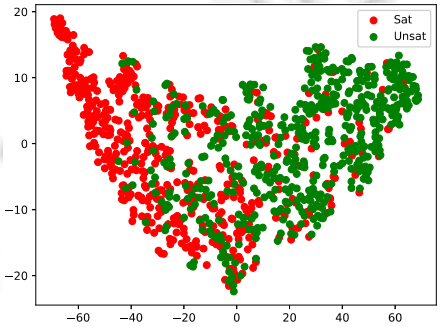


图 3 UR 3-SAT 变量数 150 数据集特征可视化

3.3 非UR 3-SAT数据集上的泛化能力

为了评估在 UR 3-SAT 数据集上训练的 MPR 模型是否可以泛化到非 UR 3-SAT 数据上, 我们在变量数为 100 的 UR 3-SAT 数据集训练 MPR 模型, 在非 UR 3-SAT 数据集上进行了测试该模型.

如表 3 所示, MPR-100 在 7 个数据组上预测的平均精度大于 99%. 该结果表明: 我们的模型可以很好地解决 CNFgen 生成的随机 SAT 问题, 通过采用 t-SNE 方法[25]对 100~299 变量的数据集特征进行可视化发现, 如图 4 所示, 我们的模型可以很好地对可满足的问题与不可满足的问题进行特征提取, 有着明显的特征区分, 因此有着极高的准确率.

为了进一步验证模型的泛化能力, 我们采用 SATLIB 的 Benchmark 数据进行了测试, 选取了 SAT-encoded Quasigroup (or Latin square) instances 数据集作为测试集. 该数据集不是随机的 3-SAT 数据, 与训练所用的随机生成的 3-SAT 数据有着不同的特征分布. 该数据集一共 22 条, 包含 10 条可满足的问题和 12 条不可满足的问题. 该数据集其变量数最大的超过 2 000, 变量数 600 以下的数据集有 6 条[26]. 在该数据集上, 对于可满足的实例, MPR-100 的准确率为 60%, 不可满足的实例准确率为 83%. 这说明 MPR-100 在超过 600 甚至更大规模的数据特征分布不同的数据集上, 泛化性还有待提高.

表 3 non-UR 3-SAT 预测准确率

变量数	子句数	子句/变量比	Sat 实例数	Unsat 实例数	Sat 预测精度	Unsat 预测精度
100–199	133–1844	1.00–9.95	100	100	0.992	0.992
200–299	235–2954	1.03–9.97	100	100	1.000	0.992
200–399	345–3868	1.04–9.89	100	100	1.000	0.984
400–499	411–4532	1.00–9.87	100	100	1.000	1.000
500–599	568–5935	1.01–9.95	100	100	1.000	1.000
600–699	685–6779	1.01–9.98	100	100	1.000	1.000
700–799	783–7865	1.02–9.99	100	100	1.000	1.000

