人工智能系统的形式化验证技术研究进展与趋势

CCF 形式化专业委员会

卜　磊[1]　陈立前[2]　董云卫[3]　黄小炜[4]　李建霖[5]　李　钦[6]
刘万伟[2]　阮文杰[7]　宋　富[8]　孙有程[9]　王竟亦[10]　吴　敏[11]
许智武[12]　薛　白[5]　杨鹏飞[5]　易新平[4]　张立军[5]　张　民[6]

[1]南京大学，南京
[2]国防科技大学，长沙
[3]西北工业大学，西安
[4]利物浦大学，利物浦
[5]中国科学院软件研究所，北京
[6]华东师范大学，上海
[7]埃克塞特大学，埃克塞特
[8]上海科技大学，上海
[9]贝尔法斯特女王大学，贝尔法斯特
[10]浙江大学，杭州
[11]牛津大学，牛津
[12]深圳大学，深圳

摘　要

随着深度学习技术在无人驾驶、智能制造、医疗诊断等安全攸关领域的应用，人们对人工智能系统的可信性提出了更高的要求，相关研究也被人工智能、形式化方法领域密切关注，并迅速成为研究热点之一。本文以人工智能系统的安全可信性为主题，从人工智能系统可信的内涵，验证、测试、模型抽象等方面介绍国内外面向人工智能系统形式化验证相关的最新方法与技术，并对该方向的发展趋势进行分析与总结。

关键词：人工智能，神经网络，可信性，形式化验证

Abstract

Deep learning has shown its rapid progress and tremendous success in many safety-critical fields, such as self-driving, intelligent manufacturing, and medical diagnosis. These fields feature with requirement of high dependability and reliability, and therefore corresponding systems and applications must be trustworthy. This report attempts to provide a comprehensive survey on state-of-art methodologies and approaches for improving the trustworthiness of artificial intelligent systems. The survey covers several aspects including concept of trustworthiness of AI, formal verification, testing, modeling, etc. We also sum up the difficulties and look forward to future research directions and trend of this field.

Keywords: Artificial Intelligence, neural network, trustworthiness, formal verification

1 引言

深度学习技术在过去几年获得广泛的关注，因其在一些长期未解决的任务比如图像识别、自然语言处理、声音识别等取得了与人类相当的能力。随着技术的长足进步，越来越多的应用随之而生。时至今日，深度学习已经被大量用于医药、金融、交通、国防、电力等行业。随着这股应用风潮的兴起，另一种声音也逐渐出现在各种媒体讨论、政府报告、行业报告、学术讨论中，也就是，需要保障人工智能系统的安全性和可靠性等可信性质，特别是当它被应用到一些涉及人身安全、关键基础设施安全、财产安全的领域。

人工智能系统的安全可信性问题不只是在人工智能或者机器学习领域得到关注，也引起了形式化方法领域的关注。传统上，形式化方法分析和验证主要针对软硬件系统。尽管以神经网络为代表的深度学习系统也可以实现为软件或者硬件，一个显著的不同在于，传统形式化方法研究的软硬件系统一般有严格的逻辑描述或者结构化描述（也就是符号系统），而神经网络缺乏逻辑结构。神经网络是通过大量的神经元之间的连接构造起来的系统，并通过优化算法来学习系统参数（也就是亚符号系统）。从符号到亚符号系统的变迁使得形式化方法领域半个世纪来发展的大量技术并不能直接被用于深度学习的可信性研究。

本文从形式化方法的角度出发观察过去三年内（主要从 2017 年起）基于形式化方法的人工智能系统安全可信研究的进展。我们从人工智能系统的安全可信内涵，形式化验证，测试，可解释性与模型抽取等方面分别梳理了当前国内外最新的研究成果。其中，人工智能系统的可解释性与系统模型抽取等技术分别从已训练模型和训练过程等方面分析和理解神经网络的性能和工作原理，试图打开“黑盒”。将这些技术成功地吸收到形式化方法中也许将会是形式化方法在处理深度学习系统可信性问题上获得成功的关键。同时，我们给出对抗攻击的一些主要工作，希望未来形式化方法在处理深度学习系统可信性问题上能更多结合深度学习领域的发展。第 4 节综合比较了国内外的研究进展情况。第 5 节分析和展望了本领域的发展趋势和前景。第 6 节对全文做了总结。

2 国际研究现状

2.1 人工智能系统安全内涵

关于人工智能系统的可信性目前并没有统一的定义。大多数文献主要围绕其某个具体的性质进行研究。目前主要研究的性质包括：鲁棒性，安全性，可靠性，可解释性等性质。这些性质也尚未形成统一的定义，其含义也有部分重叠。下文介绍相关性质目前

较为认可的含义。

狭义上讲，鲁棒性指的是对于系统的输入数据中出现的小的偏差（噪声）以及非正常分布的数据，系统的结果和性能不应当受到大的影响[1-2]。广义上讲，鲁棒性还包括对其操作人员所犯的错误具有鲁棒性，能够识别错误的执行方式，能够坚持真实世界的规则并修正模型中明显的错误，以及对真实世界中未建模的部分具有鲁棒性等含义[3]。本文后续讨论的鲁棒性特指狭义上的鲁棒性。

人工智能系统的安全性指系统在对抗环境下可通过特定的神经网络安全防御和保护技术抵御外部攻击，防止恶意的对抗样本误导模型做出错误的判断和命令，保护数据隐私等。外部攻击者多利用系统的非鲁棒性，通过施加小的对抗性扰动诱使系统错误地分类，达到攻击的目的。因此，系统的鲁棒性是安全性的前提。有些文献也将安全性简单地专指为鲁棒性。

人工智能系统的正确性指对于任意一个有效的输入，系统都可以产生正确的输出。这里的正确是相对于人而言。即假设人总是可以做出正确的判断，一个系统是正确的当且仅当对于任何一个输入，人和系统做出的判断总是相同的。然而由于人的这种判断往往依赖于经验和意识，无法被精确的定义。因此，人工智能系统往往缺少一个具体的规范，而无法被形式化地验证。

新加坡国立大学梁振凯团队围绕对抗场景下的神经网络反演展开了研究。神经网络反演技术的目标是从模型的预测结果中反向推演出模型的输入数据，从而达到窃取隐私数据的目的。梁振凯团队提出两种网络反演技术，能够在黑盒条件下训练反演网络。第一种方法基于目标分类任务的背景知识生成相应的辅助训练数据对反演网络进行训练，经过实验表明，该方法能够显著改善反演效率。第二种方法针对只能获得目标模型的部分预测结果的情况，采用截断后的预测结果作为训练数据来训练反演模型，实验表明，该方法能够有效地反演出给定分类预测结果的某个输入数据。作者使用网络反演方法在亚马逊的人脸识别开放系统中进行了实验，证明该方法能够在黑盒条件下成功反演出目标用户的人脸图像[4]。

普林斯顿大学的研究人员综合考虑了可信机器学习中的隐私性与鲁棒性，提出了新的基于对抗样本预测的“成员推断攻击”的概念[5]，并且通过分析高鲁棒模型的内部结构特征。“成员推断攻击”的原理是通过判定特定的数据是否属于目标机器学习的训练集/测试集，对学习目标进行攻击，提出“鲁棒模型往往泄漏了更多的成员数据信息”和“其训练算法本身往往使得模型对训练数据更加敏感”等结论，从而系统地解释了为什么这种模型反而具有更高的隐私风险。这篇论文的主要影响在于启示人们：安全与隐私之间是密切不可分割的两个方面，必须要将二者作为相互关联的领域进行研究。

多数的神经网络对于对抗样本具有较弱的容错能力。但是，构造对抗样本往往需要较高的计算代价。针对神经网络，文献［6］提出了一种程序性噪声的概念（称之为 Perlin 噪声），可广泛应用于计算机图形学、游戏中。论文借助于此概念以及黑盒贝叶斯优化，给出了系统学习出 Perlin 噪声的参数的方法。该方法能够以很低计算开销构造针对学习模型的黑盒攻击。这种攻击可以实施于 Inception v3 等鲁棒神经网络中。

本节给出与可信性相关的一些性质的形式化的表示并分析这些性质之间的相互蕴含关系。

对抗样本 虽然人工智能系统的正确性无法被直接验证，但是我们可以通过寻找人类能够正确分类而神经网络却不能的输入样本来表明系统可能存在的缺陷。这些输入被称为对抗样本[7]。具体地讲，假设人和系统对同一个输入 x 的判断一致，而对于另一输入 $\hat{x}$ 人的判断与输入 x 的结果一致，而系统的判断不一致，则将 $\hat{x}$ 称为 x 的对抗样本。

局部鲁棒性 我们遵循黄等人在文献［8］中对深度神经网络局部鲁棒性的定义：一个深度神经网络对于一个输入空间 η 是局部鲁棒的，当且仅当该网络对 η 中所有的输入的判定结果是相同的。假设该神经网络可以用函数 f 表示，则局部鲁棒性可定义为：

$$Robust(f,\eta) \triangleq \exists l \in L \forall x \in \eta \forall j \in L: f_l(x) \geqslant f_j(x)$$

这里，L 表示所有分类标签的集合。如果针对某个给定的判定结果 j，网络对 η 中所有的输入的判定结果均为 j，则该网络针对 j 和 η 是目标局部鲁棒的。一般来说，η 是一个相对较小的输入空间。

输出可达性 假设一个神经网络可以用一个函数 f 表示，给定一个输入空间 η，定义 $\mathrm{P}(f, \eta) = \{f(x) \mid x \in \eta\}$。即 $\mathrm{P}(f, \eta)$ 表示 η 中所有输入的输出结果集合。通过计算 $\mathrm{P}(f, \eta)$ 可用于判断是否所有的输出结果等于或包含于某个已知的安全的集合 Ψ，进而验证对应神经网络的安全性。然而由于 η 通常为连续空间，并且 f 通常为非线性函数，计算 $\mathrm{P}(f, \eta)$ 并非易事。

区间属性 给定一个输入空间 η，相比于计算 $\mathrm{P}(f, \eta)$，计算一个能包含 $\mathrm{P}(f, \eta)$ 的凸集合则相对容易。我们用 $\mathrm{I}(f, \eta)$ 表示一个凸集合，且其满足 $\mathrm{I}(f, \eta) \supseteq \{f(x) \mid x \in \eta\}$。我们称 $\mathrm{I}(f, \eta)$ 为一个区间。显然地，有些包含集合 $\mathrm{P}(f, \eta)$ 的区间可以很容易地表示。然而我们希望能计算出能包含 $\mathrm{P}(f, \eta)$ 的最小区间。该区间被称为可达集合 $\mathrm{P}(f, \eta)$ 的上近似（over-approximation）。给定一个输入空间 η 和一个安全的凸集合 Ψ，如果能验证 f 满足 $\Psi \supseteq \{f(x) \mid x \in \eta\}$，即证明了对应的神经网络在空间 η 的安全性。

利普希茨属性 利普希茨连续是目前研究神经网络安全性的又一方法。一个函数 f: $R^n \to R^m$ 被称为是利普希茨连续的当且仅当存在一个常数 K 使得对任意 x，$y \in R^n$，满足 $\|f(y) - f(x)\| \leqslant K\|y - x\|$。其中，$\|y - x\|$ 表示两个向量之间的距离。另外，向量之间的距离也可以通过不同的范数表示。直观地讲，利普希茨连续限制了函数的变化速度不能超过某个具体的值。给定一个神经网络其可以表示为函数 f，一个输入空间 η 和范数 L_p，用 $Lips(f, \eta, L_p)$ 表示函数 f 在 η 上的利普希茨值。通过比较 $Lips(f, \eta, L_p)$ 与某个已知的数值 d 的大小，可用于分析验证对应神经网络的鲁棒性和安全性。

属性之间的关系 黄等人[9]在其关于深度神经网络验证的综述中讨论了上述性质之间的关系。如图 1 所示，箭头表示后者可以通过前者计算得到。例如 $I(f, \eta)$ 的值可以通过 $Lips(f, \eta, L_p)$ 计算得出。显然地，通过 $P(f, \eta)$ 可

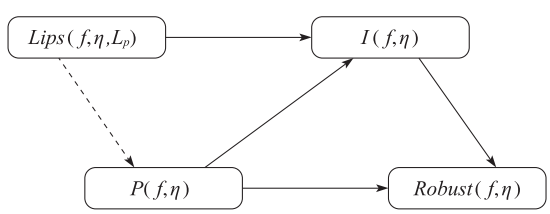


图 1 四种属性之间的关系[9]

以容易地计算出 $I(f, \eta)$，进而可以判定 f 在空间 η 上是否是鲁棒的。阮等人[10]在其工作中给出了如何通过利普希茨值 $Lips(f, \eta, L_p)$ 计算出 $P(f, \eta)$。

2.2 人工智能系统形式化验证技术

有关深度神经网络鲁棒性的形式化验证依照所用底层技术的不同，大致可以分为约束求解、基于搜索、全局优化以及上近似四类，当然各类别之间的划分并不绝对严格。此外，针对形式化验证技术可以提供的不同类型的保证，又可以将现有的深度神经网络的验证工作分为下面几个类别：

- 确定性保证，即可以“确定”地阐述某个性质是否成立。
- 单边性保证，即只需提供一个“上界”或者“下界”，便可作为判断某个性质是否成立的充分条件。
- 收敛性保证，即针对某个性质成立提供“收敛”的上下界。
- 统计性保证，即量化某个性质成立的“概率”。

下文将依照这四个提供不同类型的保证的分类，来详细介绍现下验证深度神经网络的工作。另外，读者也可参考 Liu 等人的文章[11]来从可达性、优化和搜索的角度了解不同神经网络的验证算法。

2.2.1 验证神经网络的确定性保证

在验证深度神经网络的过程中，确定性保证方法首先将所验证问题转化成为一个约束的集合（无论是否具有优化目标函数），然后使用约束求解的方法来解决这个问题。此处，大部分约束求解器对于某个查询通常能够返回一个确定的解答，也即“满足”或者“不满足”，进而提供确定性的结果。此类方法的成功得益于当下各类优秀的约束求解器，譬如 SMT 求解器、SAT 求解器、线性规划求解器、混合整数规划求解器等。

基于 SMT/SAT 的求解方法 2010 年，Pulina 和 Tacchella[12]提出了一种基于 SMT 求解的抽象精化（abstraction-refinement）方法来验证神经网络。一方面，从抽象的角度来说，某个深度神经网络可以被抽象为一系列线性算术约束的逻辑复合（Boolean combinations），因而只需确保这个抽象模型是安全的，那么其对应的具象模型便也是安全的；另一方面，从精化的角度来说，返回的可疑反例（Counter-examples）会触发精化过程，从而被用来自动化校正模型的错误行为。遗憾的是这种方法仅仅可以作用于不超过 10 个神经元、并且激活函数为逻辑函数的神经网络上。2020 年，Elboher 等人提出一种新的神经网络抽象技术，通过将多个神经元融合成一个神经元，降低整个神经网络的大小以实现高效的验证[13]。

2017 年 Katz 等人[14]提出了针对神经网络的 SMT 求解器 Reluplex，Ehlers[15]发表了基于 SMT 的 Planet 工具等。使用 SMT 求解器的优势在于，当需要验证的神经网络的性质可以被表达成 SMT 约束的逻辑复合的时候，此类求解器通常具有更好的结果。Reluplex 和 Planet 的共通点在于，两者都使用了 Davis-PutnamLogemann-Loveland（DPLL）算法的架

构来分裂不同情形以及排除有冲突的子句。其不同之处在于，Reluplex 继承了 Simplex 算法中的规则，并且在此基础上加入了针对 ReLU 激活函数的其他规则，也就是说，Reluplex 首先寻求针对线性约束的解然后再看每个神经元是否满足针对 ReLU 的规则，而另一工具 Planet 则使用线性近似来上近似神经网络的性质，并使用逻辑公式来看每个神经元是否满足 ReLU 和 max-pooling 等操作。基于 Reluplex 的工作，Katz 等人[16]于 2019 年提出了 Marabou 框架，此框架不再局限于 ReLU 激活函数，而是可以验证全连接和卷积（convolutional）神经网络。

除了以上基于 SMT 求解的验证深度神经网络的方法，Narodytska 等人[17-18]于 2018 年提出了基于 SAT 求解来验证一类特殊的神经网络，也即二值神经网络（Binarised Neural Network）。此类神经网络将权值以及隐藏层的激活值进行二值化操作，因此可以被转化为大家所熟悉的布尔可满足性质（Boolean satisfiability）编码。通过使用这种布尔编码，可以通过现有 sat 求解器验证二值神经网络的性质。同时结合其他反例制导（counter-example guided）的搜索过程来实现。文献［17-18］中的方法尤其关注神经网络对于对抗扰动（adversarial perturbations）的鲁棒性，其实验表明此方法可以作用于中度大小的深度神经网络，譬如可以用来进行图像识别任务的这类神经网络。

基于混合整数规划（MILP）的求解方法 Lomuscio 和 Maganti[19]于 2017 年将全连接（fully-connected）神经网络编码成混合整数规划，譬如对于神经网络的某个隐藏层来说，其上的 ReLU 激活函数就可以被表达成一系列混合整数规划，以便计算输出范围（output range）。然而，仅仅是简单地使用混合整数规划来验证深度神经网络并不够高效。同年，程等人[21]则在混合整数规划的基础上加入了启发法（heuristics）来加速求解过程，并且在实验过程中利用并行的 MILP 求解器也使得计算过程几乎达到线性。另外，Dutta 等人[21]也提出了 Sherlock 算法，利用局部搜索和全局搜索互相交替的方式来高效地计算输出范围。在局部搜索阶段，Sherlock 使用梯度下降方法来找到局部最大值或最小值；在全局搜索阶段，此方法将问题编码成 MILP 来检查局部最大值或最小值是否为全局输出范围。

除此之外，2017 年 Bunel 等人[22]提出了一种分枝限定（Branch and Bound，B&B）算法，并且宣称无论是基于 SMT/SAT 的方法，还是基于 MILP 的方法都可以作为此类分枝限定算法的特殊形式。

2.2.2 验证神经网络的单边性保证

“单边保证”（one-sided guarantees）是指只需计算出一个上界或者下界作为判断某个性质是否成立的充分条件。此类方法的特别之处在于，虽然对于某个参数的值只能提供一个有界估计（bounded estimation），但是却可以在更大的神经网络上工作，譬如包含 10 000个隐藏神经元的网络等。与此同时，这类方法的优势在于它避免了现有约束求解器的实现过程中可能会出现的浮点问题。事实上，大部分现有的约束求解器在进行浮点计算时也只能提供估计值，并且有可能发生计算出的估计值并不是真实最优解，甚至估计值在可行空间之外的情况[23]。也就是说，某个约束求解器有可能会错误地判断某个性

质是满足还是不满足，比如 Dutta 等人[21]就提出在工具 Reluplex 的使用过程中存在一些错误判断的情况，并且指出此类问题有可能来自不可靠的浮点实现。

