

神经网络模型的生成方法及装置、计算机可读存储介质

相关申请的交叉引用

本申请基于申请号为 201910565729.6、申请日为 2019 年 06 月 27 日的中国专利申请提出，并要求该中国专利申请的优先权，该中国专利申请的
5 全部内容在此引入本申请作为参考。

技术领域

本申请涉及计算机领域，尤其涉及一种神经网络模型的生成方法及装置、计算机可读存储介质。

背景技术

10 在相关技术中，搭建神经网络模型的过程中都是由人工调整神经网络架构，这种方法是 100% 手动，是研究人员和机器学习工程师最广泛采用的方法。工作流程非常简单，比如：设计了一个游戏人工智能（AI）的模仿学习流程，然后按顺序迭代不同的神经网络模型架构，训练出该神经网络架构的模仿学习模型权重，应用该权重到游戏中观察其 AI 效果，直到模型
15 满足 AI 的需求。

首先人工设计神经网络架构需要开发人员对深度学习、神经网络架构有较丰富的经验，这样才能够设计出一款适合 AI 需求的神经网络架构。如果缺乏机器学习的经验，那么很难设计出一个性能出色的机器学习模型。其次，基于人工的方案，在把机器学习用于新应用上时，必须投入人力来完成针对新应用的神经网络架构设计。此外，针对同一款应用要想获得一个效果比较好的 AI，需要实验很多神经网络架构，这需要耗费大量的人工
20 开销。

针对上述的问题，相关技术尚未提出有效的解决方案。

发明内容

本申请实施例提供了一种用于游戏中的神经网络模型的生成方法及装置、计算机可读存储介质，以至少解决生成神经网络模型的效率较低的技术问题。

- 5 本申请实施例提供一种用于游戏中的神经网络模型的生成方法，所述神经网络模型用于在游戏中控制游戏元素进行操作；所述方法应用于电子装置，所述方法包括：

10 从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值；其中，所述第一神经网络模型至少是输入层、具有所述超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型；

 获取对所述第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度；

15 在所述第一模型精确度低于目标精确度的情况下，将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值，得到第三神经网络模型；

 其中，使用第四神经网络模型对验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度，所述第四神经网络模型是对所述第三神经网络模型进行训练得到的。

20 本申请实施例还提供了一种用于游戏中的神经网络模型的生成装置，包括：

 确定模块，配置为从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值，其中，所述第一神经网络模型至少是输入层、具有所述超参数值的结构层、和输出层通过所述目标连接方式进行连接得到的神经网络模型；

25 获取模块，配置为获取对所述第一神经网络模型进行训练得到的第二

神经网络模型的第一模型精确度；

更新模块，配置为在所述第一模型精确度低于目标精确度的情况下，将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值，得到第三神经网络模型；

- 5 其中，使用第四神经网络模型对验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度，所述第四神经网络模型是对所述第三神经网络模型进行训练得到的。。

本申请实施例还提供了一种计算机可读存储介质，所述计算机可读存储介质中存储有计算机程序，其中，所述计算机程序被设置为运行时执行
10 上述任一项中所述的方法。

本申请实施例还提供了一种电子装置，包括存储器和处理器，所述存储器中存储有计算机程序，所述处理器被设置为通过所述计算机程序执行上述任一项中所述的方法。

在本申请实施例中，采用从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值，获取对所述第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度，如果第一模型精确度低于目标精确度，那么将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值，得到第三神经网络模型，并且使用对所述第三神经网络模型进行训练得到的第四神经网络模型对所述验证
15 样本集合进行检测的第二模型精确度高于或等于所述目标精确度，如此在超参数值的调整过程中，根据当前模型的模型精确度值调整部分超参数值，能够减小调整的超参数范围，从而提高生成神经网络模型的效率。

附图说明

此处所说明的附图用来提供对本申请实施例的进一步理解，构成本申
25 请实施例的一部分，本申请的示意性实施例及其说明用于解释本申请实施

例，并不构成对本申请实施例的不当限定。在附图中：

图 1 是本申请实施例提供的神经网络模型生成方法的系统架构示意图；

图 2 是本申请实施例提供的一种用于游戏中的神经网络模型的生成方法的示意图；

5 图 3 是本申请实施例提供的一种神经网络模型的生成方法的应用环境示意图；

图 4 是本申请实施例提供各种典型层的超参数类型的示意图；

图 5 是本申请实施例提供的块 A 的残差结构示意图；

图 6 是本申请实施例提供的块 B 的残差结构示意图；

10 图 7 是本申请实施例提供的抽象后的 ResNet50 网络结构示意图；

图 8 是本申请实施例提供的贝叶斯优化算法中配置的神经网络搜索空间定义文件示意图；

图 9 是本申请实施例提供的一种神经网络模型的生成装置的示意图；

15 图 10 是本申请实施例提供的一种神经网络模型的生成方法的实现流程示意图；

图 11 是本申请实施例提供的一种满足某款游戏需求的神经网络结构示意图；

图 12 是本申请实施例提供的一种电子装置的示意图。

具体实施方式

20 为了使本技术领域的人员更好地理解本申请实施例方案，下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本申请一部分的实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都应当属于本申请实施例保护的
25 范围。