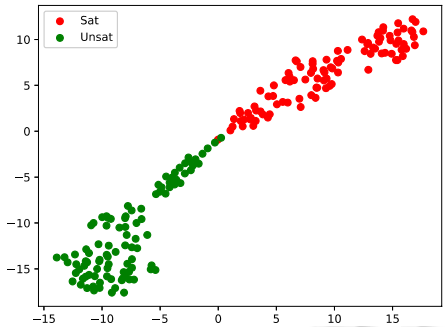


图 4　非 UR 3-SAT 变量数 100–199 数据集特征可视化

3.4 资源需求与运行时间

本文使用 DGL5.0 来构建多关系异构的 MPR 模型，并使用支持自动混合精度(AMP)的 PyTorch 1.6.0 作为后端训练. 用于训练的 GPU 型号为特斯拉 V100 16 G. 在变量数为 100 的实例上训练 Exch 和 MP 网络，分别消耗 500 MB 和 600 MB GPU 内存，而训练我们的 MPR 网络只消耗 100 MB GPU 内存. 该性能优势使我们可以直接批量使用 32 个样本训练. 我们的模型资源需求较小，主要有两个原因: 首先，我们将原始 SAT 实例表征为由变量和子句节点构成的多关系异构图，因此变量节点的数量只有 Exch 模型的一半; 其次，在消息传递层中我们只迭代了 9 次，但是 MP 网络模型迭代了 26 次以上. 在消息传递过程中，我们只使用两个 LSTM 网络来捕获特征. LSTM 有许多参数，更多的迭代需要更多的 GPU 内存. MPR 网络模型的训练大约需要 20 个小时,但是该模型在预测阶段是非常有效的. 我们的模型在 i7 8750H 的 CPU 的环境中进行测试，分别测试了 MPR 模型在变量 100–600 的数据集上的预测时间以及经典的求解器 Glucose 在变量 100–350 的数据集上的运行时间[27]. 如图 5 所示: MPR 在预测阶段的运行时间与问题的规模成线性关系; 而经典求解器 Glucose，其求解时间与 SAT 问题的规模往往成非线性关系. 可以看到: 在变量为 350 时，求解时间已经非常长了，因此没有求解更大规模的 SAT 问题.

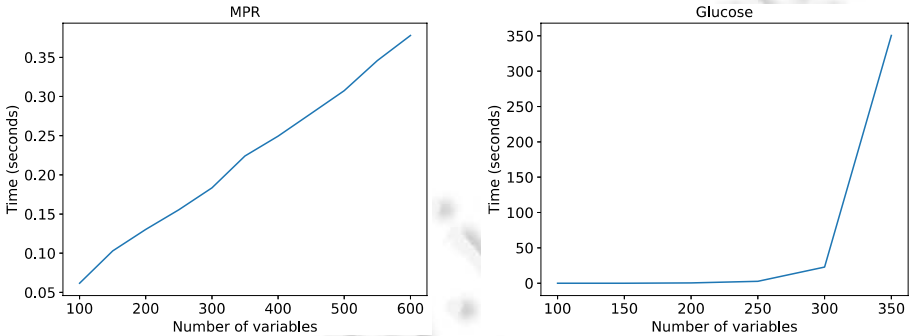


图 5　预测所需时间开销

3.5 MPR与MPR-Attention的比较

如表 4 所示: 在 UR 3-SAT 数据集上, MPR-Attention 在预测准确率方面要优于 MPR 模型，在 7 组数据上预测准确率超过 MPR 模型. 这表示加入了 Attention 机制之后，能够更好地提取 SAT 问题的结构特征，对于可满足数据和不可满足的数据，两者都没有太大的偏好.

从表 5 中可以看出, MPR-Attention 的泛化能力也优于 MPR 模型. 同时可以看到: 相比于 MPR 模型, MPR-Attention 更加稳定一些. MPR-Attention 模型在可满足的数据和不可满足的数据上得到的泛化平均准确率几乎一致, 80%左右; 而 MPR 在不可满足的数据上，泛化平均准确率超过可满足的数据大约 20%. 这表示 MPR 在泛化预测时倾向于认为输入问题是不可满足的. 这说明对于 SAT 问题中的每一个句子，对于 SAT 问题是否可

满足的影响是不同的. Attention 机制的引入对不同的子句分配了不同的注意力, 从而能够更好地提取 SAT 问题的通用特征, 从而表现的更加稳定.