抽象解释 如果把一个神经网络看作顺序执行的若干赋值语句组成的程序，则可以使用抽象解释技术分析神经网络。抽象解释是一种用于对计算机科学中复杂数学结构进行抽象和近似的理论，由法国科学家 Patrick Cousot 和 Radhia Cousot 于 20 世纪 70 年代提出，最初主要以计算机程序的语义为研究对象，提供了一个统一的理论框架来对程序语义进行抽象和推理，并在程序分析与验证领域得到了广泛应用[24]。利用静态分析，抽象解释可以在不直接运行某个程序的前提下验证该程序的性质。抽象解释的基本思想是利用抽象域（abstract domains）的概念来对于某个输入集合的计算进行上近似。抽象域一般是面向某类特定性质设计的，选择合适的抽象域决定了抽象解释是否高效和精准。在实际情况中，抽象域一般包含了某些类型的特殊形状，而这些形状又可以被表达成逻辑约束的合集。比如，在欧几里得空间中，最为广泛的抽象域有区间（interval）、环带胞形（zonotope）[25]、多面体（polyhedron）等。开源的抽象域库有 APRON[26]和 ELINA[27]等。其中，被用于神经网络验证的抽象域包括区间抽象域、zonotope 抽象域、多面体抽象域等数值抽象域。

神经网络可以看作是一类特殊的程序，输入一般是高维的，激活函数是非线性的，而且实际应用中神经网络中包含的神经元数量往往非常庞大。因此，对神经网络进行精确推理代价很大，从而需要采用抽象解释对神经网络的具体语义进行抽象，使得在抽象语义上进行推理复杂度更低、效率更高。抽象解释在神经网络验证领域做到了效率和精度的权衡，是目前最流行的神经网络验证技术。近些年，Gehr[28]、Mirman[29]、杨[30]等人都在这方面发表过一些工作。

基于抽象解释来对神经网络进行分析，需要考虑设计的抽象语义，在分析精度和计算效率之间取得权衡。这种由某种语义抽象及其上的操作所构成的数学结构称为抽象域。例如，计算机视觉的研究中以及机器学习领域中，对抗样本都是研究的热门方向。它指的是通过对输入样本做一些微小的扰动，在人类观察时并不会认为图片的内容改变了，却能够使神经网络对图片产生错误的输出。我们针对神经网络可能存在对抗样本这一现象，考虑神经网络的局部鲁棒性，即在给定输入 x 和微小扰动 d 的范围内验证是否存在对抗样本，每层神经元的输入范围使用抽象元素近似，而层与层之间的计算通过抽象域中的域操作进行可靠建模。抽象解释基于严格的理论，保证了基于上近似抽象的推理具有可靠性。基于上近似抽象推理得出的性质，在神经网络中一定成立。但是由于抽象中上近似量带来精度损失，不能够保证所有在神经网络中成立的性质都能推理得到。

苏黎世理工学院（ETH）的 Vechev 领导的研究组最早提出了一种基于抽象解释的框架 AI^2，来验证神经网络的安全性和鲁棒性[28]，其主要思想是使用一组带条件的仿射函数来建模基于 ReLU 的神经网络，可以刻画神经网络中的全连接、卷积和 max-pooling 等多种结构，在验证过程中使用区间抽象域、zonotope 抽象域及其幂集抽象域来分析这些仿射函数，最后得到输出层变量的取值范围或变量之间的约束关系。文献［25］完整地介绍了 AI^2 并且对 20 个神经网络进行了广泛的评估实验。结果表明：1）AI^2 足够精确以证

明有用的软件规范或者性质（例如，稳健性）；2）AI^2 可以用来验证最新的稠密神经网络防御的有效性；3）AI^2 明显快于基于符号分析的现有工具，后者通常需要数小时才能验证简单的全连接神经网络；4）AI^2 可以处理深度卷积网络，这是当时其他基于线性规划和 SMT 的方法无法企及的。

在这之后，一些针对神经网络验证的抽象域相继被提出。ETH 研究组对 zonotope 抽象域的抽象转换操作进行了改进，以更契合神经网络中非线性激活函数的特点，支持 ReLU、tanh、sigmoid 激活函数，并基于改进后的 zonotope 抽象域开发了 DeepZ 系统[31]，以验证神经网络的鲁棒性，并在包含 8 万多个神经元的神经网络上开展了实验。DeepZ 不再通过对使用抽象域的交约束和并操作处理激活函数，而是在经典 zonotope 抽象域的基础上为 ReLU、sigmoid 和 tanh 的激活函数增加了抽象变换，取得更精确的结果。

另外，该研究组还面向神经网络验证设计了专门的子多面体抽象域[32]，其主要思想是为每个神经元节点维护一个值区间、一条符号化上界约束和一条符号化下界约束，并针对仿射转换、ReLU 函数、sigmoid 函数、tanh 函数、max-pooling 函数等神经网络中的常见函数设计了定制化的抽象转换操作。该抽象域能够证明诸如输入图像被旋转等复杂扰动下神经网络的鲁棒性。基于该抽象域，ETH 研究组开发了相应的工具 DeepPoly，并考虑了抽象域的可靠浮点实现方法以提高验证的可扩展性和效率，在支持关系型约束和提高分析精度的同时，也能有效地验证大型网络。

为了提高基于抽象解释的神经网络验证的精度，ETH 研究组将基于抽象解释的方法与更为精确的基于线性规划的方法进行了结合[33]，设计了启发式策略来定位哪些神经元在采用了抽象解释的上近似分析（如使用 zonotope 抽象域）之后所得的区间信息仍需要采用基于线性规划的方法来进行进一步精化。基于该方法，ETH 研究组实现了一个神经网络验证系统 RefineZono，并通过实验表明该方法既能提高基于抽象解释的不完备（incomplete）验证方法的精确性，又能够验证一些目前完备验证方法因可扩展性限制不能验证的鲁棒性质。

除了在神经网络验证方面，ETH 研究组还在结合形式化方法的神经网络训练与查询方面开展了研究，提出一种称为可微抽象解释（differentiable abstract interpretation）的方法，能够利用抽象解释来训练大规模神经网络，并保证训练出来的神经网络天然满足一些鲁棒性质[29]。该研究组还提出了一种面向深度学习的可微逻辑（Differentiable Logic）[34]，并开发了系统 DL2，以支持带逻辑公式约束的神经网络的训练和查询。DL2 支持对模型的输入、输出和内部进行逻辑规约，但不支持量词。使用 DL2，用户可以通过编写逻辑约束的方式对领域知识进行声明性规约并要求神经网络训练过程中必须遵循，或者对神经网络模型提出查询，以找到满足给定逻辑约束（如违反鲁棒性）的输入。DL2 的内部工作原理是将逻辑约束转化为具有良好数学性质的可微损失函数（如将合取操作转换为损失函数的加法、析取操作转换为损失函数的乘法等），然后使用标准的基于梯度的方法对损失函数进行最小化优化。DL2 在无监督学习、半监督学习、有监督学习等学习场景下都取得了较好效果。

在验证大规模卷积神经网络时，使用复杂抽象域往往面临内存占用过大等不切实际

的困难。为了对特定抽象域进行优化，符号传播的技术在神经网络的验证中也有很显著的效果[30]。在局部鲁棒性的意义下，大量神经元保持始终激活或始终不激活，ReLU 函数会退化成线性函数或常数 0，因此可以使用符号传播的技术来提高抽象解释的精确度。实验结果表明了使用区间抽象域和 zonotope 抽象域的精确度在加入符号传播后都有明显提升。文献［30］中的工作同时使用抽象解释为每个神经元分析得到的取值范围来帮助 SMT 求解器，在验证 ACAS Xu 网络的对抗样本性质时，取值约束显著加速了的求解速度。

线性近似 针对神经网络的形式化验证工作，一些科研工作人员也提出了与线性近似（linear approximation）相关的方法。例如，Weng 等人[35]在考虑激活函数为 ReLU 的神经元时，提出了在局部对神经网络用线性函数做上近似和下近似，从而帮助验证网络的鲁棒性的 Fast-Lin 算法，并由此给出 ReLU 神经网络在局部的利普希茨常数的上界的 Fast-Lip 算法，从而对神经网络的输出范围做出可靠而又相对精确的近似。另外，张等人[36-37]则将上述方法分别进行了扩展：一方面，在 Fast-Lin 的基础上提出了 Crown 算法，此方法除了允许对于上界和下界的线性表达式可以不同之外，也将激活函数从 ReLU 扩展到其他如 tanh、sigmoid 和 arctan 等函数；另一方面，在 Fast-Lip 的基础上则是提出了 RecurJac 算法来增强对于利普希茨常数的计算。需要注意的是，以上工作受限于激活函数的类型并且只能作用于简单的全连接神经网络，最近研究人员则将这些工作扩展到了更为复杂的卷积神经网络和循环（recurrent）神经网络上。2019 年，Boopathy 等人[38]提出了可以用来验证卷积网络的 CNN-CERT 算法，此方法不仅可以工作在不同激活函数上，而且也能验证如卷积层、max-pooling 层、batch normalisation 层以及 residual blocks 等不同的神经网络结构。同年，Ko 等人[39]则提出了针对循环网络结构的 POPQORN 算法，可以用来验证简单的 RNN（vanilla RNN）、长短期记忆（Long Short-Term Memory，LSTM）和 GRU（Gated Recurrent Unit）等。

凸优化 2018 年，Wong 和 Kolter[40]提出了凸优化（convex optimisation）的方法来学习包含 ReLU 激活函数的神经网络，即使这些网络已经被证明在训练数据集上对于对抗扰动是鲁棒的。这种方法的基本思想是，对于那些由对抗扰动而能达到的激活值的集合，采用凸外上近似（convex outer over-approximation）的方法，然后采取一些鲁棒优化过程来使得在这个凸区域上最坏情况下的损失（Loss）最小化。这里提到的凸区域可以由线性规划来获得，并且这一过程与神经网络中的反向传播算法类似，从而使得计算鲁棒损失的单边保证可以具有非常高效的优化途径。实验表明，这类方法可以被运用在一些已经被证明具有对抗扰动鲁棒性的神经网络上，譬如针对 MNIST 数据集，使用这种方法可以产生某个卷积神经网络，使得当对抗攻击在 Lp 范数小于 0.1 的情况下，所产生的测试错误可以减小至 5.8%。

除此之外，Dvijotham 等人[41]针对类似问题提出了不同的变形，他们通过利用拉格朗日松弛（Lagrangian relaxation）算法作用在优化上，以此绕过了那些非凸（non-convex）的优化问题。

区间分析 2018 年，王等人[42]提出区间运算（interval arithmetic）可以用于计算深

度神经网络的输出结果的界值。其中心思想是，在给定运算数（operands）范围的前提下，只需要得到运算数的上界或者下界的值，便可以计算出最后输出结果的上近似范围。对深度神经网络来说，从输入层之后的第一个隐藏层开始，这类计算可以直接层层递进至网络的最后输出层。除了这种可以进行确切计算的方法之外，符号区间分析以及其他的优化算法也可以用来最小化输出结果上下界的上近似。具体采用区间分析来验证含有ReLU 激活函数的深度神经网络的安全性质的工具可以参考 ReluVal，其优势之处在于，相比那些基于约束求解的途径，它更容易进行并行计算。总的来说，区间分析与上文提到的基于区间的抽象解释有其共通之处。

另外，在 2017 年 Peck 等[43]也提出，利用神经网络不同层上的激活函数也可以推演出对抗扰动的下界值，从而改变神经网络的最终分类结果。这里的下界值具有理论上的保证，那就是只要是任何低于此下界值的对抗扰动都可以被认为是安全的，也就是说不会产生对抗样本。并且，这种计算方法比较高效，其基本上与给定网络的超参数的数量以及输入维度的大小成线性相关，也就使得它的适应性更强，可以运作在不同分类器的鲁棒性分析上。

输出可达集估计 2018 年，向等人[44]提出了另一种叫作输出可达集估计的方法来评估神经网络的鲁棒性。在给定某个神经网络和输入集合的前提下，其对应的输出值会有一个可达集，称之为输出可达集。此类方法的原理是，一方面可以计算出这个输出可达集的某个近似估计，另一方面也可以检查这一输出可达集与某些安全规格（specification）的非（negation）的交集是否为空。他们提出了一个最大灵敏度（sensitivity）的概念，当神经网络的激活函数为单调函数时，此最大灵敏度可以通过解决凸优化问题来计算，因此神经网络的输出可达集估计问题就可以转换为一系列优化问题。实验表明，输出可达值估计的方法可以被用于神经网络的自动化安全性质的验证，譬如运用在验证机器人手臂模型的安全性问题上。

2.2.3 验证神经网络的收敛性保证

上述提到的验证神经网络的方法通常只能运用在小型网络上，譬如隐藏神经元在千个数量级别的网络，但是在实际应用中，现下的深度神经网络基本含有至少上百万数量级别的隐藏神经元。因此在应对现实生活中的运用时，能否验证此类较大型神经网络的鲁棒性就变得尤为重要。这里要介绍的是对于较大型深度神经网络的验证可以提供收敛性保证的方法，也即同时提供针对某个性质成立“上界”以及“下界”，并且证明这两个上下界收敛。

逐层精化 黄等人[8]于 2016 年提出了一个可以自动化验证深度神经网络的基于 SMT 的框架。此框架的主要特征为：首先，它能确保如果一个对抗样本存在，那么它一定能被找到；其次，它采用了逐层精化（layer-by-layer refinement）的方法，也就是说，从神经网络的输入层分析至其隐藏层并且最终可以达到输出层。在此工作中，神经网络的安全性，尤其是（局部）鲁棒性，可以表达为在某个输入点周围的空间中，可能存在的任何对抗扰动都不会对神经网络最终的输出结果发生改变。具体来说，对于输入层或是某

个隐藏层，与其相关的输入向量空间周边的有限区域都可以通过单路径或者多路径的搜索方式来进行穷举探索。至于逐层精化，则是通过 Z3 求解器来实现，其目的在于确保在神经网络中，某个更深的隐藏层的局部鲁棒性可以推导出较浅隐藏层的局部鲁棒性，也就是说如果某一隐藏层是鲁棒的，那么其之前的所有隐藏层也都是鲁棒的。

基于此验证神经网络的框架而开发出来的工具为 DLV，其可以运作在较大型的神经网络上，譬如可以分类 MNIST、CIFAR10、ImageNet 等图片数据集的神经网络。值得一提的是，后来这里提到的穷举搜索也可以使用蒙特卡洛树搜索（Monte Carlo tree search）的方法来实现，并获得更优的实验结果[45]。

双选手博弈 吴等人[46]于 2018 年针对深度神经网络的鲁棒性提出了两个研究问题：一是最大安全半径（maximum safe radius）问题，也即针对某个输入样本计算出它到反例的最小距离，那么在这个最小距离以内所有的对抗扰动都不会产生反例，同时在与原输入样本的距离大于该最大安全半径时，则一定存在某个对抗扰动使得神经网络的分类结果发生变化；另一个是特征鲁棒性（feature robustness）问题，也即针对输入样本譬如某张图片等，来分析和量化其上不同特征（feature）的鲁棒性。

在所考虑的神经网络满足利普希茨连续（Lipschitz continuity）的前提下，这两个问题都可以通过将输入空间离散化，然后使用有限优化的途径来进行近似，并且此近似过程还可以提供可证的保证，也即其中的误差（error）在一定的界限范围内。之后，此优化问题又可以转换成双选手轮流博弈（two-player turn-based game）的最优解，在此博弈中，一位选手选择输入图片上的特征，另一位选手则判断在选定的特征内如何干扰。在博弈过程中，第二位选手的目标是最小化到反例的距离，第一位选手的目标则根据优化目标的不同而进行调整：在求解最大安全半径问题时，该选手是合作（cooperative）的；在求解特征鲁棒性问题时，该选手是竞争（competitive）的。对于此博弈最优解的近似可以通过单调地递增下界并且同时单调地递减上界来完成，其中上界的计算采用蒙特卡洛树搜索完成，下界的计算则是在双选手互相合作时采用 Admissible A * 算法，在互相竞争时采用 Alpha-Beta 剪枝算法。吴等人将此验证算法框架集成在 DeepGame 工具中，并且能够在 MNIST、CIFAR10 和 GTSRB 等数据集上提供上下界逐渐收敛的保证。

基于此工作，2020 年吴等人[47]又将 DeepGame 扩展到了循环神经网络并以此来提供基于视频而不仅仅是图片的收敛性保证。其中，光流法（optical flow）被用于抽取视频中每一帧上的空间特征以及相邻帧之间的时序特征。至此，双选手博弈中一位选手选取不同的光流特征，另一位选手则决定如何干扰被选中的光流。从实验结果层面，这一方法可以提供 VGG16[48]和长短期记忆网络在视频数据集 UCF101[49]上的收敛性保证。

全局优化 2018 年，阮等人[10]证明了现有深度神经网络中大部分的层都是利普希茨连续的，从而在此基础上利用全局优化（global optimisation）提出了针对神经网络的 DeepGO 验证算法。对于输入中的单维度，这一方法通过利用利普希茨常数来计算下界，并且保证下界值最终收敛到最优解。至于输入中的多维度，则是利用单维度上的算法来穷举搜索出最优的维度组合。此方法可以运作在现下的较大型神经网络上，但是受限于被干扰的输入维度的数量。

同年，阮等人[50]针对汉明距离（Hamming distance）提出了量化神经网络的全局鲁棒性（global robustness）的 DeepTRE 算法，其中全局鲁棒性定义为在整个测试数据集上所有样本的最大安全半径的期望。对于神经网络的全局鲁棒性，此方法可以迭代地产生最大安全半径的下界和上界，并且上下界的值随着计算严格收敛最终达到最优解。此外，此算法以批处理的方式，通过利用深度学习中张量（tensor）的概念，以便高效地在 GPU 上进行计算。

2.2.4 验证神经网络的统计性保证

下面介绍一些在验证深度神经网络的过程中可以提供统计性保证的工作。此处统计保证的意思是，利用一个量化的概率值表示性质是否被满足或者某个值是否是下界等问题的结果。

基于极限值理论的利普希茨常数估计 2018 年，Weng 等人[51]提出了一种名为 CLEVER 的度量（metric），并以此来估计利普希茨常数的值。此方法利用极限值理论（extreme value theory）来取样梯度范数，从而对神经网络的鲁棒性性质的下界值限制概率分布。不过，Goodfellow[52]指出这一方法只能找到下界值的估计，因而存在可靠性（soundness）问题。

鲁棒性估计 2016 年，Bastani 等人[53]针对神经网络中存在反例的频率和严重度分别提出了两种对于鲁棒性的统计度量方法，此种方法基于上文引入的最大安全半径问题。此类基于统计的鲁棒性估计是建立在满足局部呈线性的假设上，然而此假设在最大安全半径足够小的时候才成立。除非针对的是激活函数为 ReLU 的神经网络，此时由于存在利普希茨常数从而假设成立[10]。