需要说明的是，本申请实施例的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象，而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换，以便这里描述的本申请的实施例能够以除了在这里图示或描述的那些以外的顺序实施。此外，术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含，例如，包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

10 参见图 1，图 1 是本申请实施例提供的神经网络模型生成方法的系统架构示意图，如图 1 所示，该系统架构中包括，第一终端 100、服务器 200、网络 300 和第二终端 400，为实现支撑一个示例性应用，第一终端 100 和第二终端 400 分别通过网络 300 连接到服务器 200，第一终端 100 可以是由真实玩家操控的智能终端，该智能终端可以是笔记本电脑，平板电脑，台式计算机，移动设备（例如，移动电话，便携式音乐播放器，个人数字助理，15 专用消息设备，便携式游戏设备）等任意具有屏幕显示功能的终端。第二终端 400 可以是智能机器人，例如可以是游戏机器人。网络 300 可以是广域网或者局域网，又或者是二者的组合，使用无线链路实现数据传输。

服务器 200 可以基于本申请实施例提供的用于游戏中的神经网络模型的生成方法，生成神经网络模型，并采集真实玩家进行游戏时的数据，得到训练数据，进而根据训练数据对生成的神经网络模型进行训练，并将训练好的神经网络模型下发给第二终端 400，第二终端 400 基于该训练好的神经网络模型控制游戏对局中的游戏元素，以完成相应的游戏任务，如人机对战任务、游戏陪玩任务及自动挂机任务等。

25 服务器 200 可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是基于云技术的云服务器。云技术

是指在广域网或局域网内将硬件、软件、网络等系列资源统一起来，实现数据的计算、存储、处理和共享的一种托管技术。在本申请实施例中，当服务器 200 为云服务器时，提供的人工智能云服务可以包括生成神经网络模型，并基于训练数据训练生成的神经网络模型，以使得神经网络模型能够学习交互过程中的操作技巧（反映为人工神经网络模型参数）。

训练数据是由真实玩家控制游戏角色进行战斗时，战斗过程中游戏基本信息、场景实时信息、以及真实玩家的操作数据。其中，游戏基本信息可以包括：真实玩家控制的游戏角色是否胜利、场景尺寸（场景，即游戏中供对象移动的空间，例如战斗游戏中的擂台）、各技能冷却总时间；场景数据包括：我方（即真实玩家控制的游戏角色的位置）位置、敌方（即与真实玩家控制的游戏角色进行战斗的游戏角色）位置、我方各技能的当前冷却时间、敌方各技能的当前冷却时间；操作数据包括：技能使用方式（各技能是否释放、是否进行普通攻击）、摇杆移动角度，真实玩家控制的游戏角色是否跳跃等。

在利用训练数据对生成的神经网络模型进行训练时，首先初始化包括输入、输出、以及神经网络模型参数的损失函数，在人工神经网络模型每次迭代训练过程中，根据将所述训练数据代入所述损失函数，求解所述损失函数取得最小值时对应的神经网络模型参数，并根据所求解的人工神经网络模型参数更新所述神经网络模型。

作为示例，第二终端 400 在获取到训练好的神经网络模型后，执行与第一终端 100 的人机对战任务，第一终端 100 和第二终端 400 输出游戏应用的虚拟场景，其中包括第一对象 110 和第二对象 120，第一对象 110 可以是受用户（也称为真实玩家，以区别于机器人模型）控制的游戏角色，第一对象 110 受控于真实玩家，将响应于真实玩家针对第一终端 100 中的控制器（包括键盘、鼠标和摇杆等）的操作而在虚拟场景中运动，服务器采

集真实玩家的实时游戏数据，并发送至第二终端 400，第二终端 400 基于训练好的神经网络模型来预测第二对象 120 所要执行的操作数据，并通过控制第二对象 120 执行操作数据来实现对应的行为，实现第一对象 110 和第二对象 120 之间的智能化的交互。

5

在本申请实施例中，对服务器生成用于游戏中的神经网络模型的实现方式进行说明。图 2 为本申请实施例提供的用于游戏中的神经网络模型的生成方法，如图 2 所示，该方法包括：