表 4 MPR 与 MPR-Attention 的预测精度比较

变量数	MPR			MPR-attention		
	All	Sat	Unsat	All	Sat	Unsat
100	0.777	0.678	0.871	**0.793**	0.796	0.789
150	**0.780**	0.767	0.793	0.773	0.720	0.825
200	0.782	0.790	0.779	**0.791**	0.770	0.811
250	**0.811**	0.838	0.778	0.802	0.821	0.782
300	0.825	0.783	0.857	**0.825**	0.826	0.823
350	0.824	0.821	0.822	**0.829**	0.841	0.816
400	0.785	0.826	0.746	0.777	0.781	0.772
450	0.824	0.784	0.871	**0.827**	0.806	0.848
500	**0.840**	0.826	0.840	0.825	0.811	0.838
550	0.826	0.814	0.842	**0.826**	0.795	0.857
600	0.850	0.841	0.849	**0.854**	0.855	0.853
平均精度	0.811	0.797	0.823	**0.813**	0.802	0.819

表 5 MPR 与 MPR-Attention 的泛化能力比较

变量数	MPR-100			MPR-attention-100		
	All	Sat	Unsat	All	Sat	Unsat
150	0.771	0.683	0.861	**0.773**	0.814	0.732
200	0.780	0.677	0.888	**0.788**	0.760	0.815
250	**0.800**	0.697	0.901	0.788	0.772	0.804
300	0.801	0.668	0.925	**0.817**	0.776	0.858
350	0.802	0.657	0.942	**0.825**	0.801	0.849
400	**0.761**	0.865	0.657	0.753	0.924	0.581
450	0.801	0.664	0.940	**0.810**	0.772	0.847
500	**0.830**	0.723	0.938	0.815	0.808	0.821
550	0.790	0.626	0.953	**0.827**	0.794	0.859
600	0.811	0.677	0.937	**0.842**	0.841	0.843
平均精度	0.795	0.694	0.923	**0.804**	0.806	0.801

在我们的 non-UR 3-SAT 数据集上, MPR-Attention 的测试结果见表 6. 可以看到: 相较于表 3 中 MPR 模型的结果, MRP-Attention 在预测精度上要差上一些, 但总体上相差不到. 我们在 SATLIB 的 SAT-encoded Quasigroup (or Latin square) instances 数据集也同样进行了测试, 可满足的实例准确率为 80%, 不可满足的实例准确率为 67%.

总体来说, MPR-Attention 与 MPR 模型在大规模的泛化能力上都需要提高, 尤其是在 SATLIB 上的真实数据集上. 本文的模型主要还是针对于 UR 3-SAT 数据集上训练而来, 在 UR 3-SAT 数据集上表现良好. 对于 non-UR 3-SAT 数据样本分布的特征与 UR 3-SAT 还是有很大区别的, 因此没能很好地泛化, 这也是今后需要进一步研究的方向.

表 6 non-UR 3-SAT 上 MPR-Attention 预测准确率

变量数	子句数	子句/变量比	Sat 实例数	Unsat 实例数	Sat 预测精度	Unsat 预测精度
100–199	133–1844	1.00–9.95	100	100	0.969	1.000
200–299	235–2954	1.03–9.97	100	100	0.906	1.000
200–399	345–3868	1.04–9.89	100	100	0.978	1.000
400–499	411–4532	1.00–9.87	100	100	0.930	1.000
500–599	568–5935	1.01–9.95	100	100	1.000	1.000
600–699	685–6779	1.01–9.98	100	100	1.000	1.000
700–799	783–7865	1.02–9.99	100	100	1.000	1.000

4 相关工作

4.1 Survey Propagation的图表征

我们表征 SAT 的方法看似与 Survey Propagation[28]类似, 但其实在 3 个方面具有主要区别: (1) 本文为变

量或子句节点添加了一个 embedding 向量; (2) 在本文的图表示中, 有 4 种类型的边并为每种类型定义了一个关系函数来实现特征学习能力; (3) 本文提出的模型通过对所有子句节点的特征进行提取进而得到图特征, 从而输出问题实例的可满足概率.