苏黎世理工学院（ETH）的 Vechev 领导的研究组最早提出了一种基于抽象解释的框架 AI^2，来验证神经网络的安全性和鲁棒性[28]，其主要思想是使用一组带条件的仿射函数来建模基于 ReLU 的神经网络，可以刻画神经网络中的全连接、卷积和 max-pooling 等多种结构，在验证过程中使用区间抽象域、zonotope 抽象域[25]及其幂集抽象域来分析这些仿射函数，最后得到输出层变量的取值范围或变量之间的约束关系。

2.3 人工智能系统的测试技术

与传统软件类似，深度学习模型在部署前也应当进行有效而充分的测试来保障其安全性。人工智能系统测试是一个检测系统异常行为的过程，目的是完善人工智能系统的学习机制，处理可能引起系统错误行为的数据，提高人工智能系统的安全性和鲁棒性。机器学习系统是一种数据驱动的系统，可以根据输入的数据进行在线学习或通过训练好的模型直接识别数据，根据系统的输出行为是否满足需求来判断系统的正确性。机器学习测试包含以下四个步骤：

1）选择合适的数据集作为原始样本集，或根据样本生成算法生成新的数据集和对应的 oracle 集。

2）若需要测试的模型是训练模型，则将第 1 步生成的样本集作为训练样本输入并训练，生成预测模型，用于预测给定测试数据的标签。

3）若需要测试的是已经训练好的预测模型，则将第 1 步生成的样本集作为测试数据，输出测试结果。

4）对比实际输出和第 1 步生成的 oracle，判断测试输出是否错误。若输出有误，则对系统进行修正，重新学习计算并测试修正后的系统；若输出无误，则结束测试过程。

近年来，软件工程领域的学者尝试借鉴传统软件测试的思路将其中一些已被广泛研究应用的知名解决方案（比如差分测试[54]、变异测试[55-56]、动态符号执行测试[57]等）引入到深度学习模型测试中去。大量工作显示测试是暴露深度学习模型缺陷的有效方法，并可通过重训练进一步提高模型的可靠性[54,58]。

接下来，我们从三个方面对当前国际上比较流行的深度学习模型测试方法加以总结，即：现有深度学习模型的测试标准，现有针对深度学习模型生成测试用例的测试技术，以及其他测试技术在神经网络中的应用。

2.3.1 测试标准

为了衡量测试的完成程度，一系列针对深度学习模型的测试指标也被相继提出。这些指标大多是借鉴传统软件测试中的覆盖指标，将神经网络与程序进行类比，把神经元的激活与否看作程序中的条件判断语句。由于神经网络的结构以及测试样本不仅要覆盖神经网络主要功能行为，还要覆盖极端行为（corner-case behaviors）的要求，本文根据覆盖粒度，将神经网络的测试覆盖准则分为三类：神经元覆盖、网络层级覆盖、神经元对覆盖。

神经元覆盖 最早被提出的深度学习模型测试标准。神经元覆盖率指标可以视为软件测试中语句覆盖的一个变种[54]。它将神经元被激活与否作为该神经元是否被覆盖的标准并将整个测试集中被激活的神经元的比例，即神经元覆盖率，作为衡量模型测试程度的指标。这种定义非常简洁，但后来有学者相继发现，该指标在实际中过于容易实现因而对评价模型的被测试程度并没有太多指导意义。例如，从 MNIST 数据集中随机提取 25 张图片即可达到 100% 的覆盖率。

网络层级覆盖 神经网络中比较活跃的神经元决定了神经网络的主要功能，因此测试集应该尽可能多地覆盖活跃度较高的神经元。鉴于神经元覆盖率的局限性，马等人在其工作 DeepGauge[58] 中提出网络层级的 top-k 神经元覆盖准则。相比于简单观察一个神经元是否被激活，该系列指标对神经元被激活时的取值及层间神经元的激活模式进行了细化总结，因此有助于更系统地评价模型的被测试程度。具体地，在神经元覆盖层面，进一步考察每个神经元输出值的分段覆盖情况以及边界覆盖情况作为测试标准；在神经层覆盖层面，把每一神经层上最经常被激活的神经元以及它们在行为上的一些共性进行提取作为测试标准。

神经元对覆盖 另一个基于神经元覆盖的扩展指标是 MC/DC[57]。将第 $l-1$ 层的神经元作为条件，第 l 层的神经元作为对应的判定，描述上层神经元对下层神经元的影响，

即上层神经元符号或距离的改变会引起下层神经元符号或值的改变，本质上是为了描述神经元的改变对神经网络最终输出的影响。MC/DC 覆盖指标关注条件覆盖，直观上讲是说如果一个神经元被覆盖，那么所有可能影响到该神经元激活情况的所有神经元（激活与否）都应当被测试所覆盖。

除了以上评价整个测试集的指标外，还有一些工作通过某些表征对单个测试用例的价值予以评定[59]。例如，近来有学者提出应当考虑测试用例对于训练数据的新奇度，而一个好的测试集应该涵盖不同新奇度的样本。据此，他们基于神经元激活情况定义了一些距离指标来衡量测试用例的新奇度。在神经元覆盖指标外，也有一些试图将输入空间通过各种角度进行划分并评价测试集对输入空间分段覆盖的工作。例如，有学者提出将输入空间按照神经元的激活模式划分为不同的超矩形并试图覆盖所有可能的超矩形[45]。

2.3.2 测试技术

神经网络测试技术的主要手段是针对当前模型生成针对提升某一测试指标更加有效的测试用例，添加所生成测试用例到原先训练数据集进行模型的重新训练，以期来增强最终训练出神经网络的安全性和鲁棒性。

变异测试 南京大学陈等人[60]提出了模型层面的变异算子，对神经网络的结构直接进行变异操作生成变异神经网络，用于评估神经网络测试数据集的完整性。哈尔滨工业大学马等人[55]考虑到引起系统错误行为的因素除了神经网络，还包括训练数据和训练程序，因此提出了源级变异算子，对训练数据和训练程序进行变异操作，重复训练过程，训练出多个变异神经网络。利用测试数据集测试变异神经网络和原始神经网络，通过对比输出行为的差异来评估测试数据集的质量。

蜕变测试 蜕变测试是根据多个输入输出之间的关系是否满足，来判断被测软件的正确性。利用该方法，DeepTest[63]将蜕变测试应用在基于深度神经网络的自动驾驶系统上，主要用来测试深度神经网络是否对同样场景不同天气条件的情况输出相似的转向角，即自动驾驶系统不受天气因素的影响或影响较小。周等人[61]认为增加非兴趣区域的数据点不会造成兴趣区域障碍物无法检测的情况，并将该需求作为激光雷达障碍物感知系统的蜕变关系。实验结果显示感知系统受非兴趣区域噪声点的影响，存在道路障碍物检测丢失的问题，且噪声点越多，感知系统性能受影响越大。

测试用例生成方法 有一系列针对覆盖指标生成测试用例的方法。例如，文献［54］结合差分测试，通过求解联合目标优化问题生成测试用例以使得多个神经网络判别出现分歧同时尽可能提高神经元覆盖率。结合模糊测试，文献［62］通过对覆盖率的反馈分析优化挑选种子池来不断生成测试用例提升 DeepGauge 覆盖率系列指标。文献［57］则提出用带量词的线性算术来描述测试要求，并结合遗传算法设计了针对神经网络的动态符号执行算法来产生测试用例以提升神经网络针对给定测试要求的覆盖。

另外，大量实验也表明简单地通过各种各样的对抗式攻击所产生的攻击样本也可以提高大多数测试指标[58]。常见的攻击方法包括 FGSM、BIM、JSMA、CW 等（详见第 5 节）。基于规则的随机变异也是生成测试用例的有效方法。例如，文献［63］通过对原

始图片施加一系列现实场景中可能出现的变换操作来产生测试用例，提高了神经元覆盖率并发现了原神经网络在这些变换后图片的大量错误判断。类似地，也有学者提出用对抗式生成网络从正常训练样本中产生更加实际的测试用例[64]。实验表明，神经网络测试技术在提升神经元覆盖率的过程中可以有效产生大量对抗样本用于模型的重新训练，一定程度上提高了神经网络的鲁棒性。

除上述测试方法之外，一些新的测试技术为神经网络测试提供了不同的角度和思想。例如，文献［56］发现攻击样本在输入和模型扰动下相比正常样本有更高的敏感度，因此提出利用此特点并结合变异测试从输入样本中检测攻击样本。与之前专注于模型的安全性和鲁棒性不同，文献［65］提出一种针对模型公平性的测试技术来找到数据中可能导致模型偏见的样本点来提高模型的公平性。

2.4 人工智能系统的可解释性

以深度学习为代表人工智能模型的构建以及训练往往可以用相对简单的代码完成，但其产生的结果却极其复杂，使得即使是专家也难以理解一个人工智能模型是如何判断并产生输出结果的，无论这个输出正确与否。于是，包括提取出输入数据的重要特征在内的人工智能可解释性研究成了帮助人们理解人工智能工作原理的有效手段。对可解释性的需求是人工智能的挑战也是特性，针对人工智能系统的形式化验证工作应当合理利用或结合其可解释性的研究成果，甚至主动应用到可解释性的场景中去。

人工智能的可解释性是近年来提出的一个概念，指人工智能系统决策机制能够被人类理解的程度[66]。可解释性又分为全局和局部可解释。全局可解释性指模型的整个逻辑可以被理解，并遵循整个推理系统推导所有不同的结果。相反，如果只有单个或者特定的决策是可解释的，则称为局部可解释性。影响可解释性的因素还包括用户可用的时间或允许用户花在理解解释上的时间以及所需要的背景知识和经验。

2.4.1 特征排序

目前主要的特征排序方法可以归为两大类：基于反向传播（backpropagation based）和基于输入扰动（perturbation based）。以深度神经网络为例，基于反向传播的方法始于神经网络的输出值，输入特征对神经网络输出的影响通过从最后一层到输入层的（逐层）分析估测出来。不同于反向传播，基于输入扰动的特征排序方法通过改变部分输入特征来改变后续神经元的激活状态以及神经网络最终输出值，进而估测出输入特征的重要性。两种特征排序方法又往往都依赖于启发式算法来实现。

Simonyan 等人[67]针对 ConvNet 模型的图片分类可视化工作是反向传播方法早期的典型代表。其中，ConvNet 图像分类器在一个具体输入点附近的行为被近似为一个线性方程，这个线性方程的参数是通过对网络在这个具体输入点上进行求导获得的。最终，这个线性近似模型上每一个像素点相应的参数值也即是这个像素点对神经网络分类这个输入的贡献。以后的基于反向传播的特征排序工作也多沿袭了对机器学习模型或者其一部

分的函数求导。周等人[68]找到了一种了针对在最终输出层前拥有 Global Average Pooling，并且不包含全连接层的卷积神经网络的后向传导方法，并用以来计算输入特征对每一类输出的重要性。GradCAM[69]将文献［68］中的方法拓展到包括最后一层是全连接层在内的更一般的网络结构。Selvaraju 等[70]通过泰勒多项式展开的方法从输出层开始逐层计算每个神经元对输出分类结果的贡献直到组成一个输入的像素点。泰勒多项式展开可以应用于任意可导的神经网络激活函数。给定一个输入，DeepLIFT[71]通过将其与一个参考输入对每一层神经元激活状态的比较来反向传播并获得不同输入特征对神经网络分类结果的贡献。参考输入的选取涉及对具体问题的分析。Sundararajan 等[72]提出了两条特征排序方法应当满足的性质：1）敏感性，也即如果当前输入和参考输入只有一个不同特征，但是他们造成了神经网络不同的输出分类，那么这个特征的重要性不应为 0；2）恒定性，也即如果两个神经网络等价（给定相同的输入，它们永远产生相同的分类结果），那么针对他们的特征排序结果也应该相同。这两个性质也被认为是 DeepLIFT[71]等方法的不足之处。文献［72］进一步提出了相应的反向传播启发式特征排序算法来满足这两个条件。更多的对特征排序结果的合理性检查可以参考 Adebayo 等人的工作[73]。

与上文提到的只需要一次反向传播的特征提取方法不同，基于扰动的方法通过反复采样输入空间来寻求更高的准确度。LIME[74]是基于输入扰动的特征排序的诸多方法中很典型的一种。给定一个输入，LIME 通过覆盖（mask）一部分输入特征来在给定输入点附近进行采样，并通过分析这些样本与机器学习模型分类结果是否变化之间的关系来构建这个机器学习模型在采样点附近的线性行为模型，然后以此对输入特征的重要性进行排序。在文献［75］中，选定的若干输入特征的自然分布被自定义的分布所取代，通过量化新的特征分布对机器学习模型输出的影响，相应选定特征的重要性可以被估计出来。特征之间的关联性使得在自定义特征分布时需要考虑不同特征的组合，最终博弈论中经典的 Shapley Value 理论被用来对特征进行排序。Lundberg 等人[76]发现文献［71，74-75］中的特征排序都可以放在一个统一的理论框架下，首先用一个线性模型近似原有的复杂的机器学习模型，再应用 Shapley Value 的理论来求解特征之间的线性组合。L2X[77]通过测量输入特征变化与输出变化之间的互信息（mutual information）来计算每一个特征的重要性。Fong 等人[78]引入了针对扰动区域的限制条件，其定义的极值（extremal）扰动，也即基于特定区域的扰动并获取的最大化的输出变化，有效改善了重要特征的提取。软件的思想也在人工智能的可解释性上也有所体现。Galhotra 等人[79]考虑离散化的输入特征，通过统计基于软件测试方法生成的输入集和来量化特征的不同离散值对输出的影响。孙等人[80]将软件工程中错误定位的方法应用到特征排序中，对神经网络输出起到重要作用的特征被当作需要定位的“错误”来对待，并取得了优于传统特征排序方法的最新结果。

2.4.2 量化追踪

针对深度学习的可信性研究，其中一个重要的趋势是，理解深度神经网络 DNN 的训练过程，进而理解训练后的模型泛化性能，从而确保深度学习模型的可操控性。在深度

神经网络中，端到端的训练过程是一个复杂的动态过程，涉及大量参数优化，从微观的角度分析（如参数选择，损失函数优化）可能并不能满足模型泛化需求。最新的研究发现，通过对训练过程中 SGD 所导致的训练动态的宏观角度分析，特别是信息流在各个隐藏层表示的统计特性的追踪，是理论理解深度学习的训练过程的关键。

信息论作为研究信息的基础理论被引入深度学习理论，用来量化训练过程中数据流的信息量的变化。信息论是建立在统计的角度上的对信息的宏观量化，其中信息熵被广泛用来量化信息的不确定性，互信息被用来衡量信息之间的相关性。用信息论工具来量化深度学习的训练过程，起源于以色列理工的 Tishby 团队 2015 年和 2017 年的开创性工作[81-82]。其主要思想是将 DNN 的训练过程放入信息瓶颈（information bottleneck）的框架之中，把 DNN 的学习过程和泛化过程用两个互信息量来衡量。信息瓶颈理论最先是 Tishby 团队于 1999 年提出[83]，基本理论建立在信息论的信源编码理论基础之上。在过去的二十年里，信息瓶颈理论得到广泛的发展，包括各种变形和参数化，并应用于文字聚类（word clustering）、文档分类（document classification）、文本分割（text segmentation）等。

在深度学习的训练过程中，信息瓶颈的主要思想是[81-82]：1）将输入 X，输出 Y，和各个隐藏层表示 Ti 分别建模成多维随机变量；2）中间层 Ti 作为输入 X 的压缩表示，二者之间的相关性，互信息 I(X；Ti)，作为泛化能力的度量；3）中间层 Ti 作为输出层 Y 的关联表示，二者之间的相关性，互信息 I(Ti；Y)，作为学习能力的度量；4）DNN 训练过程可以建模成学习能力和泛化能力的折中优化问题。在信息瓶颈框架下，深度学习的训练过程显现两个阶段：1）学习阶段（fitting phase），I(X；Ti) 和 I(Ti；Y) 同时随训练过程增加，表明中间表示层作为媒介学习输入特征并传送给输出，从而提高了训练精确度；2）压缩阶段（compression phase），I(Ti；Y) 持续增加，但 I(X；Ti) 逐渐减少，表明中间层逐步剔除输入的无关信息，提取和输出相关的特征。最后，Tishby 等断言，正是由于漫长的压缩阶段的特征提取过程，深度学习才拥有了优越的泛化能力。

然而，信息瓶颈理论是否是深度学习的基本理论，受到了来自哈佛大学的 Saxe 等研究人员的挑战[84]。根据信息瓶颈框架在更复杂的数据集的观测结果，他们认为压缩阶段并不一定存在，而且泛化能力与压缩阶段并没有直接联系，因为实际中泛化能力好的模型不一定有压缩阶段。Saxe 等人[84]断言，激活函数在深度学习的信息瓶颈理论中至关重要，只有使用双边的激活函数（如 tanh）的深度神经网络的训练过程才会有压缩阶段。

对深度学习的信息瓶颈理论的争议来自对互信息量 I(X；Ti) 的不同估计方法。来自麻省理工学院的 Goldfeld 等[85]研究人员指出，之前研究所观测的压缩现象其实并不是源于 I(X；Ti) 的减小，因为根据信息论，I(X；Ti) 的理论值在确定性 DNN 中不会减小，而是源于人为引入的信息量的估计方法的动态性（dynamics），如 Binning 方法 Bin 的尺寸随网络层变化。虽然这种人为引入的动态性改变了深度模型，但是实验证明这种动态性的引入有利于信息量的估计，进而对训练过程的解释。Goldfeld 等人[85]通过在神经元加白噪声引入人为的统计动态性，对这种加噪后的模型进行更为精确但比较繁复的信息估计和量化，信息压缩的现象在更多的神经网络模型中可以被观测到。

来自利物浦大学的 Jin 等人[86]研究人员认为，信息量 I(X；Ti) 的估计精度并不是压缩现象的关键要素，对香农熵 H(Ti) 的量化追踪对深度学习的泛化性能更有指导意义。Jin 等人指出，现有的基于距离的信息量或香农熵的估计或量化（如 Binning 方法）没有考虑多维随机变量的自相关性，而且已有方法对中间隐藏层的信息追踪并没有设置公平比较的基准。针对这两个问题，Jin 等人[86]提出了核矢量量化（kernelised vector quantisation）方法，其主要思想包括：1）建立高维核空间作为统一的码本（codebook）空间，使得多维随机变量的统计相关性在高维线性空间里近似统计独立；2）运用核方法（kernel method）把各个隐藏层表示（hidden representation）投影到高维核空间，并对投影表示进行矢量量化，映射到码本的具体码字（codeword)；3）使用现有的香农熵估计方法，对矢量量化后的码字进行熵估计。通过核矢量量化方法，中间隐藏层的自相关性和信息度量的公平性得到了解决。实验表明，决定信息量或香农熵减小的因素，不大可能是激活函数的选择，而更有可能是在隐藏表示中各个元素之间的相关性：更小的相关性预示更好的泛化能力。

2.5 人工智能系统建模

虽然神经网络在很多应用领域中都展示了它具有良好的性能（效率高、准确率高等)，但是，神经网络的一个主要缺点是神经网络没能对它内在的推理机制提供一个很好的解释或者说明，这很大程度上限制了它的应用和推广，尤其是在安全攸关方面的应用。大多数研究者认为主要的原因是目前缺乏相应的技术理解神经网络的决策过程。此外，对神经网络所学习到的知识及神经网络系统本身如何进行形式化建模也依然处于探索阶段。为此，研究者们提出了各种技术来对提取神经网络系统进行的知识萃取或形式化建模，包括有限自动机、规则集合、决策树和程序等。

由于有限自动机与神经网络（尤其是 RNN）之间存在着比较大的联系或者相似之处[87]，大多数研究工作采用有限自动机㊀来表示神经网络的知识萃取或模型。在 20 世纪 90 年代早期，研究者们就开始尝试从处理序列数据的 RNN 中提取自动机。一般来说，从 RNN 中提取自动机的技术可以归纳为以下四个步骤[88]：

1）量化：神经网络（比如 RNN）连续状态空间的量化。

2）状态生成：根据输入生成可能状态和输出以及其分类（如果需要的话)。

3）规则构造：基于观察到的状态迁移构造规则。

4）规则最小化：对规则集合进行合并优化。

早期的自动机提取技术主要采用层次聚类分析（Hierarchical cluster analysis）来分析神经网络的连续状态空间[89-91]，但是这种方法可能不易找出类与类之间的（临时）关系。随后，一些研究者提出将状态空间均匀地抽象成 n 个超立方体（即宏观状态)，然后采用深度优先搜索策略来搜索神经网络的可能状态集[92-94]。然而，这种技术最大的问题

㊀ 从 RNN 中提取的规则几乎都表示为有限自动机，因此这类工作在此也归并为自动机方面的工作。

在于聚类数目会随着状态节点数的增长而指数增长。有些研究者基于向量的量化来对状态空间进行分类。曾等人[95]提出了基于 k-means 的宏观状态聚类方法，类似的工作还有文献［96-98］。但是，为了支持合适的状态聚类，这种方法需要在训练的时候引入一些偏见（bias）。为此，Alquezar 和 Sanfeliu 采用早期的层次聚类分析，并结合前序树对状态空间进行剪枝[99-100]。

除了对状态空间进行搜索，有些研究者提出了基于采样的方式来提取自动机。Watrous 和 Kuhn[101]提出了第一个基于采样的方法，该方法在处理状态空间的同时进行采样，并动态地更新每个宏观状态的区间。Manolios 和 Fanelli[102]使用一个简单的向量量化器，且对给定的测试集进行状态空间采样，但不能保证该过程的终止性。类似的，Tino 和 Sajda[103]提出的方法也是对测试集的状态空间进行采样，不同的是他们的方法采用星拓扑自组织映射（self-organizing map）[104]来量化状态空间。但是，这些基于采样的方法可能存在着宏观状态的不一致性（即不确定性）问题，从而导致提取失败。为了解决这个不一致性问题，Schellhammer 等人[105]引入迁移频率概念且忽视那些最小频率的不一致性迁移。

解决不一致性问题的另外一个方案是概率自动机[106]。Tino 和 Vojtek[107]提出了一个从 RNN 提取概率自动机的方法。该方法采用文献［103］的自组织映射来量化空间，同时结合域驱动（根据输入观察输出）和自驱动（上次的输出作为下次的输入）来生成状态空间。随后，Tino 等人[108-109]提出了一个改进方法，将其中自组织映射改为动态细胞结构（dynamic cell structure）[110]。然而，使用概率自动机的弊端是难以找到所提取概率自动机与网络之间的联系。最近，Rabusseau 等人[111]提出了基于光谱学习（Spectral Learning）的权重自动机提取方法，其中权重自动机是概率自动机的一种一般化。

上述所提的方法几乎都属于白盒方法，就是说，这些方法需要分析神经网络的结构及其内部状态。一些研究者们提出了一些不需要了解神经网络的结构及其内部状态的黑盒方法。Vahed 和 Omlin[112-113]关注限定长度的输入及其输出，提出基于机器学习的自动机提取方法，但其前序树的复杂度比较高。Weiss 等人[114]采用了 L * 算法[115]和抽象技术从 RNN 中学习出有限自动机。该方法只能够有效地应用到字母表比较小的正则语言。Mayr 和 Yovine[116]提出了基于有限界 L * 算法的自动机提取算法，且该算法能够保证所提取的自动机符合 ε-近似正确。最近，Ayache 等人[117]提出了一种从应用于序列数据的黑盒系统中提取权重自动机的学习方法。基于 L * 算法的扩展算法，Okudono 等人[118]提出了权重自动机的学习算法。不同于 Ayache 等人的方法，Okudono 等人的方法充分利用内部状态来完成等价查询。

随着人工智能（特别是深度学习）的发展，一些研究者关注结构比较复杂的新型网络。王等人[119]将基于 k-means 的方法应用到三个新型网络中，即 LSTM（long-short-term-memory networks），GRU（gated-recurrent-unit networks）和 MI-RNN（multiplicative integration recurrent neuron networks）。Koul 等人[120]关注应用于强化学习和模仿学习的 RNN 策略网络，并引入量化瓶颈插入技术（quantized bottleneck insertion）来从 RNN 策略网络中提取摩尔机网络。Ikram 等人[121]提出了基于 k-means 从 LSTM 中提取自动机

的方法。Lu 和 liu[122]提出了一种基于注意力抽象的方法，从 DOB-net 中提取有限自动机，即关键摩尔机器网络（Key Moore Machine Network），以捕获其控制转换机制。

近两年来，一些研究者们试图从形式语言的角度来理解 RNN，比如比较 RNN 与有限自动机的状态或计算能力，和从 RNN 中提取形式语言。William[123]通过将神经网络关联到有限自动机来解释神经网络的计算能力。Joshua 等人[124]分析和比较 RNN 在识别正则语言时的内部状态与接受该语言的最小有限自动机（MDFA）的状态的关联关系，发现 RNN 的内部状态可以映射到 MDFA 的超状态。Wang 等人[125]试图将具有不同阶隐藏交互的 RNN 与不同复杂度的正则文法关联起来。Christian 等人[126]从精确性和可解释性分析了用正则语言训练的简单 RNN 的行为，他们发现适当的调整参数能使网络同时具备较强的泛化能力和可解释为有限状态自动机。Reda 等人[127]从理论上研究了从 RNN 中提取有限状态自动机的一些性质。Bishwamittra 和 Daniel[128]结合了近似可能正确（Probably Approximately Correct）和约束求解，提出了从 RNN 中提取线性时序逻辑（LTL）的方法，以解释 RNN 的决策过程。

此外，还有一些研究工作从网络中提取符号化规则[129-133]、决策树[134-139]和逻辑程序[140-141]等。

2.6 对抗攻击与形式化验证

攻击技术在于给缺少理论保证的和鲁棒性的深度神经网络提供实例证明（如对抗样本）。与对抗技术的相对应的是防御技术，其可以通过提高深度神经网络的鲁棒性来减少对抗样本的数量，或者可以从矫正输入中去除对抗样本。我们分析了不同的攻击技术，然后将这些技术和一些验证方法相比较，最后给出总结。

2.6.1 对抗攻击

基于给定输入，一个对抗攻击（也称为攻击者）试图找到一个扰动或者失真，使得对于同一个训练好的深度神经网络将叠加这个扰动或者失真的输入错误分类。一般而言，这要求对抗样本有很大的概率是被错分。

根据对抗样本被错误分类的不同，可以将对抗技术粗略地分为两类：

1）目标扰动：攻击者能够控制被错分的类别。

2）无目标扰动：攻击者仅仅实现了错误分类，但是不能控制被错误分的类别。

根据攻击者获取的信息量，对抗扰动也可以被分为两类：

1）白盒扰动：攻击者需要获得训练好的深度神经网络的参数和内部结构，可能也需要获得训练数据。

2）黑盒扰动：攻击者仅仅知道训练好的深度神经网络的扰动输入，但是不能获得深度神经网络的内部结构和参数。

另外，根据衡量原始输入和扰动输入不同的范数距离的定义，可以将对抗攻击分为 ℓ_0，ℓ_1，ℓ_2，或者 ℓ_∞ 攻击。需要注意的是，所有的扰动都可以用其中任何一种范数来衡

量，但是攻击技术产生更适合特定范数的对抗样本。

现存的大多数攻击技术主要是针对计算机视觉方面的对抗样本。最近来产生对抗样本的多种技术已经出现。概括来讲，这些攻击能够从一个技术的观点上来进行分类，如损失梯度[142-143]，或者神经网络的前向梯度[144]，或者沿着最有可能方向的扰动，或者直接用解决一个优化问题来发现扰动（可能使用梯度下降/上升）[145-146]。

另外，对抗样本具有在不同的网络结构之间转化的性质，同时也可以在深度神经网络上训练不同的子数据集[7,144]。对抗样本的在现实世界中也展示出了抗议转换的特性[147]。特别的，对抗图像仍然有可能被错误分类，即使被手机摄像头重新捕获或者打印。接下来，我们将仔细分析几个比较代表性的工作。

有限存储的 BFGS 算法（L-BFGS） 文献［7］指出来对抗样本的存在，同时将它们描述为深度神经网络的盲点。对抗样本是被深度神经网络错误分类，并且出现在它周围的是正确分类的样本。值得注意的是，因为对抗样本出现的概率很低，它们不可能通过正确分类样本的采样而有效地获得。然而，对抗样本可以通过最优化的方法获得。例如，假设我们有一个分类器 f：$\mathbb{R}^{s_1}\rightarrow\{1,\cdots,s_K\}$，它将输入映射到 s_K 中的一个类别输出，给定一个输入 $x\in\mathbb{R}^{s_1}$，一个 $t\in\{1,\cdots,s_K\}$，但是 $t\neq\arg\max_l f_l(x)$ 中的输出标签，为了发现一个额外的对抗样本 $x\in\mathbb{R}^{s_1}$ 满足下面的优化表达：

最小化 $\|r\|_2$：

1. $\operatorname{argmax}_l f_l(x+r)=t$

2. $x+r\in\mathbb{R}^{s_1}$

因为精确计算是困难的，所以用基于 L-BFGS 的算法来代替。而且，文献［7］指出对抗样本是多样的。根据上述架构，一个网络可以产生无限个对抗样本。他们也注意到这些对抗样本一个根据模型或者训练数据集来产生。一个从 DNN 分类器中产生的对抗样本将会类似于另外一个不同结构或者数据集的分类器产生的对抗样本。

FGSM Fast Gradient Sign Method[142]能够找到一个对应于特定范数 ℓ_∞ 约束是对抗扰动。FGSM 是对每个像素值进行逐步修改，所以损失函数在特定的 ℓ_∞ 范数下是增长的。其作者认为这个方法也提供给对抗样本一个线性解释。他们指出既然单独的输入特征的精度是非常有限的，例如图像通常是用 8 位字节来表示每个像素，因此它的精度最高是 1/255，对应不同输入的一个分类器，如果它们仅仅在特征的数量上面不同，而数量又是低于精度水平的，那么这个分类器是不合理的。然而，考虑到权重向量 w 和对抗样本 $x'=x+r$ 之间的点积，

$$w^{\mathrm{T}}x' = w^{\mathrm{T}}x + w^{\mathrm{T}}r$$

当 $r=\varepsilon\operatorname{sign}(w)$，激活增长通过这种方式最大化。如果 w 是 n 维的向量，其平均幅值是 m，则激活增长是 εmn，即对应维度问题的线性增长，当 $\|\eta\|_\infty$ 保持小于 ε。因此，对于高维度的问题，FGSM 能够使得微小变化的输入产生一个很大不同的模型输出。基于线性解释，文献［142］建议一种更快的算法去产生对抗样本。用 θ 作为模型参数，x 作为模型输入，y 是对应于 x 的输出标签，训练模型的损失函数 $\mathcal{J}(\theta,x,y)$，对抗扰动 r 可以由下面的式子产生：

$$r = \varepsilon \text{sign}(\nabla_x \mathcal{J}(\theta, x, y))$$

越大的 ε 使得被攻击的概率越大，但是这个结果和人类的视觉大不相同。这些攻击方法已经扩展到一些有目标的攻击和迭代攻击中[147]。

JSMAPapernot 等人[144]呈现了一种基于 DNN 前向积分的算法，定义对应输入维度的输出在类别集 $\mathcal{L}$ 上的概率分布的雅可比矩阵，这个用来增强这些 DNN 预测敏感的特征，当有扰动的时候很容易产生误分类。对于给定类别 $c \in \mathcal{L}$ 和输入 $c \in [0, 1]^{s_1}$，输入的每一维都会分配一个基于前向微分的显性值。每一个输入维度的显性值能够捕获输出分配给类别 c 的概率大敏感性。

对于对抗扰动，具有最大显性值的输入维度具有最大的失真参数 $\tau > 0$。如果扰动导致误分类，那么算法就会终止。否则，前向导数通过失真的输入被重新计算，算法再被执行。当最大距离阈值 $d > 0$ 达到的时候，算法也会终止。这个算法不需要计算基于 ℓ_p 范数的扰动的导数，但是能够产生基于 ℓ_0 范数的对抗扰动。这个方法通常是比 FGSM 慢一些，其目标主要是发现基于 ℓ_0 范数距离的对抗图像，对应于原始图像。

DeepFool 文献［145］提出一种迭代的算法来产生无目标的对抗样本通过最小化 ℓ_p 范数（$p \in [1, \infty)$）。首先，他们考虑产生一个对应于仿射二分类器 $g(x) = \text{sign}(w^T \cdot x + b)$ 的对抗样本。在这种情况下，对应给定输入图像 x_0 的对抗样本可以通过 x_0 在超平面 $\mathcal{F} = \{x \mid w^T \cdot x + b = 0\}$ 上的正交映射进行解析计算。这个模型可以扩展到多分类的情形，$W \in \mathbb{R}^{m \times k}$，$b \in \mathbb{R}^k$，让 W_i 和 b_i 分别表示 W 和 b 的第 i 个成分，我们有下面公式：

$$g(x) = \underset{i \in [1 \cdots k]}{\text{argmax}} g_i(x), \text{where} \quad g_i(x) = W_i^T x + b_i$$

现在，为了寻找最优的对抗样本，将输入 x_0 映射到最近的超平面 P，定义如下：

$$P(x_0) = \bigcap_{i=1}^{k} \{x \mid g_{k_0}(x) \geqslant g_i(x)\}$$

当 $k_0 = g(x_0)$。换句话说，P 是一个对应于所有输入，其都被分类为与 x_0 相同类别的集合。为了归纳这些非仿射多分类器即深度神经网络，最优的对抗样本是通过迭代找到的，其中的每一步对抗样本都能通过线性近似分类器及上述描述的仿射映射来更新。尽管这是一个贪心的启发式算法，并不能保证能够找到最优的对抗样本，但是扰动是越来越小的，被认为是一个很好的最优近似。

C&W 该攻击是一种基于对抗攻击的优化方法[146]，它把对抗样本的搜索问题变成图像距离最小化的问题，例如 ℓ_0，ℓ_2，ℓ_∞ 范数问题。尤其是，它定义了基于损失函数的优化问题：

$$\ell(v) = \| v \|_p + c \cdot f(x + v)$$

其中 f 是一个函数，当 DNN $\mathcal{N}$ 对于 $x + v$ 的有效输入错误分类的时候，f 是一个负值。这个优化问题使用 Adam[148]梯度下降的方法解决。这种方法可以应用在 ℓ_0，ℓ_2，ℓ_∞ 三种不同的距离标准上，对算法的每一个响应做微小的调整。尤其是对于 ℓ_0 的情况，一个迭代的算法用来识别对分类影响比较小，因此不会被认为是扰动的子特征集，这些子特征集随着每次迭代而不断增加，直到它的补集变得足够小，给出一个对分类敏感的非常小的特集。每次迭代，被排除的特征 i 是一个最小的 $\nabla f(x + v)_i \cdot v_i$。一个 C&W 攻击中的小的

技巧是它引入了一个新的优化变量，用来避免盒子约束（就是图像像素需要在［0，1］内）。这种方法直觉上是比较类似于文献［144］，其将基于一阶导数的显著性特征子集区分开来，而文献［146］展现出一个比上述方法更有效的方法。C&W 攻击能够找到对抗样本，那些有非常小的图像距离，尤其是基于 ℓ_2 范数的情况下。

2.6.2 对抗攻击通过自然转换

除了上述能够发现像素级别的对抗攻击的方法，通过自然转换来挑选对抗样本的研究也已经出现了。

旋转和平移 文献［149］指出现在存在的很多产生对抗样本的对抗攻击的技术都是精心设计、不是自然发生的。它展示出 DNN 很容易受到样本的攻击是发生在没有更多设计的集合中，例如旋转或者平移输入图像，目标分类器的性能会衰减。从技术上来看，文献［149］目标是寻找一幅给定图像的最优旋转角度和平移幅值，它允许的误分类范围是（$\pm 30°$）×（± 3）个像素。一些新的方法已经被提出，包含：（i）使用 DNN 的损失函数的导数的一阶迭代方法；（ii）网格搜索的方法，通过离散化参数空间和无限的测试所有可能；（iii）k 个最差的方法，通过随机的采样 k 个可能的参数值，并选择使得 DNN 的性能最差的参数值。

空间转换的对抗样本 文献［150］提出一种通过空间变换而不是直接改变像素值的改变像素定位的方法来产生实际的对抗样本。这些空间变换是通过一个转移场或者流场来定义的，通过定义每一个像素的位置来取代一个新位置的像素值。通过二线性插值技术，得到的对抗输出是不同于相对应的流场的。这有助于用优化方法来产生对抗流场。从技术层面讲，文献［150］引入了距离衡量方法 $\mathcal{L}_{fow}(\cdot)$ 而不是用 ℓ_p 距离去捕获局部的几何损失。流形的产生采用类似于文献［146］中提出的优化问题的方法，损失函数通过平衡对抗损失函数和流形损失 $\mathcal{L}_{fow}$ 函数定义。文献［150］通过人类的感知研究展示了空间转换对抗样本图像和文献［142，146］中产生的对抗样本图像相比较，原始图像的分辨对人类来说更加困难。

更加实际的机器学习验证 计算机视觉系统的实例（VeriVis）。裴等人[151]为了评估 DNN 等的鲁棒性，提出一个带有 12 个实际转换的通用模型（VeriVis），分别是均值平滑、中值平滑、腐蚀、膨胀、对比度、亮度、闭合、旋转、修剪、尺度化、平移和映射。每一个变化都是通过一个多项式域的关键参数定义。对所有的输入都穷尽这些变换以验证给定的 DNN 的鲁棒性。VeriVis 是通过测试大量的当前流行的分类器，他们的结果显示所有分类器都存在一定数量的安全隐患。这也说明，相对比其他的基于梯度的对抗技术，VeriVis 有能够产生更多的干扰的能力。