4.2 使用图神经网络预测SAT问题可满足性

文献[12]将 SAT 问题转化为由变量节点组成的二分图, 达到了 70%的预测精度. 文献[13]提出了基于消息传递网络求解 SAT 问题的启发式方法, 把解决 SAT 问题的过程分为两步: 首先, 训练模型来预测 SAT 问题可满足的概率; 然后, 在测试阶段将提取的变量特征进行解码, 对于可满足的 SAT 实例, 输出 SAT 问题变量的赋值. 然而, 该方法使用的是文字和句子作为图节点, 且没有利用文字和句子之间的关系信息; 预测 SAT 可满足性的准确率达到了 79%, 但泛化能力有待提高.

4.3 端到端学习

可变矩阵网络[29]可以接受维度变化的数组作为输入, 因此, 文献[13]将原始 CNF 数据编码为稀疏矩阵, 其中, 行表示子句、列表示变量, 矩阵项表示变变量是否出现在子句中以及出现的是正文字还是反文字. 该工作表明: 与其他模型相比, 可变矩阵模型通过端到端学习, 可以获得更好的预测精度, 具有更好的泛化能力; 并通过端到端学习的方式训练 MP 网络[13], 提高了预测精度. 然而, 该工作同样只表征了文字和句子的关系, 并没有表达变量和句子之间的关系. 在我们的工作中, 表征了变量和句子的关系, 采用了端到端学习对模型训练, 在预测精度和泛化能力上均有所提高.

5 总 结

本文首次将 SAT 问题转化为一种多关系异构图, 同时提出了 MPR 网络来预测 SAT 问题的可满足性. 我们的模型可以从 UR 3-SAT 数据中提取结构特征, 并以较高的精度进行可满足性预测. 此外, MPR 网络模型在小规模 UR 3-SAT 数据集上训练之后, 可以泛化到大规模的 UR 3-SAT 实例和更大规模的非 UR 3-SAT 实例. 此外, MPR 网络模型在训练和预测时, 仅仅需要有限的资源. 总结而言, 本文提出的方法在 UR 3-SAT 数据集上预测可满足方面优于其他现有工作的精度、泛化能力和资源需求. 在未来的研究中, 我们将进一步的研究模型特征提取的可解释性以及更贴近工业界的领域相关的 SAT 问题的相关研究, 例如 SAT Competition 上的工业数据, 而不仅仅局限于 CNF gen 生成的 UR 3-SAT 数据.

References:

[1] Mitchell D, Selman B, Levesque H. Hard and easy distributions of SAT problems. AAAI, 1992, 92: 459–465.

[2] Cheeseman PC, Kanefsky B, Taylor WM. Where the really hard problems are. IJCAI, 1991, 91: 331–337.

[3] Mu Z, Hoos HH. On the empirical time complexity of random 3-SAT at the phase transition. In: Proc. of the 24th Int'l Joint Conf. on Artificial Intelligence. 2015.

[4] Sandholm TW. A second order parameter for 3SAT. In: Proc. of the AAAI-96. 1996. 259–265.

[5] Xu L, Hoos H, Leyton-Brown K. Predicting satisfiability at the phase transition. In: Proc. of the 26th AAAI Conf. on Artificial Intelligence, 2012. 584–590.

[6] Nudelman E, Leyton-Brown K, Hoos HH, *et al*. Understanding random SAT: Beyond the clauses-to-variables ratio. In: Proc. of the Int'l Conf. on Principles and Practice of Constraint Programming. Berlin, Heidelberg: Springer, 2004. 438–452.

[7] Xu L, Hutter F, Hoos H, *et al*. Features for SAT. Technology Report, Vancouver: University of British Columbia, 2012.

[8] Liang JH, Oh C, Mathew M, *et al*. Machine learning-based restart policy for CDCL SAT solvers. In: Proc. of the Int'l Conf. on Theory and Applications of Satisfiability Testing. Cham: Springer, 2018. 94–110.

[9] Li Z, Chen Q, Koltun V. Combinatorial optimization with graph convolutional networks and guided tree search. In: Proc. of the Advances in Neural Information Processing Systems. 2018. 539–548.