2.6.3 输入无关的对抗攻击

上述方法等一个关键特征是对抗扰动都是对应特定输入产生的，因此不能应用于其他的输入。所以那些与输入无关的扰动将会是更强有力的工具。

广义的对抗扰动 文献［143］定义了广义的对抗扰动（UAP）因为这些是对任何

的样本输入都有很高的错误分类率。当用 v 表示当前的扰动，这个算法通过对输入分布 X 的输入采样子集 X 的迭代。每次迭代中，对于 $x_i \in X$，通过如下的过程不断更新它的扰动 v，首先，它发现对应 ℓ_2 范数的最小的 Δv_i，使得 $x_i + v + \Delta v_i$ 是被 DNN$\mathcal{N}$ 误分类的。一旦计算完成，$v + \Delta v_i$ 是映射到半径为 d 的 ℓ_p 范数球，来确保扰动是非常小的，即：

$$v = \arg\min_{v'} \{ \| v' - (v + \Delta v_i) \|_2 \}, \quad \text{subject to} \| v' \|_p \leqslant d$$

这个算法一直持续到采样样本集的数值误差足够大，即不大于 $1 - \delta$ 对于一个提前设定的阈值 δ。寻找最小的 Δv_i 的优化问题通过 DeepFool 算法[145]来解决。

可以生成的对抗扰动 文献［152］通过训练通用的对抗网络（UAN）扩展了在文献［146］中的方法，并且产生了输入无关而不是针对特定输入的扰动。给定一个最大的扰动距离 d 和 ℓ_p 范数，一个通用对抗网络 U$\mathcal{U}_\theta$ 从正态分布中采样随机输入向量 z，输出一个原始扰动 v，它被尺度化为一个参数 $w \in \left[0, \frac{d}{\| v \|_p}\right]$来得到 $v' = w \cdot v$。

新的输入 $x + v$ 需要重新被 DNN 网络 $\mathcal{N}$ 检查来确定它是否是对抗样本。参数 θ 是通过损失函数的梯度下降算法来学习到的，类似于文献［146］中的方法。

文献［153］采用类似于文献［152］中的方法来产生通用对抗扰动。一个随机的噪声被加入 UAN 中，原始的输出被尺度化到满足一个 ℓ_p 约束，然后叠加到输入数据中，经过修剪，最后被加入一个训练好的分类器中。他们的方法和文献［152］的方法在两个方面是不同的，第一点：他们测试两种 UAN 的结构 U-Net[154]和 ResNet[155]产生器，并发现大多数情况下 ResNet 是优于 U-Net 的。第二点：他们采用了一种多目标分类器的方法来训练 UAN，目的是让产生的 UAP 是被显性的训练能够去迷惑多分类器。这里多分类的损失函数是这么定义的：

$$l_{multi-fool}(\lambda) = \lambda_1 \cdot l_{fool_1} + \cdots + \lambda_m \cdot l_{fool_m}$$

其中 m 是目标分类器的数量，l_{fool_i}是属于目标分类器 i 的损失。而且，λ_i 是给优先目标分类器的权重参数，比如，对于那些很难迷惑的分类器一般设置很高的权重。

2.6.4 对抗攻击技术的总结

表 1 分别从五个方面（距离标准、有无目标、获得信息（即模型结构/参数、逻辑、输出信任度和标签、测试数据集和核心方法），总结了存在的对抗攻击方法的主要相似点和不同点。

表 1 主流的对抗攻击方法比较

	距离标准	有无目标	获得信息	测试数据集	核心方法
L-BFGS[7]	ℓ_2	无目标	模型参数	MNIST，CIFAR10	L-BFGS
FGSM[142]	ℓ_∞	无目标	模型参数	MNIST，CIFAR10	快速线性算法
DEEPFOOL[145]	ℓ_p，$p \in [1, \infty)$	两者都可以	模型参数	MNIST，CIFAR10	迭代线性化
C&W[146]	ℓ_0，ℓ_2，ℓ_∞	两者都可以	逻辑	MNIST，CIFAR10	Adam 优化
JSMA[144]	ℓ_0	两者都可以	模型参数	MNIST，CIFAR10	Jacobian 显著性
DEEPGAME[46]	ℓ_0，ℓ_1，ℓ_2，ℓ_∞	无目标	逻辑	MNIST，CIFAR10	基于游戏的方法

（续）

	距离标准	有无目标	获得信息	测试数据集	核心方法
L0-TRE[156]	ℓ_0	无目标	逻辑	MNIST，CIFAR10，ImageNet	基于张量的网格搜索
DLV[8]	ℓ_1，ℓ_2	无目标	模型参数	MNIST，CIFAR10，GTSRB	一层一层的搜索
SAFECV[45]	ℓ_0	两者都可以	逻辑	MNIST，CIFAR10	随机搜索
文献［149］	不适用（自然变换）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	旋转/变换输入图片
文献［150］	$\mathcal{L}_{fow}$（测量几何距离）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	最小化对抗样本和 $\mathcal{L}_{fow}$ 损失
VERIVIS[151]	不适用（自然变换）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	12 个真是世界中的变换
UAP[143]	ℓ_2（通用扰动）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	通用化 DEEPFOOL 到一般的对抗攻击
UAN[152]	ℓ_p（通用扰动）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	通用化 C&W 到一般的对抗攻击
文献［153］	ℓ_p（通用扰动）	两者都可以	逻辑	MNIST，CIFAR10，ImageNet	训练一个一般网络

3　国内研究进展

人工智能系统的可信性研究近几年在国内得到极大的重视并迅速展开研究。在人工智能系统的安全可信内涵、形式化验证技术、测试技术、对抗与防御技术等方面都取得了相关的初步成果。

3.1　人工智能系统安全内涵

中国科学院何积丰在 2019 年的世界人工智能大会高端论坛“人工智能安全前沿探索”上做了题为“安全可信人工智能”的主旨演讲，系统而全面的丰富了其在 2017 年的香山会议上首次提出的可信人工智能的概念，强调人工智能系统不可信原因主要来自两方面，一是传统软件理论对未知环境的假设不够成熟，二是机器学习理论对训练数据的高度依赖性。他提出人工智能算法的关键安全问题包括：第一，尽量避免人工智能的副作用；第二，避免奖励条件的错误解读；第三，数据分布函数转变的稳定性；第四，探索的安全性；第五，可扩展的监管。提出可信人工智能应该具备与人类智能类似的特质，表现为鲁棒性、自我反省、自适应、透明性和可解释性，以及公平性。其中对人工智能系统应具备的自我反省与自适应特质的关注，引领世界前沿，丰富了安全可信人工

智能的内涵[157]。

为了对智能系统不确定环境的形式化建模理论进行扩展，何积丰提出了基于程序代数的概率程序演算模型，在传统概率模型中引入不确定元，用代数语义将概率算子与非确定算子的语义统一在一个框架下，突破了传统概率语义中对无限不确定性的近似处理，支持对不确定环境中的两种无限行为（有界非确定性和无界非确定性）进行描述和推理[158]。在该成果的启发下，华东师范大学的研究团队在智能驾驶决策系统的安全性研究工作中，考虑了环境车辆的不确定行为对智能驾驶决策系统的影响，提出了基于场景的智能决策系统安全性定量验证方法，支持对不同场景下智能决策模型的安全性进行度量和形式化验证[159]。

在人工智能系统的可信度量方面，李德毅在其专著《不确定性人工智能》中提出用超熵来度量智能系统的不确定性，利用认知的物理学方法，从定性定量双向认知转换的云模型，到云变换、数据场，到数据挖掘、智能控制和群体智能逐层展开，寻找不确定性知识和智能处理中的规律性，并对不确定性人工智能研究的发展方向进行了展望[160]。北京航空航天大学的康锐团队提出确信可靠性理论，用来度量不确定环境下的智能系统行为的可靠性[161]。华东师范大学的陈仪香团队从人工智能系统的多维属性出发，建立了基于人工智能系统多维属性的可信性度量模型，依据该度量模型计算出人工智能系统的可信性度量值。依据这个可信性度量值以及属性的重要性，确立人工智能系统的可信性量化等级，从而为人工智能系统的可信度量评估提供科学、合理、可操作的量化评估体系[162]。

东北大学王义团队针对人工智能系统在开放环境中不断演进（例如负责目标识别的神经网络的结构或参数随系统的部署环境不断演变）所导致系统的时间行为难以分析和预测的问题，提出了一种全新的、面向多核架构的、具有确定性的（deterministic）系统模型，系统化的解决智能系统在不断演化更新过程中的高效建模、设计、分析与优化的挑战[163]。为了改善基于机器学习的智能系统的训练效果，同时解决由于智能系统行为不确定性和解释性导致的智能系统测试 oracle 问题，西北工业大学的董云卫团队提出了基于蜕变神经元频谱的错误定位和修复方法[164]。通过蜕变神经元频谱记录智能系统执行蜕变测试组时的运行结果和神经元的激活状态，利用风险评估公式对可疑神经元进行定位，找出对神经网络错误行为影响较大的关键神经元。

在人工智能安全体系架构和标准体系方面，方滨兴在其专著《人工智能安全》中提出人工智能安全（safety and security）体系架构，从人工智能助力安全、内生安全、衍生安全等方面对人工智能系统面临的关键安全问题进行了系统阐述，并提出了相关的人工智能安全伦理准则[165]。由中国电子技术标准化研究院、清华大学、百度、华为、360、阿里巴巴等联合编写的《人工智能安全标准化白皮书（2019 版）》，从算法模型安全、数据安全和隐私保护、基础设施安全、应用安全等方面对人工智能系统安全风险的内涵进行了分析，并提出了一系列针对人工智能安全的标准体系，积极推进人工智能安全标准化工作[166]。

3.2 人工智能系统形式化验证技术

目前，国内对人工智能形式验证的工作主要集中在神经网络、深度学习程序以及黑盒系统的验证工作。

在神经网络的形式化验证方面，中科院软件所张立军的研究团队、国防科技大学陈立前与刘江潮、英国利物浦大学黄小炜等人合作研究提出了一种基于符号传播的方法来提高基于抽象解释的神经网络验证的精确性[30]。其主要思想是，在抽象域的基础上，引入符号变量表示每个神经元的值，然后从输入层向输出层逐层进行前向符号传播，以改进每个神经元的值范围。实验表明，该方法一方面可以提高基于抽象域的神经网络验证的精确性，从而可以证明更多的性质，另一方面，改进后的神经元的边界值信息可用于提高基于 SMT 的神经网络验证方法（如 Reluplex）的性能。浙江理工大学林望，华东师范大学何积丰、杨争峰，南京大学陈鑫，西南大学刘志明等人合作研究将鲁棒性的验证问题转化为一个等价的非线性优化求解问题，其目标是寻找该输入区域中找到最容易受干扰的点，并检查该点是否会分类错误[167]。为了形式验证该网络的鲁棒性，其采用区间计算和线性包含将原优化问题进行抽象，从而规避了由于浮点计算所产生的计算误差。为说明算法的有效性，他们分别在 MNIST 手写数字数据集、GTSRB 德国交通标志数据集等标准数据集上进行了验证。实验表明，通过改变扰动大小和网络结构进行的多组对照试验中，他们的方法相较于已有的方法均表现出更好的验证效果。华东师范大学的张民，上海科技大学的宋富等人提出一种基于目标标签的神经网络验证加速方法[168]。该方法通过线性松弛和符号传播技术计算输入样本在一定扰动区间内被误判为其他标签的概率区间，然后通过区间比较对错误的标签从大到小进行排序，最后依次验证神经网络针对该输入样本在每个标签上的鲁棒性，直到找到反例或者被证明是鲁棒的。该方法具有一定的通用性，将该方法应用于一些最新的神经网络验证工具，如 MIPVerify、DeepZ 和 Neurify 等，实验结果表明无论在验证结果和所需要的时间方面都使得原来的工具有很大的提升。

当前大部分研究工作关注的是模型层面的性质或缺陷。最近，北京大学熊英飞、谢涛等人、国防科技大学陈立前、香港科技大学张成志等人合作研究关注开发者编写的深度学习程序本身（即神经网络架构，如采用 TensorFlow 编写的程序）的缺陷，主要技术途径是将基于抽象解释的静态分析方法应用于人工智能深度学习程序中数值缺陷的检测，并针对深度学习程序的特点设计了相适应的抽象技术，如张量划分抽象等[169]。

在黑盒系统形式验证方面，中科院软件所薛白、北京大学孙猛等人合作提出了一种在可能近似正确学习框架下计算黑盒系统安全输入特征集的线性规划方法[170]。安全输入特征为使黑盒系统输出满足安全性质的输入特征。可能近似正确学习框架下的安全输入特征集为“在一定信心内，输入特征为安全输入特征的概率大于等于指定阈值”的集合。其主要思想是，将黑盒系统的安全输入特征集计算问题转化为一个鲁棒优化问题。然后，基于从黑盒系统中提取的有限数据集，利用控制论中的场景优化方法构建线性规划求解此鲁棒优化，得到其在可能近似正确学习框架下的解。受上述方法启发，中科院

软件所薛白、同济大学张苗苗、华东师范大学李钦等人合作进一步提出了验证有限时间连续“黑盒”动力系统的可能近似正确模型检验方法[171]。此模型检验方法可以给出连续“黑盒”动力系统在给定时间内满足安全需求的形式刻画：在一定的信心内系统满足安全需求的最少时间。

3.3 人工智能系统的测试技术

近年来，国内学者在神经网络测试方面做了大量的工作，并取得系列重要进展。本文囿于篇幅所限，仅对其中的测试覆盖准则、测试用例生成等主要方面进展做简要介绍。

在神经网络测试方面，哈尔滨工业大学马雷等人[58]提出网络层级的 top-k 神经元覆盖准则并开发出相应的验证工具 DeepGauge，基于变异测试技术设计开发出测试工具 DeepMutation[55]。在 DeepGauge 基础上，马雷等人也提出了基于变异的覆盖制导模糊测试技术 DeepHunter，以生成测试用例实现高覆盖度测试[172]。南京大学马晓星、许畅等对基于结构覆盖率的测试指标的有效性提出了疑问，认为 DNN 软件系统和常规软件系统具有本质的区别，相关标准未必适用。他们认为结构化覆盖率标准对发现 DNN 软件系统中的恶意输入太粗粒度，以至于很容易满足其标准，而对于发现误判的正常输入则太细粒度，以至于很难满足其标准。提出覆盖指标所引导发现的攻击样本是由于攻击样本本身在输入空间的大量普遍存在，先前的 DNN 软件系统测试显得覆盖率高可能是因为其基于面向恶意输入的搜索，而要达成找到很多恶意输入的效果实际是很容易做到的[173]。对应的，许畅等针对神经网络在部署时的置信度测试问题，提出了把软件测试视为通过统计抽样进行可靠性估计的思想，通过一种基于条件概率信息熵的机制从测试集中更高效地计算神经网络在实际部署时的置信度，将结构覆盖率理念重新解读为减少方差的条件评估，以此得到一种新的利用 DNN 模型学得特征进行条件评估，并对操作域数据高维稀疏分布进行交叉熵最小化的测试方法[174]。另一方面，DNN 模型经常以高置信度给出错误的预测。对此他们提出了一种基于贝叶斯操作的校准方法[175]，从较大的未标记操作数据集中仅选择少量数据进行标记，然后利用该标记数据纠正校准模型的置信度。

南京大学冯洋与陈振宇等人基于变异测试技术提出了模型层面的变异算子，从如何利用测试方法所产生测试用例来提升模型鲁棒性的角度，提出了一种基于神经网络最后一层神经元输出概率的基尼指数来衡量测试用例对提升模型鲁棒性的作用以进行测试用例筛选的方法，并开发了对应的工具。该团队还提出了 DeepGini[176]技术，用于神经网络的测试用例排序。DeepGini 将测试用例检测到错误行为的概率度量问题转化为对测试数据集纯度的评估分析问题，并根据 Gini 函数对测试用例进行加权。该技术能够帮助测试人员快速定位到可能检测到错误行为的测试用例，从而较好地提升深度神经网络的健壮性。相关团队进一步设计并实现了面向深度神经网络的 NerualVis 工具[177]。该工具可以可视化深度神经网络的静态结构与动态行为，并允许用户与神经网络交互。基于该工具，工程师可以分析神经网络结构并理解其行为，从而完成相关的工程任务。

清华大学软件学院姜宇等提出了基于差分对抗模糊测试的测试用例生成和后门检测

加固工作[178-180]。利用预测标签概率值的差异性引导用例变异，实现对抗样本的快速生成和后门数据的高效植入，提升了传统基于神经元引导的深度学习测试攻击方法的对抗样本生成数量和后门植入成功率，并在微众银行的人脸比对、活体检测等场景中进行了不同网络架构和系统平台的应用适配。

北京大学孙猛团队提出从模型输出不确定性的角度来区分正常样本和攻击样本，并发现了许多不易被之前测试方法所覆盖的攻击样本。深度学习系统会在对抗样本上做出错误决策，然而这些对抗样本对于人类认知往往非常平凡。该工作从模型的置信度以及非确定性两个维度出发提出了刻画良性数据以及对抗样本特点的方法。通过收集刻画置信度以及非确定性的多种度量标准，并比较研究了不同度量区分对抗样本和普通样本的能力。通过挑选区分能力较好的度量标准，以对良性数据以及对抗样本的行为进行刻画与分类。当前的良性数据以及现有工具生成的对抗样本具有明显的特征。以非确定性度量标准为向导，利用基于搜索的测试方法以生成多种异常数据，其与现有的良性数据与对抗样本具有完全不同的特点。实验结果发现这些异常数据能够绕过多种对抗样本防御技术[181]。谢涛团队则从实际开发者角度对深度学习软件开发过程所面临的各方面挑战进行了深入实证研究[182]。浙江大学王新宇团队提出了一种基于梯度的神经网络公平性测试方法用来高效搜索输入空间中的歧视样本[183]。

在上述技术研究以外，相关学者也在具体人工智能系统上展开了测试生成相关研究工作，如北京大学张路[184]等则针对机器翻译软件这一深度学习的重要应用场景提出了有针对性的测试和提升方法。天津大学王赞团队则针对常用的深度学习库进行测试，通过引入模型变异来更多地探索深度学习库的行为[185]。腾讯公司研究团队等[186]在机器学习翻译系统上进行的测试工作研究，南方科技大学张煜群等则针对无人车系统进行真实物理场景的系统的自动生成测试[187]，南京大学卜磊与上海科技大学宋富等在图像识别网络上进行的黑盒对抗样本生成等[188]。微软研究院对深度学习失败任务进行了系统性总结[189]。

3.4 人工智能系统的可解释性

在人工智能可解释性方面，相关研究近两年来得到了广泛的关注，国内研究者在该方面有了一定的快速发展。其中，上海交通大学的张拳石团队对卷积神经网络、生成网络等的可解释性都有较为充分的研究[190-191]。该团队提出了一种可解释的卷积神经网络模型，利用高卷积层过滤器对输入图像的不同特征部位进行表征，能够使神经网络在训练过程中自动学习过滤器与物体部位之间的对应关系，从而使该神经网络的识别过程转变为一个可解释的过程。该团队在此工作基础上进一步使用生成决策树的方法，将高卷积层过滤器与基于概念表示的决策树建立对应关系，得到关于 CNN 预测结果的语义解释。该系列研究通过解释图、决策树等方法对卷积神经网络进行可解释分析和图解。北京大学的朱占星团队在经过对抗训练的卷积神经网络进行了初步的可解释性探索[193]。该研究发现，在目标识别任务上，经过对抗训练的卷积神经网络更易于学习 shape-biased 表示。

国内在特征排序的研究上也取得了一系列的成果。在文献［194］中，来自中科院、南京大学、京东安全中心的学者与美国宾夕法尼亚州立大学和弗吉尼亚理工大学的学者合作，研究了针对安全软件的深度学习系统的特征排序。清华大学的刘奕群、马少平等人在文献［195］中将用户评论作为推荐系统的特征，通过对评论的排序来提高推荐系统的性能。这有别于例如文献［74］中单词级别的特征排序。在文献［196］中，来自微软中国、北京大学、清华大学、中国科技大学的学者合作提出了对推荐系统基于强化学习的（单词级别）的输入特征排序。上海交通大学的张拳石与美国加州大学洛杉矶分校的学者共同提出了通过构建一个决策树来对输入特征进行排序[74]，输入特征也即决策树上的结点，决策树有利于在语义层面上量化分析输入特征之间的联系以及其对输出的影响。

3.5 人工智能系统建模

在人工智能系统建模方面，国内研究者也取得了一些相关成果。在早期，上海交通大学钱大群和上海工业大学孙振飞[197]针对前馈神经网络的不同结构，给节点取不同的约束条件，并从中提取规则知识。西安交通大学刘振凯和贵忠华[198]对当时从前馈神经网络提取规则知识方面的相关工作进行总结分析。近期，南京大学周志华团队提出了一种从神经网络集成器中提取符号化规则的方法 REFNE[199]，和一种基于聚类算法从 RNN 中提取自动机的算法[200]。深圳大学软件理论实验室提出了一种基于 L * 和抽象解释从 CNN 中提取自动机的算法[201]，但是该算法需要折中考虑模型的准确率和抽象粒度。上海交通大学张拳石等人[192]提出了从 CNN 中提取决策树的方法，该方法的思想是将中间特征层分别与语义对象和 CNN 的预测关联起来。

3.6 人工智能系统的对抗攻击技术

近年来，国内学者在神经网络的对抗攻击方面取得一些初步的研究成功，主要工作集中在 2018 年以后，主要以和国外的研究机构合作为主，现阶段作者全部是国内团队的工作还比较少。本章囿于篇幅，将选取一些具有代表性的对抗攻击算法做简要介绍。其中本章将着重于讲黑盒对抗攻击算法，相比白盒攻击算法，黑盒攻击对算法有更高的要求。近期的研究表明，当前由深度神经网络训练的图像分类器在目标模型透明，也即白盒的情况下很容易被攻击生成对抗样本。但当一个封装良好的黑盒机器学习模型被攻击时，因为现有黑盒攻击算法往往需要对模型做大量的输入输出查询（query），会带来被攻击模型鲁棒的假象。

京东 AI 研究院的易津锋与美国加州洛杉矶大学和 IBM 合作，于 2018 年将针对硬标签的黑盒攻击问题转化为一种找寻到决策边界最短距离的方法，提出了 Opt-Attack 的黑盒对抗攻击算法[202]，并利用实值连续优化方法来提高查询效率。同时，为了验证黑盒攻击模型的鲁棒性，在 2019 年他们提出了一种可以降低查询次数的黑盒攻击算法，取名为 AutoZoom[203]，该黑盒攻击算法采用自适应随机梯度估计的思想，是一种能够平衡生

成对抗样本时所需查询次数和失真的方法，并且可以提高自动编码器的攻击过程。与此同时，易津锋与澳大利亚悉尼大学的陶大程研究组合作，通过一种 input-free 攻击的视角，提出了一种利用初始化灰度图像中的对抗样本来生成黑盒对抗样本的方法[204]。而针对目前生成高迁移性的对抗样本的方法基本用不同网络结构的被攻击模型进行集成作为替代模型，采用白盒攻击的方法进行对抗样本生成中的问题，天津大学智能与计算学部韩亚洪团队提出了一种 Curls&Whey 攻击方法[205]，能够解决使用 Curl 迭代沿梯度上升方向向输入样本单调添加噪声/扰动的问题，同时使用 Whey 优化技术从精心制作的对抗示例中消除过多的冗余噪声。

与此同时，在 2018 年为了寻找到合适的对抗攻击策略并且能够有效地提升模型本身的鲁棒性。清华大学的朱军团队提出了基于动量的迭代算法来构造对抗攻击样本，取名为 MI-FGSM[206]，MI-FGSM 的主要优点是能够有效地减轻白盒攻击成功率和迁移性能之间的耦合，并且能够同时成功攻击白盒和黑盒模型，具有更好的适用性。清华大学朱军团队随后还提出了一种基于平移不变性（Translation-Invariant）思想的攻击算法[207]，此算法主要思想是通过利用卷积神经网络的平移不变性来计算梯度，并它可以与任何基于梯度的攻击算法来进行集成，进一步提升对各种防御模型的攻击效果。

此外，国内还有一些团队结合进化算法和遗传算法提出了一些新的对抗攻击算法。比如，浙江工业大学的陈晋音团队提出了一种利用不同噪声点的像素阈值、噪声点数量和噪声点的大小三个参数来产生不同的扰动的攻击方法，取名为 POBA-GA[208]，该算法利用遗传算法（genetic algorithm）的思想，通过定义适应度函数（Fitness function）来评估攻击算法的有效性。与此同时，香港科技大学的张潼与谷歌和中佛罗里达大学的相关研究学者进行合作，在 2019 年提出了一种通过定义以正常样本为中心的局部范数球（Lp norm ball），然后通过学习这个局部范数球的概率密度来达到黑盒攻击的目的[209]。该算法同时采用了自然进化策略（Natural Evolution Strategy）思想，它的主要优势是能够改进投影梯度（PGD）攻击方法中的梯度估计不够可靠的问题。

以上的算法主要集中于对图片分类问题的深度学习模型进行对抗攻击。国内还有一些团队针对文本序列中的对抗攻击样本问题做了一些重要的工作。例如，在 2018 年中国人民大学的梁彬团队提出一种通过类似 HotFlip[210] 的方法来寻找最有影响力的字母、单词和句子，然后对这些热门字母、单词和句子进行添加、删除和修改等来实现对抗攻击[211]。京东 AI 实验室的易津锋等人和加州大学以及 IBM 的研究学者一起提出了名叫 Seq2Sick 的文本攻击算法[212]，该模型主要能够用来攻击机器翻译和文本摘要中的序列对序列（Seq2Seq）的深度学习模型。该算法主要的思路是将攻击问题转化为优化问题，并且通过优化使铰链状（hinge-like）的损失函数最小化，从而来解决离散扰动到达白盒攻击的目的。另外，好未来（TAL）AI 实验室的刘子韬研究员和密歇根州立大学的学者合作设计了一种针对神经对话模型（Neural Dialogue Model）的黑盒攻击算法，该算法的主要思想是采用强化学习（Reinforcement Learning）的框架来有效搜寻针对性响应的触发输入。该算法的黑盒设置更严格，同时放松了对生成响应的要求，即它只要求预期生成的响应在语义上与目标响应相同，但不一定需要与它们完全匹配。

4 国内外研究进展比较

人工智能系统安全可信的形式化验证方面的研究近几年在国内得到极大的重视，国内多个科研单位迅速展开研究，在形式化验证、测试、建模等各个方面都取得了国际领先的研究成果。本节分别从以上几个方面对国内外的研究进展进行比较分析。

针对人工智能安全内涵，当前国内外都尚未形成统一的标准。当前人工智能系统的“安全危机”主要包含数据质量、算法缺乏解释性、运算过程不可控、系统形态不可靠性等问题。由于现在的人工智能系统的设计与开发基本是靠数据驱动的，样本数据的质量直接影响系统的结果。此外，恶意的数据篡改和被污染的数据都会使得系统行为出现不可预判的风险。如何从不同的维度全面地刻画人工智能系统的安全，并利用形式化的方法给出严格的定义，是目前国内外都需要解决的关键问题。

在人工智能系统的形式化验证方面，目前国内外都比较聚焦于系统鲁棒性的验证。鲁棒性方面的验证之所以被广泛研究主要有两个方面的原因。一是鲁棒性是智能系统可靠性的一个重要的参数之一。另一个原因是鲁棒性在数学中有精确的定义，因此在形式化方法中相对容易描述与验证。与鲁棒性相对，智能系统的公平性，可解释性等性质目前尚缺少统一的定义，因此这方面的研究也相对较少。与国际上关于智能系统的形式化验证相比，国内的相关的研究相对较少，同时缺乏原创性的成果，大部分已有成果是在国外方法的基础上进行一定的优化在精度或者效率方面有一定的提升。整体上讲，当前人工智能系统的形式化验证国内外基本上同处在起步阶段，目前已有的验证方法大部分只能验证规模较小的模型。由于智能系统中智能模块的不可解释性和结构复杂性，对于系统的形式化描述，性质的形式化定义，以及算法的可扩展性等核心问题上至今国内外研究依然停留在理论阶段，尚无有效的方法解决工业领域智能系统的验证问题。

在人工智能系统测试方面，近三年来国内在人工智能系统测试领域取得了一系列重要前沿成果，并处在国际前沿水平。从上文的调研结果可以看出，国内多所高校和科研单位都在从事人工智能系统测试方面的研究，不仅从理论方法，而且在实际系统应用中取得了较大的进展。同样地，与传统的软件测试相比，人工智能系统测试技术目前依然处在起步阶段，由于当前大部分智能系统本身的不可解释性，相关的测试标准依然缺少足够的理论基础以确保测试结果的可靠性和准确性。此外，对抗攻击作为生成测试用例的一种特殊方法，如何通过对抗攻击与传统的测试用例生成技术相结合，生成有效的测试数据，如何根据测试结果更好地提升系统的安全可靠性，都是当前需要解决的问题。

在人工智能系统可解释方面，相对于国外研究，国内众多团队形成了自己的研究特色。不过，目前的国内外研究大多基于启发式方法，还缺乏对可解释性的严格统一定义，亟待提出严格的理论框架对可解释进行建模和相应的理论分析。输入特征排序是人工智能的一个非常活跃的领域，国内研究目前还与世界先进水平有一定差距，这体现在近年的高影响力文章仍以国外论文为主。不过，国内在某些特定应用场景下的特征排序形成

了自己的研究特色。另外值得注意的一点是，国际上近年来产生了一批受到广泛欢迎的特征排序工具，例如 LIME 和 SHAP，这些工具为扩大它们背后的理论算法的影响力起到了很大的作用，这也可以给国内的相关研究提供借鉴意义。

另外，关于神经网络知识模型构建方面的研究，绝大多数活跃在国外，国内对于这方面的研究尚未开始重视，活跃在这一领域的学者还不够多，同传统的形式化建模尚未形成一定的联系。

5 发展趋势与展望

人工智能系统的可信性已经逐渐成为制约人工智能技术在安全攸关领域应用的关键问题。可以预见，人工智能的可信性问题将会是工业界与学术界共同关注的焦点。由于这类系统的复杂性，不透明性以及不可解释性等，可信性很难得到绝对的保证，也给当前的形式化验证以及测试技术带来巨大的挑战。正如图灵奖获得者 Joseph Sifakis 所言，由于缺少系统规范，人工智能系统的形式化验证是对当前形式化技术的一大挑战。尽管目前学术界在深度神经网络验证方面已经取得了一定的进展并开发出相应的工具，然而这些方法与工具能否真正适用于真实世界的实际系统，还需要工业界验证。同样地，如何利用与改进传统的软件测试技术，使之用于人工智能系统的测试，也将会是未来研究的热点之一。

与当下人工智能效能的提高相对应的是人工智能系统复杂度的提升，这反过来又限制了人工智能的发展，在需要可信性保障的场景下人们至少需要对人工智能系统的判断结果有一定的理解。另外，人工智能经常被用来处理非常复杂的、高维度的输入数据，一个最直接的问题便是决定输入数据的哪些特征真正对人工智能的判断起了关键影响。在可预见的未来，特征排序会在人工智能领域起到越来越重要的作用。然而当前的特征排序研究存在很多挑战。由于缺乏统一标准，不同特征排序方法之间的公平比较往往非常困难，怎么样有效的从理论和实践两个方面克服这个困难也会成为影响特征排序研究领域发展的重要瓶颈。此外，目前最流行的一些特征排序方法往往关注人工智能对具体输入的判断，与之相比，对整个数据集层面的特征排序是一个更具挑战性的任务。更重要的一点是，特征排序只是理解人工智能行为的一种方法，一方面它是人们了解人工智能工作的最直观的途径，另一方面由于认知理解这类问题的复杂性与模糊性，特征排序并不是量化分析人工智能系统的最精确手段，特征排序在未来的发展很大程度上需要与本文中的其他人工智能保障性研究（例如测试方法，验证方法以及隐藏表示的可视化和追踪）相互配合。

随着神经网络规模越来越大，结构越来越复杂，以往的算法可能不再适用，如何从这些新型复杂网络中提取知识，特别是可解释可分析的形式化模型，可能是未来研究的一个关注点。另外一方面，目前已有的研究工作大多数是关注于处理序列数据的神经网络，处理非序列数据的神经网络同样值得关注。从模型的角度，如何从神经网络提取表

达能力更强的模型也是未来研究方向之一。此外，如何落地应用到实际使用的人工智能系统也是值得关注的问题。

6 结束语

人工智能系统对可信性要求的日益增高以及系统本身不可解释等特点，给传统的系统验证与测试方法带来巨大挑战的同时，也为今后颠覆性的技术创新带来新的机遇。本文总结了近些年来关于在人工智能系统的可信性方面国内外主要的研究进展，从人工智能系统可信性内涵、形式化验证、系统测试、对抗攻击、特征排序与量化追踪以及模型与知识提取这六个方面分别进行了介绍，以期对相关领域的科研人员提供参考。

致谢

本文是在 CCF 形式化方法专委的组织与指导下完成的。专委王戟主任，詹乃军副主任、董威秘书长统筹整个组稿过程。CCF 评审专家对本文从结构和技术层面都提出了很多宝贵意见。在此一并表示感谢。

参考文献

[1] Peter J Huber. Robust Statistics[C]. Wiley, New York. 1981.

[2] Frank R Hampel, Elvezio M Ronchetti, Peter J. Rousseeuw, et al. Robust statistics: the approach based on influence functions[C]. Wiley, New York. 1986.

[3] Thomas G. Dietterich. Steps Toward Robust Artificial Intelligence[J]. AI Magazine 38(3): 3-24. 2017.

[4] Ziqi Yang, Ee-Chien Chang, Jiyi Zhang, and Zhenkai Liang. Neural network inversion in adversarial setting via background knowledge alignment[C]. CCS 2019, pp. 225-240, doi: 10. 1145/3319535. 3354261.

[5] Liwei Song, Reza Shokri, Prateek Mittal. Privacy Risks of Securing Machine Learning Models against Adversarial Examples[C]. CCS 2019, pp. 241-257.

[6] Kenneth T Co, Luis Muñoz-González, Sixte de Maupeou, Emil C. Lupu. Procedural Noise Adversarial Examples for Black-Box Attacks on Deep Convolutional Networks[C]. CCS 2019, pp. 275-289.

[7] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks[C]. ICLR 2014. Citeseer

[8] Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu. Safety verification of deep neural networks [C]. CAV 2017, pp. 3-29.

[9] Xiaowei Huang, Daniel Kroening, Marta Kwiatkowska, Wenjie Ruan, Youcheng Sun, Emese Thamo, Min Wu, and Xinping Yi. Safety and Trustworthiness of Deep Neural Networks: A Survey[J]. arXiv preprint arXiv: 1812.08342, 2018.

[10] Wenjie Ruan, Xiaowei Huang, Marta Kwiatkowska. Reachability Analysis of Deep Neural Networks with Provable Guarantees[C]. IJCAI 2018, pp. 2651-2659.

[11] Changliu Liu, Tomer Arnon, Christopher Lazarus, Clark W. Barrett, and Mykel J. Kochenderfer. Algorithms for verifying deep neural networks[J]. arXiv preprint arXiv: 1903.06758, 2019.

[12] Luca Pulina and Armando Tacchella. An abstraction-refinement approach to verification of artificial neural networks[C]. CAV 2010. pp. 243-257.

[13] Yizhak Yisrael Elboher, Justin Gottschlich, and Guy Katz. An Abstraction-Based Framework for Neural Network Verification[C]. CAV 2020. pp. 43-65.

[14] Guy Katz, Clark Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. Reluplex: An efficient SMT solver for verifying deep neural networks[C]. CAV 2017. pp. 97-117.

[15] Ruediger Ehlers. Formal verification of piece-wise linear feed-forward neural networks[C]. ATVA 2017. pp. 269-286.

[16] Guy Katz et al. The marabou framework for verification and analysis of deep neural networks[C]. CAV 2019. pp. 443-452.

[17] Nina Narodytska, Shiva Kasiviswanathan, Leonid Ryzhyk, Mooly Sagiv, and Toby Walsh. Verifying properties of binarized deep neural networks[C]. AAAI 2018

[18] Nina Narodytska. Formal analysis of deep binarized neural networks[C]. IJCAI 2018. pp. 5692-5696.

[19] Alessio Lomuscio and Lalit Maganti. An approach to reachability analysis for feed-forward ReLU neural networks[C]. arXiv preprint arXiv: 1706.07351, 2017.

[20] Chih-Hong Cheng, Georg Nührenberg, and Harald Ruess. Maximum resilience of artificial neural networks[C]. ATVA 2017. pp. 251-268.

[21] Souradeep Dutta, Susmit Jha, Sriram Sankaranarayanan, and Ashish Tiwari. Output Range Analysis for Deep Feedforward Neural Networks[C]. NFM 2018. pp. 121-138.

[22] Rudy Bunel, Ilker Turkaslan, Philip HS Torr, Pushmeet Kohli, and M Pawan Kumar. Piecewise linear neural networks verification: A comparative study[C]. 2018.

[23] Arnold Neumaier and Oleg Shcherbina. Safe bounds in linear and mixed integer linear programming[C]. Mathematical Programming, pp. 99(2): 283-296, 2004.

[24] Patrick Cousot and Radhia Cousot. Systematic design of program analysis frameworks[C]. POPL 1979. pp. 269-282. ACM.

[25] Khalil Ghorbal, Eric Goubault, and Sylvie Putot. The zonotope abstract domain taylor1 + [C]. CAV 2009.

[26] Bertrand Jeannet, Antoine Min. Apron: A library of numerical abstract domains for static analysis[C]. CAV 2009. pp. 661-667.

[27] ELINA: ETH Library for Numerical Analysis. http://elina.ethz.ch.

[28] Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. AI2: Safety and robustness certification of neural networks with abstract interpretation[C]. IEEE Symposium on Security and Privacy 2018. pp. 3-18.