[10] Selsam D, Bjørner N. Guiding high-performance SAT solvers with unsat-core predictions. In: Proc. of the Int'l Conf. on Theory and Applications of Satisfiability Testing. Cham: Springer, 2019. 336–353.

[11] Loreggia A, Malitsky Y, Samulowitz H, *et al*. Deep learning for algorithm portfolios. In: Proc. of the 30th AAAI Conf. on Artificial Intelligence. 2016.

[12] Bünz B, Lamm M. Graph neural networks and Boolean satisfiability. arXiv: 1702.03592, 2017.

[13] Selsam D, Lamm M, Bünz B, *et al*. Learning a SAT solver from single-bit supervision. In: Proc. of the ICLR (Poster). 2019.

[14] Cameron C, Chen R, Hartford J, *et al*. Predicting propositional satisfiability via end-to-end learning. Proc. of the AAAI Conf. on Artificial Intelligence, 2020, 34(4): 3324–3331.

[15] Gilmer J, Schoenholz SS, Riley PF, *et al*. Neural message passing for quantum chemistry. In: Proc. of the Int'l Conf. on Machine Learning. PMLR, 2017. 1263–1272.

[16] Schlichtkrull M, Kipf TN, Bloem P, *et al*. Modeling relational data with graph convolutional networks. In: Proc. of the European Semantic Web Conf. Cham: Springer, 2018. 593–607.

[17] Duvenaud D, Maclaurin D, Aguilera-Iparraguirre J, *et al*. Convolutional networks on graphs for learning molecular fingerprints. In: Proc. of the NIPS. 2015. 2224–2232.

[18] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735–1780.

[19] Elman JL. Finding structure in time. Cognitive Science, 1990, 14(2): 179–211.

[20] Li Y, Tarlow D, Brockschmidt M, *et al*. Gated graph sequence neural networks. arXiv: 1511.05493, 2015.

[21] Crawford JM, Auton LD. Experimental results on the crossover point in random 3-SAT. Artificial Intelligence, 1996, 81(1-2): 31–57.

[22] Lauria M, Elffers J, Nordström J, *et al*. CNFgen: A generator of crafted benchmarks. In: Proc. of the Int'l Conf. on Theory and Applications of Satisfiability Testing. Cham: Springer, 2017. 464–473.

[23] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. In: Proc. of the 14th Int'l Conf. on Artificial Intelligence and Statistics. 2011. 315–323.

[24] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv: 1412.6980, 2014.

[25] Van der Maaten L, Hinton G. Visu-Alizing data using t-SNE. Journal of Machine Learning Research, 2008. 2579–2605.

[26] Zhang H, Stickel M. Implementing the davis–putnam method. Journal of Automated Reasoning, 2000, 24(1): 277–296.

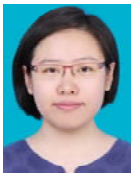
[27] Simon L, Audemard G. Predicting learnt clauses quality in modern sat solver. In: Proc. of the 21st Int'l Joint Conf. on Artificial Intelligence (IJCAI 2009). 2009.

[28] Braunstein A, Mézard M, Zecchina R. Survey propagation: An algorithm for satisfiability. Random Structures & Algorithms, 2005, 27(2): 201–226.

[29] Hartford J, Graham D, Leyton-Brown K, *et al*. Deep models of interactions across sets. In: Proc. of the Int'l Conf. on Machine Learning. PMLR, 2018. 1909–1918.

包冬庆(1996－), 硕士生, CCF 学生会员, 主要研究领域为机器学习, 形式化方法.

翟树茂(1998－), 男, 硕士生, CCF 学生会员, 主要研究领域为形式化方法, 机器学习.



葛宁(1983－), 女, 副教授, 博士生导师, CCF 专业会员, 主要研究领域为形式化方法, 模型驱动, 智能化软件工程.

张莉(1968－), 女, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为模型驱动, 软件工程.