[29] Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably

robust neural networks[C]. ICML 2018. pp. 3575-3583.

[30] Pengfei Yang, Jiangchao Liu, Jianlin Li, Liqian Chen, and Xiaowei Huang. Analyzing deep neural networks with symbolic propagation: Towards higher precision and faster verification[J]. arXiv preprint arXiv: 1902.09866, 2019.

[31] Gagandeep Singh, Timon Gehr, Matthew Mirman, Markus Püschel, Martin T. Vechev. Fast and effective robustness certification[C]. NeurIPS 2018. pp. 10825-10836.

[32] Gagandeep Singh, Timon Gehr, Markus Püschel, Martin T. Vechev. An abstract domain for certifying neural networks[C]. PACMPL 3(POPL) 2019. pp. 41: 1-41: 30.

[33] Gagandeep Singh, Timon Gehr, Markus Püschel, Martin Vechev. Boosting Robustness Certification of Neural Networks[C]. ICLR 2019.

[34] Marc Fischer, Mislav Balunovic, Dana Drachsler-Cohen, Timon Gehr, Ce Zhang, Martin Vechev. DL2: Training and Querying Neural Networks with Logic[C]. ICML 2019.

[35] Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane S Boning, Inderjit S. Dhillon. Towards Fast Computation of Certified Robustness for ReLU Networks[C]. ICML 2018: 5273-5282.

[36] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions[C]. Advances in neural information processing systems 2018. pp. 4939-4948.

[37] Huan Zhang, Pengchuan Zhang, and Cho-Jui Hsieh, Recurjac: An efficient recursive algorithm for bounding jacobian matrix of neural networks and its applications[C]. AAAI 2019. vol. 33, pp. 5757-5764.

[38] Akhilan Boopathy, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. Cnn-cert: An efficient framework for certifying robustness of convolutional neural networks[C]. AAAI 2019. pp. 3240-3247.

[39] Ching-Yun Ko, Zhaoyang Lyu, Lilu Weng, Luca Daniel, Ngai Wong, and Dahua Lin. POPQORN: Quantifying Robustness of Recurrent Neural Networks[C]. ICML 2019. pp. 3468-3477.

[40] Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope[C]. ICML 2018 pp. 5283-5292.

[41] Krishnamurthy Dvijotham, Robert Stanforth, Sven Gowal, Timothy A Mann, and Pushmeet Kohli. A dual approach to scalable verification of deep networks[C]. UAI 2018. pp. 550-559.

[42] Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Formal security analysis of neural networks using symbolic intervals[C]. {USENIX} Security Symposium 2018. pp. 1599-1614.

[43] Jonathan Peck, Joris Roels, Bart Goossens, and Yvan Saeys. Lower bounds on the robustness to adversarial perturbations[C]. NIPS 2017. pp. 804-813.

[44] Weiming Xiang, Hoang-Dung Tran, and Taylor T Johnson. Output reachable set estimation and verification for multi-layer neural networks[C]. IEEE Transactions on Neural Networks and Learning Systems 2018. pp. 29: 5777-5783.

[45] Matthew Wicker, Xiaowei Huang, and Marta Kwiatkowska. Feature-guided black-box safety testing of deep neural networks[J]. TACAS(1) 2018. pp. 408-426.

[46] Min Wu, Matthew Wicker, Wenjie Ruan, Xiaowei Huang, and Marta Kwiatkowska. A game-based approximate verification of deep neural networks with provable guarantees[J]. arXiv preprint arXiv: 1807.03571, 2018.

[47] Min Wu and Marta Kwiatkowska. Robustness guarantees for deep neural networks on videos[C]. in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020. pp. 311-320.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition[C]. ICLR 2015.

[49] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild[C]. CRCV-TR-12-01, 2012.

[50] Wenjie Ruan, Min Wu, Youcheng Sun, Xiaowei Huang, Daniel Kroening, and Marta Kwiatkowska[C]. Global robustness evaluation of deep neural networks with provable guarantees for the hamming distance. IJCAI 2019.

[51] Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel. Evaluating the robustness of neural networks: An extreme value theory approach[J]. arXiv preprint arXiv: 1801.10578, 2018.

[52] Ian Goodfellow. Gradient masking causes CLEVER to overestimate adversarial perturbation size[J]. arXiv preprint arXiv: 1804.07870, 2018.

[53] Osbert Bastani, Yani Ioannou, Leonidas Lampropoulos, Dimitrios Vytiniotis, Aditya Nori, and Antonio Criminisi. Measuring neural net robustness with constraints[C]. NIPS 2016. pp. 2613-2621.

[54] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems[C]. SOSP 2017. pp. 1-18.

[55] Lei Ma, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Felix Juefei-Xu, Chao Xie, Li Li, Yang Liu, Jianjun Zhao, and Yadong Wang. Deepmutation: Mutation testing of deep learning systems[C]. ISSRE 2018. pp. 100-111.

[56] Jingyi Wang, Guoliang Dong, Jun Sun, Xinyu Wang, and Peixin Zhang. Adversarial sample detection for deep neural network through model mutation testing[C]. ICSE 2019. pp. 1245-1256.

[57] Youcheng Sun, Min Wu, Wenjie Ruan, Xiaowei Huang, Marta Kwiatkowska, Daniel Kroening. Concolic testing for deep neural networks[C]. ASE 2018. pp. 109-119.

[58] Lei Ma, Felix Juefei-Xu, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Chunyang Chen, Ting Su, Li Li, Yang Liu, et al. Deepgauge: Multi-granularity testing criteria for deep learning systems[C]. ASE 2018.

[59] Jinhan Kim, Robert Feldt, and Shin Yoo. Guiding deep learning system testing using surprise adequacy [C]. ICSE 2019.

[60] Weijun Shen, Jun Wan, Zhenyu Chen. MuNN: Mutation Analysis of Neural Networks. Proc[C]. of 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), IEEE. 2018. pp. 108-115.

[61] Quanzhi Zhou, Liqun Sun. Metamorphic testing of driverless cars[C]. ACM 2019. pp. 62(3): 61-67.

[62] Xiaofei Xie, Lei Ma, Felix Juefei-Xu, Hongxu Chen, Minhui Xue, Bo Li, Yang Liu, Jianjun Zhao, Jianxiong Yin, and Simon See. Coverage-guided fuzzing for deep neural networks[C]. ISSTA 2019.

[63] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray. Deeptest: Automated testing of deep-neural-network-driven autonomous cars[C]. ICSE 2018. pp. 303-314.

[64] Shiqing Ma, Yingqi Liu, Wen-Chuan Lee, Xiangyu Zhang, and Ananth Grama. MODE: automated neural network model debugging via state differential analysis and input selection[C]. FSE 2018. pp.

175-186.

[65] Sakshi Udeshi, Pryanshu Arora, and Sudipta Chattopadhyay. Automated directed fairness testing[C]. ICSE 2018. pp. 98-108.

[66] Riccardo Guidotti, Anna Monreale, Franco Turini, Dino Pedreschi, Fosca Giannotti. A Survey of Methods for Explaining Black Box Models[J]. ACM Computing Surveys 2018. pp. 51(5): 1-42.

[67] Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps[J]. arXiv preprint arXiv: 1312.6034, 2013.

[68] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, Antonio Torralba. Learning deep features for discriminative localization[C]. CVPR 2016. pp. 2921-2929.

[69] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]. ICCV 2017. pp. 618-626.

[70] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition[C]. Pattern Recognition 2017. pp. 65: 211-222.

[71] Avanti Shrikumar, Peyton Greenside, Anshul Kundaje. Learning important features through propagating activation differences[C]. ICML 2017. pp. 3145-3153.

[72] Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks[C]. ICML 2017. pp. 3319-3328.

[73] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian J. Goodfellow, Moritz Hardt, Been Kim. Sanity checks for saliency maps[C]. NeurIPS 2018. pp. 9505-9515.

[74] Marco Túlio Ribeiro, Sameer Singh, Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier[C]. ACM SIGKDD international conference on knowledge discovery and data mining 2016. pp. 1135-1144.

[75] Anupam Datta, Shayak Sen, Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems[C]. IEEE symposium on security and privacy (SP) 2016. pp. 598-617.

[76] Scott M Lundberg, Su-In Lee. A unified approach to interpreting model predictions[C]. NIPS 2017. pp. 4765-4774.

[77] Jianbo Chen, Le Song, Martin J. Wainwright, and Michael I. Jordan. Learning to explain: An information-theoretic perspective on model interpretation[J]. arXiv preprint arXiv: 1802.07814, 2018.

[78] Ruth C Fong, Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation[C]. ICCV 2017. pp. 3429-3437.

[79] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. Fairness testing: testing software for discrimination[C]. Joint Meeting on Foundations of Software Engineering 2017. pp. 498-510.

[80] Youcheng Sun, Hana Chokler, Xiaowei Huang and Daniel Kroening. Explaining Deep Neural Networks Using Spectrum-Based Fault Localization.

[81] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle[C]. ITW 2015. pp. 1-5.

[82] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information [C]. CoRR, abs/1703.00810, 2017.

[83] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method[C]. Annual Allerton Conference on Communication, Control, and Computing 1999. pp. 368-377.

[84] Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox. On the information bottleneck theory of deep learning[C]. ICLR 2018.

[85] Ziv Goldfeld, Ewout van den Berg, Kristjan Greenewald, Igor Melnyk, Nam Nguyen, Brian Kingsbury, and Yury Polyanskiy. Estimating information flow in deep neural networks[C]. ICML 2019.

[86] Gaojie Jin, Xiaowei Huang, and Xinping Yi. Tracking information evolution of hidden representations for neural networks via kernelised vector quantisation[C]. Thirty-third Conference on Neural Information Processing Systems 2019.

[87] Jeffrey L Elman. Finding structure in time[C]. Cognitive Science. pp. 14, 179-211, 1990.

[88] Jacobsson, Henrik. Rule Extraction from Recurrent Neural Networks: ATaxonomy and Review. Neural Computation 2005[J]. pp. 17(6): 1223-1263.

[89] Axel Cleeremans, David Servan-Schreiber, James L. McClelland. Finite state automata and simple recurrent networks[C]. Neural Computation 1989. pp. 1, 372-381.

[90] Servan-Schreiber, D Cleeremans, A & McClelland J L. Learning sequential structure in simple recurrent networks[C]. D. S. Touretzky (Ed.), Advances in neural information processing systems, 1(pp. 643-652). San Mateo, CA: Morgan Kaufmann, 1989.

[91] David Servan-Schreiber, Axel Cleeremans, James L. McClelland. Graded state ma-chines: The representation of temporal contingencies in simple recurrent net-works[C]. Machine Learning, pp. 7, 161-193. 1991.

[92] Giles, C L, Miller, C B, Chen, D, Chen, H H, & Sun, G Z. Learning and extracting finite state automata with second-order recurrent neural networks[C]. Neural Computation. pp. 4(3), 393-405, 1992.

[93] Giles, C L, Chen, D, Miller, C, Chen, H, Sun, G, & Lee, Y. Second-order recurrent neural networks for grammatical inference[C]. Proceedings of International Joint Conference on Neural Networks. pp. 273-281. Piscataway, NJ: IEEE, 1991

[94] Christian W Omlin, C Lee Giles. Extraction of rules from discrete-time recurrent neural networks[J]. Neural Networks. pp. 9(1), 41-51, 1996.

[95] Zheng Zeng, Rodney M Goodman, Padhraic Smyth. Learning finite state machines with self-clustering recurrent networks. Neural Computation[J]. pp. 5(6), 976-990, 1993.

[96] Paolo Frasconi, Marco Gori, Marco Maggini, Giovanni Soda. Representation of finite state automata in recurrent radial basis function networks[J]. Machine Learning. pp. 23(1), 5-32, 1996.

[97] Marco Gori, Marco Maggini, Enrico Martinelli, Giovanni Soda. Inductive inference from noisy examples using the hybrid finite state filter. IEEE Transactions on Neural Networks[J]. pp. 9(3), 571-575, 1998.

[98] Armando Blanco, Miguel Delgado, Maria del Carmen Pegalajar. Extracting rules from a (fuzzy/crisp) recurrent neural network using a self-organizing map[C]. International Journal of Intelligent Systems. pp. 15, 595-621, 2000.

[99] Alquézar, R, Sanfeliu, A. A hybrid connectionist symbolic approach to regular grammar inference based on neural learning and hierarchical clustering[C]. Proceedings of ICGI'94 (pp. 203-211). Berlin:

Springer-Verlag, 1994.

[100] Sanfeliu, A, & Alquézar, R. . Active grammatical inference: A new learning methodology. In: Shape, Structure and Pattern Recognition, 5th IAPR International Workshop on Structural and Syntactic Pattern Recognition (pp. 191-200)[M]. Singapore: World Scientific, 1995.

[101] Watrous, R L, & Kuhn, G M. Induction of finite-state automata using second-order recurrent networks. In: J. E. Moody, S. J. Hanson, & R. P. Lippmann (Eds.), Advances in neural information processing systems, 4(pp. 309-317)[C]. San Mateo, CA: Morgan Kaufmann, 1992.

[102] Manolios, P, Fanelli, R. First order recurrent neural networks and deter-ministic finite state automata [C]. Neural Computation. pp. 6(6), 1155-1173, 1994.

[103] Tinǒ, P, & Sǎajda, J. . Learning and extracting initial mealy automata with a modular neural network model[J]. Neural Computation. pp. 7(4), 822-844, 1995.

[104] Kohonen, T. Self-organizing maps[M]. Berlin: Springer, 1995.

[105] Ingo Schellhammer, Joachim Diederich, Michael Towsey, Claudia Brugman. Knowledge extraction and recurrent neural networks: An analysis of an Elman network trained on a natural language learning task [C]. CoNLL 1998. pp. 73-78.

[106] Paz, A. . Introduction to probabilistic automata[C]. Orlando, FL: Academic Press, 1971.

[107] Peter Tiño, V Vojtek. Extracting stochastic machines from recurrent neural networks trained on complex symbolic sequences[C]. Neural Network World. pp. 8(5), 517-530, 1998.

[108] Peter Tiño, Miroslav Koteles. Extracting finite-state representations from recurrent neural networks trained on chaotic symbolic sequences[J]. IEEE Transactions on Neural Networks. pp. 10(2), 284-302, 1999.

[109] Peter Tiño, Georg Dorffner, Christian Schittenkopf. Understanding state space organization in recurrent neural networks with iterative function systems dynamics. S. Wermter & R. Sun (Eds.), Hybrid neural symbolic integration. pp. 256-270[M]. Berlin: Springer-Verlag, 2000.

[110] Jörg Bruske, Gerald Sommer. Dynamic cell structure learns perfectly topology preserving map[C]. Neural Computation. pp. 7, 845-865, 1995.

[111] Guillaume Rabusseau, Tianyu Li, Doina Precup. Connecting Weighted Automata and Recurrent Neural Networks through Spectral Learning[C]. CoRR, abs/1807.01406. 2018.

[112] Vahed, A, & Omlin, C W. Rule extraction from recurrent neural networks using a symbolic machine learning algorithm (Tech. Rep. No. US-CS-TR-4)[C]. Stel-lenbosch, South Africa, University of Stellenbosch, 1999.

[113] Vahed, A, & Omlin, C W. A machine learning method for extracting symbolic knowledge from recurrent neural networks[C]. Neural Computation. pp. 16, 59-71, 2004.

[114] Gail Weiss, Yoav Goldberg, Eran Yahav. Extracting automata from recurrent neural networks using queries and counterexamples[C]. ICML 2018. pp. 5244-5253.

[115] Angluin, D. Learning regular sets from queries and coun-terexamples[C]. Inf. Comput.. pp. 75(2): 87-106, 1987.

[116] Franz Mayr, Sergio Yovine. Regular Inference on Artificial Neural Networks[C]. Machine Learning and Knowledge Extraction. pp. 350-369. 2018.

[117] Stéphane Ayache, Rémi Eyraud, and Noé Goudian. Explaining black boxes on sequential data using weighted automata[C]. ICGI 2018. pp. 81-103.

[118] Takamasa Okudono, Masaki Waga, Taro Sekiyama, Ichiro Hasuo. Weighted Automata Extraction from Recurrent Neural Networks via Regression on State Spaces[C]. CoRR, abs/1904.02931. 2019.

[119] Qinglong Wang, Kaixuan Zhang, Alexander G. Ororbia II, Xinyu Xing, Xue Liu, C. Lee Giles. A comparison of rule extraction for different recurrent neural network models and grammatical complexity [C]. CoRR, abs/1801.05420. 2018.

[120] Anurag Koul, Alan Fern, Sam Greydanus. Learning Finite State Representations of Recurrent Policy Networks[C]. ICLR 2019.

[121] Ikram Chraibi Kaadoud, Nicolas P Rougier, Frédéric Alexandre. Knowledge extraction from the learning of sequences in a long short term memory (LSTM) architecture[C]. CoRR abs/1912.03126. 2019.

[122] Wenjie Lu, Dikai Liu. A2: Extracting cyclic switchings from DOB-nets for rejecting excessive disturbances[C]. Neurocomputing. pp.400: 161-172, 2020.

[123] William Merrill. Sequential Neural Networks as Automata[C]. CoRR abs/1906.01615. 2019.

[124] Joshua J Michalenko, Ameesh Shah, Abhinav Verma, Richard G Baraniuk, Swarat Chaudhuri, Ankit B Patel. Representing Formal Languages: A Comparison Between Finite Automata and Recurrent Neural Networks[C]. ICLR 2019.

[125] Qinglong Wang, Kaixuan Zhang, Xue Liu, C. Lee Giles. Connecting First and Second Order Recurrent Networks with Deterministic Finite Automata[C]. CoRR abs/1911.04644. 2019.

[126] Christian Oliva, Luis F Lago-Fernández. On the Interpretation of Recurrent Neural Networks as Finite State Machines[J]. ICANN (1) 2019. pp.312-323.

[127] Reda Marzouk, Colin de la Higuera. Distance and Equivalence between Finite State Machines and Recurrent Neural Networks: Computational results[C]. CoRR abs/2004.00478. 2020.

[128] Bishwamittra Ghosh, Daniel Neider. A Formal Language Approach to Explaining RNNs[C]. CoRR abs/2006.07292. 2020.

[129] Geoffrey G. Towell, Jude William Shavlik. The extraction of refined rules from knowledge-based neural networks[C]. Machine Learning. pp.13, 71-101, 1993.

[130] Sabrina Sestito, Tharam S. Dillon. Knowledge acquisition of conjunctive rules using multilayered neural networks, International Jour-nal of Intelligent Systems 8[C]. pp.779-805, 1993.

[131] Rudy Setiono. Extracting rules from neural networks by pruning and hidden-unit splitting, Neural Computation 9[C]. pp.205-225, 1997.

[132] Rudy Setiono, Wee Kheng Leow. FERNN: An algorithm for fast extraction of rules from neural networks, Applied Intelligence[C]. pp.12, 15-25, 2000.

[133] Masumi Ishikawa. Rule extraction by successive regularization. Neural Networks[C]. pp.13, 1171-1183, 2000.

[134] Mark W Craven, Jude W. Shavlik. Extracting tree-structured representations of trained networks. NIPS 1996[C]. pp.24-30.

[135] Gregor P J Schmitz, Chris Aldrich, F S Gouws. ANN-DT: An algorithm for extraction of decision trees from artificial neural networks[C]. IEEE Press, 1999.

[136] Bondarenko Andrey, Aleksejeva Ludmila, Jumutc Vilen, et al. Classification Tree Extraction from Trained Artificial Neural Networks. Procedia Computer Science[C]. pp.104: 556-563, 2017.

[137] Nicholas Frosst, Geoffrey E. Hinton. Distilling a neural network into a soft decision tree[J]. arXiv preprint arXiv: 1711.09784, 2017.

[138] Mike Wu, Michael C Hughes, Sonali Parbhoo, Maurizio Zazzi, Volker Roth, and Finale Doshi-Velez. Beyond sparsity: Tree regularization of deep models for interpretability[C]. AAAI 2018.

[139] Andy Shih, Adnan Darwiche, and Arthur Choi. Verifying Binarized Neural Networks by Angluin-Style Learning[C]. SAT 2019.

[140] Steffen Hölldobler, Yvonne Kalinke. Towards a new massively parallel computational model for logic programming[C]. ECCAI 1994. pp 68-77.

[141] Jens Lehmann, Sebastian Bader, Pascal Hitzler. Extracting reduced logic programs from artificial neural networks. Applied intelligence[C]. pp. 32, no. 3: 249-266, 2010.

[142] Ian J Goodfellow, Jonathon Shlens, Christian Szegedy. Explaining and harnessing adversarial examples [C]. CoRR, abs/1412.6572.

[143] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, Pascal Frossard, Universal adversarial perturbations[C]. In: CVPR, pp. 86-94, 2017.

[144] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, Ananthram Swami. The limitations of deep learning in adversarial settings[C]. EuroS&P 2016. pp. 372-387.

[145] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks[C]. CVPR 2016. pp. 2574-2582.

[146] Nicholas Carlini, David A Wagner. Towards evaluating the robustness of neural networks[C]. Security and Privacy (SP) 2017. pp. 39-57.

[147] Alexey Kurakin, Ian J. Goodfellow, Samy Bengio. Adversarial machine learning at scale[C]. ICLR (Poster) 2017.

[148] Diederik P Kingma, Jimmy Ba Adam: a method for stochastic optimization [C]. Computer Science, 2014.

[149] Logan Engstrom, Dimitris Tsipras, Ludwig Schmidt, Aleksander Madry. A rotation and a translation suffice: Fooling cnns with simple transformations[J]. arXiv preprint arXiv: 1712.02779, 2017.

[150] Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, Dawn Song. Spatially transformed adversarial examples[J]. arXiv preprint arXiv: 1801.02612, 2018.

[151] Kexin Pei, Yinzhi Cao, Junfeng Yang, Suman Jana. Towards practical verification of machine learning: The case of computer vision systems[J]. arXiv preprint arXiv: 1712.01785, 2017.

[152] Jamie Hayes, George Danezis. Learning universal adversarial perturbations with generative models[C]. IEEE Security and Privacy Workshops (SPW) 2018. pp. 43-49.

[153] Omid Poursaeed, Isay Katsman, Bicheng Gao, Serge J Belongie. Generative adversarial perturbations [C]. CVPR 2018.

[154] Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-net: convolutional networks for biomedical image segmentation[C]. MICCAI 2015. pp. : 234-241.

[155] Justin Johnson, Alexandre Alahi, Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution[C]. ECCV 2016. pp. 694-711.

[156] Wenjie Ruan, Min Wu, Youcheng Sun, Xiaowei Huang, Daniel Kroening, Marta Kwiatkowska. Global robustness evaluation of deep neural networks with provable guarantees for L0 norm[J]. arXiv preprint arXiv: 1804.05805, 2018b.

[157] 何积丰. 安全可信人工智能[C]. 信息安全与通信保密. pp. 5-8, 2019.

[158] Jifeng He. Linking theories of probabilistic programming[C]. Symposium on Real-Time and Hybrid

Systems 2018. pp. 186-210, doi: 10.1007/978-3-030-01461-2_10.

[159] Bingqing Xu, Qin Li, Tong Guo, Yi Ao, Dehui Du. A Quantitative Safety Verification Approach for the Decision-making Process of Autonomous Driving[C]. TASE 2019. pp. 128-135, doi: 10.1109/TASE.2019.000-9.

[160] 李德毅, 杜鹢. 不确定性人工智能(第2版)[M]. 国防工业出版社, 2014.

[161] Qingyuan Zhang, Rui Kang, Meilin Wen. Belief Reliability for Uncertain Random Systems[C]. IEEE Trans. Fuzzy Syst. , vol. 26, no. 6, pp. 3605-3614, doi: 10.1109/TFUZZ.2018.2838560.

[162] Hongwei Tao, Hengyang Wu, and Yixiang Chen. An approach of trustworthy measurement allocation based on sub-attributes of software[C]. Mathematics, vol. 7, no. 3, pp. 1-15, 2019, doi: 10.3390/math7030237.

[163] Yi Wang. Towards Customizable CPS: Composability, Efficiency and Predictability[C]. ICFEM 2017, doi: 10.1007/978-3-319-68690-5_1.

[164] Tingting Wu, Yunwei Dong, Yu Zhang, and Aziz Singa. ExtendAIST: Exploring the space of ai-in-the-loop system testing. Appl[C]. Sci. , 2020, doi: 10.3390/app10020518.

[165] 方滨兴. 人工智能安全[M]. 电子工业出版社, 2020.

[166] 全国信息安全标准化技术委员会[C]. 人工智能安全标准化白皮书(2019版). 2019.

[167] Wang Lin, Zhengfeng Yang, Xin Chen, Qingye Zhao, Xiangkun Li, Zhiming Liu, Jifeng He. Robustness Verification of Classification Deep Neural Networks via Linear Programming[C]. CPVR 2019. pp. 11418-11427. IEEE, 2019.

[168] Wenjie Wan, Zhaodi Zhang, Yiwei Zhu, Min Zhang, Fu Song. Accelerating Robustness Verification of Deep Neural Networks Guided by Target Labels[J], arXiv preprint arXiv: 2007.08520, 1-20, 2020.

[169] Yuhao Zhang, Luyao Ren, Liqian Chen, Yingfei Xiong, Shing-Chi Cheung, Tao Xie. Detecting Numerical Bugs in Neural Network Architectures[C]. In Proc. of the 2020 ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020). Sacramento, California, United States, November 8-13, 2020. (To appear).

[170] Bai Xue, Yang Liu, Lei Ma, Xiyue Zhang, Meng Sun and Xiaofei Xie. Safe Inputs Approximation for Black-Box Systems[C]. In Proceeding of the 24th International Conference on Engineering of Complex Computer Systems (ICECCS) 2019, pp. 180-189.

[171] Bai Xue, Miaomiao Zhang, Arvind Easwaran and Qin Li. PAC Model Checking of Black-Box Continuous-Time Dynamical Systems[C]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (IEEE TCAD), 2020. (To appear).

[172] Xiaofei Xie, Lei Ma, Felix Juefei-Xu, Minhui Xue, Hongxu Chen, Yang Liu, Jianjun Zhao, Bo Li, Jianxiong Yin, Simon See. DeepHunter: a coverage-guided fuzz testing framework for deep neural networks[C]. In Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2019.

[173] Zenan Li, Xiaoxing Ma, Chang Xu, and Chun Cao. Structural Coverage Criteria for Neural Networks Could Be Misleading[C]. In Proceedings of the 41st ACM/IEEE International Conference on Software Engineering (ICSE 2019 NIER), pp. 89-92, Montreal, QC, Canada, May 2019.

[174] Zenan Li, Xiaoxing Ma, Chang Xu, Chun Cao, Jingwei Xu, and Jian Lu. Boosting Operational DNN Testing Efficiency through Conditioning[C]. In Proceedings of the 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2019),

pp. 499-509, Tallinn, Estonia, Aug 2019.

[175] Zenan Li, Xiaoxing Ma, Chang Xu, Jingwei Xu, Chun Cao, and Jian Lu. Operational Calibration: Debugging Confidence Errors for DNNs in the Field[C]. In Proceedings of the 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020), Sacramento, California, USA, Nov 2020.

[176] Feng, Yang and Shi, Qingkai and Gao, Xinyu and Wan, Jun and Fang, Chunrong and Chen, Zhenyu. DeepGini: prioritizing massive tests to enhance the robustness of deep neural networks[C]. Proceedings of the 29th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA), 2019.

[177] Xufan Zhang, Ziyue Yin, Yang Feng, Qingkai Shi, Jia Liu, and Zhenyu Chen. NeuralVis: visualizing and interpreting deep learning models[C]. IEEE/ACM International Conference on Automated Software Engineering (ASE) 2019. IEEE, 2019.

[178] Jianmin Guo, Yu Jiang, Yue Zhao, Quan Chen, Jiaguang Sun. DLFuzz: Differential Fuzzing Testing of Deep Learning Systems[C]. ESEC/SIGSOFT FSE 2018.

[179] Jianmin Guo, Yue Zhao, Xueying Han, Yu Jiang, Jiaguang Sun. RNN-Test: Adversarial Testing Framework for Recurrent Neural Network Systems[C]. ARXIV 2019.

[180] Jianmin Guo, Yue Zhao, Yu Jiang. Coverage Guided Differential Adversarial Testing of Deep Learning Systems[C]. TNSE 2020.

[181] Xiyue Zhang, Xiaofei Xie, Lei Ma, Xiaoning Du, Qiang Hu, Yang Liu, Jianjun Zhao, Meng Sun. Characterizing Adversarial Defects of Deep Learning Software from the Lens of Uncertainty[C]. in Proceedings of ICSE 2020. pp. 739-751. ACM, 2020.

[182] Chen, Zhenpeng, Yanbin Cao, Yuanqiang Liu, Haoyu Wang, Tao Xie, and Xuanzhe Liu. Understanding Challenges in Deploying Deep Learning Based Software: An Empirical Study[C]. In Proc. of the 2020 ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020). Sacramento, California, United States, November 8-13, 2020. (To appear).

[183] Peixin Zhang, Jingyi Wang, Jun Sun, Guoliang Dong, Xinyu Wang, Xingen Wang, Jin Song Dong, and Ting Dai. White-box fairness testing through adversarial sampling[C]. 2020.

[184] Zeyu Sun, Jie M. Zhang, Mark Harman, Mike Papadakis, Lu Zhang. Automatic Testing and Improvement of Machine Translation[C]. in Proceedings of ICSE 2020. ACM, 2020.

[185] Zan Wang, Ming Yan, Junjie Chen, Shuang Liu, Dongdi Zhang. Deep Learning Library Testing via Effective Model Generation[C]. In Proceedings of the 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020), Sacramento, California, USA, Nov 2020.

[186] Wenyu Wang, Wujie Zheng, Dian Liu, Changrong Zhang, Qinsong Zeng, Yuetang Deng, Wei Yang, Pinjia He, Tao Xie. Detecting Failures of Neural Machine Translation in the Absence of Reference Translations[C]. DSN (Industry Track) 2019. pp. 1-4.

[187] Mengshi Zhang, Yuqun Zhang, Lingming Zhang, Cong Liu, Sarfraz Khurshid. DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems[C]. ASE 2018. pp. 132-142.

[188] Yuchao Duan, Zhe Zhao, Lei Bu, Fu Song. Things You May Not Know About Adversarial Example: A Black-box Adversarial Image Attack[C]. CoRR abs/1905.07672, 2019.

[189] Zhang, Ru, Wencong Xiao, Hongyu Zhang, Yu Liu, Haoxiang Lin, and Mao Yang. An Empirical Study on Program Failures of Deep Learning Jobs.

[190] Quanshi Zhang, Yingnian Wu, and Song-chun Zhu. Interpretable Convolutional Neural Networks. Proc [C]. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 8827-8836, 2018, doi: 10.1109/CVPR.2018.00920.

[191] Quanshi Zhang, Yu Yang, Haotian Ma, Ying Nian Wu. Interpreting cnns via decision trees[C]. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. , vol. 2019-June, pp. 6254-6263, 2019, doi: 10.1109/CVPR.2019.00642.

[192] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting CNN Knowledge via an Explanatory Graph[C]. AAAI, 2018.

[193] Tianyuan Zhang, Zhanxing Zhu. Interpreting Adversarial Trained Convolutional Neural Networks[C]. ICML 2019.

[194] Wenbo Guo, Dongliang Mu, Jun Xu, Purui Su, Gang Wang, and Xinyu Xing. Lemna: Explaining deep learning based security applications[C]. ACM SIGSAC Conference on Computer and Communications Security 2018. pp. 364-379.

[195] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Neural attentional rating regression with review-level explanations[C]. World Wide Web Conference 2018. pp. 1583-1592. International World Wide Web Conferences Steering Committee, 2018.

[196] Wang, Xiting, Yiru Chen, Jie Yang, Le Wu, Zhengtao Wu, and Xing Xie. A Reinforcement Learning Framework for Explainable Recommendation[C]. ICDM 2018. pp. 587-596.

[197] 钱大群, 孙振飞. 神经网络的知识获取与行为解释[J]. 自动化学报, 1994(03): 348-351.

[198] 刘振凯, 贵忠华. 基于人工神经网络的知识获取方法[J]. 计算机应用研究, 1999, 016(005): 7-9.

[199] Zhihua Zhou, Yuan Jiang, Shifu Chen. Extracting symbolic rules from trained neural network ensembles [C]. IOS Press, 2003.

[200] Bo-Jian Hou, Zhi-Hua Zhou. Learning with interpretable structure from rnn[C]. CoRR, vol. abs/1810.10708, 2018.

[201] Zhiwu Xu, Xiongya Hu, Cheng Wen, and Shengchao Qin. Extracting Automata from Neural Networks Using Active Learning[C]. Accepted by The 3rd National Formal Methods and Applications Conference.

[202] Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, Cho-Jui Hsieh. Query-Efficient Hard-label Black-box Attack: An Optimization-based Approach[C]. In International Conference on Learning Representations (ICLR 2018).

[203] Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks[C]. Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2019). Vol. 33. 2019.

[204] Yali Du, Meng Fang, Jinfeng Yi, Jun Cheng, and Dacheng Tao. Towards query efficient black-box attacks: An input-free perspective[C]. In Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security, AISec'18, pages 13-24,

[205] Yucheng Shi, Siyu Wang, and Yahong Han. Curls & whey: Boosting black-box adversarial attacks[C]. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), pp.

6519-6527. 2019.

[206] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, Jianguo Li. Boosting adversarial attacks with momentum[C]. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2018), pp. 9185-9193, 2018.

[207] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks[C]. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), pp. 4312-4321. 2019.

[208] Jinyin Chen, Mengmeng Su, Shijing Shen, Hui Xiong, Haibin Zheng. POBA-GA: Perturbation optimized black-box adversarial attacks via genetic algorithm[C]. Computers & Security 85(2019): 89-106.

[209] Yandong Li, Lijun Li, Liqiang Wang, Tong Zhang, and Boqing Gong. NATTACK: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks[C]. In Proceedings of the 36th International Conference on Machine Learning (ICML 2019), Long Beach, California, USA, 09-15 Jun 2019. PMLR.

[210] Javid Ebrahimi, Anyi Rao, Daniel Lowd, Dejing Dou. HotFlip: White-Box Adversarial Examples for Text Classification[C]. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018) (pp. 31-36).

[211] Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled[C]. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI 2018), pp. 4208-4215. 2018.

[212] Minhao Cheng, Jinfeng Yi, Pin-Yu Chen, Huan Zhang, Cho-Jui Hsieh. Seq2Sick: Evaluating the Robustness of Sequence-to-Sequence Models with Adversarial Examples[C]. In AAAI 2020 (pp. 3601-3608).

作者简介

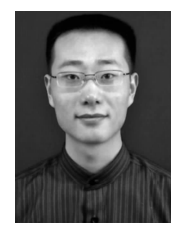
卜 磊 南京大学教授。主要研究领域为软件工程与形式化方法，包括模型检验技术，实时混成系统，信息物理系统等。现任 CCF 系统软件专委会秘书长、形式化方法专委会委员。

陈立前 国防科技大学，副教授，主要研究形式化分析与验证、抽象解释，CCF 形式化专委会委员。

董云卫 西北工大学计算机学院教授，博士生导师。研究方向为智能嵌入软件，信息物理融合系统。中国计算机学会高级会员，形式化专委会常委，嵌入式专委会常委。

黄小炜 利物浦大学副教授。主要研究方向为人工智能安全和验证。

李建霖 中国科学院软件研究所硕士研究生。主要研究方向为形式化方法。



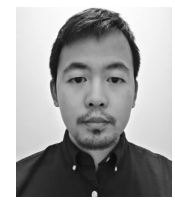
李 钦 华东师范大学软件工程学院副教授。主要研究领域为形式化方法、高可信软件、安全可信人工智能系统等，主要研究方向为人机物融合系统可信建模与验证、安全可信智能系统建模理论与验证方法、多智能体协同决策的形式化建模与分析等。

刘万伟 国防科技大学副教授。主要研究方向为形式化方法，中国计算机学会高级会员，中国计算机学会形式化方法专委会委员。

阮文杰 英国埃克塞特大学计算机系高级讲师（Senior Lecturer）。主要研究领域为深度神经网络的鲁棒性和安全性。

宋　富　上海科技大学，助理教授、研究员。主要研究方向为形式化验证、软件与 AI 安全。中国计算机学会形式化方法专委会委员、中国计算机学会高级会员。

孙有程　贝尔法斯特女王大学助理教授。主要研究方向为人工智能系统的自动化测试和验证。

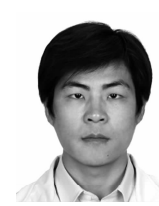


王竟亦　浙江大学研究员。主要研究方向为软件工程、形式化方法、信息物理系统安全、人工智能安全。

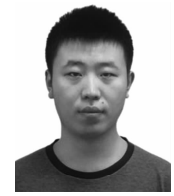
吴　敏　牛津大学博士研究生。主要研究方向为形式化方法、人工智能安全。

许智武　深圳大学副教授。主要研究方向为形式化分析与验证、类型系统、机器学习。形式化方法专委会委员。

薛　白　中国科学院软件研究所副研究员。主要研究方向为混成系统及 AI 形式验证。中国计算机学会计算机科学普及工作委员会执行委员。



杨鹏飞 中国科学院软件研究所博士生。主要研究方向为概率模型检验和 AI 验证。



易新平 利物浦大学助理教授，主要研究方向为信息论与机器学习。

张立军 中国科学院软件研究所研究员。主要研究方向为形式化方法、程序验证。中国计算机学会会员，形式化方法专委会委员。

张 民 华东师范大学软件工程学院副教授，主要研究方向为形式化验证，智能软件可靠性。中国计算机学会会员，形式化方法专委会委员。