S102，从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值。

其中，第一神经网络模型至少是输入层、具有超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型；

S104，获取对第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度。

其中，第一模型精确度用于指示使用第二神经网络模型对验证样本集合进行检测的精确度。

S106，在第一模型精确度低于目标精确度的情况下，将第一神经网络模型中的部分超参数的超参数值更新为超参数值集合中对应的目标超参数值，得到第三神经网络模型。

其中，使用第四神经网络模型对验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度，所述第四神经网络模型是对所述第三神经网络模型进行训练得到的。

生成第三神经网络模型可以用于在游戏中控制游戏元素进行操作，游戏元素指游戏对局中可被用户或后台控制的元素，包括但不限于游戏角色及游戏对局内的组织单位（如建筑、资源及兵种等）。

在本申请实施例中，上述神经网络模型的生成方法可以应用于如图 3

所示的服务器 200 所构成的硬件环境中。如图 3 所示，服务器 200 的处理流程包括如下步骤：步骤 S202，服务器 200 从超参数值集合 204 中确定第一神经网络模型 206 中的超参数所对应的超参数值，其中，第一神经网络模型是输入层、具有超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型。步骤 S204，服务器 200 获取对第一神经网络模型进行训练得到的第二神经网络模型 208 的第一模型精确度，其中，第一模型精确度用于指示使用第二神经网络模型 208 对验证样本集合进行检测的精确度。步骤 S206，在第一模型精确度低于目标精确度的情况下，服务器 200 将第一神经网络模型 206 中的部分超参数的超参数值更新为超参数值集合中对应的目标超参数值，得到第三神经网络模型 210，其中，使用对第三神经网络模型 210 进行训练得到的第四神经网络模型 212 对验证样本集合进行检测的第二模型精确度高于或等于目标精确度。

在本申请实施例中，上述神经网络模型的生成方法可以但不限于应用于为应用生成神经网络模型的场景中。其中，上述应用可以但不限于为各种类型，例如，在线教育应用、即时通讯应用、社区空间应用、游戏应用、购物应用、浏览器应用、金融应用、多媒体应用、直播应用等。需要说明的是，上述神经网络模型的生成方法在应用在除游戏应用之外的其他应用中时，实际是应用在这些应用的游戏场景中，以提高生成神经网络模型的效率。上述仅是一种示例，本申请实施例中对此不做任何限定。

在本申请实施例中，在机器学习的上下文中，超参数是在开始学习过程之前设置值的参数。而其他参数的值通过训练得出。比如：卷积层的超参数有卷积核的数量，卷积核的通道数，高度，宽度，水平方向的步长，垂直方向的步长等。全连接层的超参数有神经元的数量。激活函数层的超参数有激活函数的类型，函数的参数等。如图 4 所示为各种典型层的超参数类型。

在本申请实施例中，将神经网络模型划分为输入层、结构层、和输出层。其中，输入层和输出层的超参数被确定下来，在生成神经网络模型的过程中只对结构层的超参数进行调整。

在本申请实施例中，每得到一个神经网络模型则使用训练样本集合和
5 验证样本集合对其精确度进行验证，对于精确度不符合条件（即第一模型精确度低于目标精确度）的模型，对结构层的部分超参数进行调整，得到新的模型。将精确度符合条件（即第一模型精确度高于或等于目标精确度）的模型确定为目标模型。

可见，通过上述步骤，将神经网络模型划分为输入层、具有超参数值的
10 结构层、和输出层，对结构层的超参数进行调整，在调整的过程中，根据当前模型的模型精确度值调整结构层的超参数中的部分超参数值，从而减小了调整的超参数范围，从而实现了提高生成神经网络模型的效率的技术效果，进而解决了生成神经网络模型的效率较低的技术问题。

在本申请一些实施例中，结构层包括第一残差网络和第二残差网络，
15 其中，从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数数值包括：

S1，从第一超参数值子集合中确定第一残差网络的第一超参数所对应的
的第一超参数值，其中，第一残差网络所具有的多个超参数包括第一超参数；

20 S2，从第二超参数值子集合中确定第二残差网络的第二超参数所对应的第二超参数值，其中，第二残差网络所具有的多个超参数包括第二超参数，超参数值集合包括第一超参数值子集合和第二超参数值子集合。

在本申请实施例中，以深度残差网络 ResNet50 为基础模型，标准的
ResNet50 网络的结构是一个非常深的网络。可以把其用到的残差结构抽象
25 出来，称为 Block A 和 Block B。如图 5 所示为块（Block）A 的残差结构，

如图 6 所示为 Block B 的残差结构。

在本申请实施例中，第一残差网络可以但不限于为上述 Block A，第二残差网络可以但不限于为上述 Block B。

在本申请实施例中，针对 Block A 的结构可以只探索其第一超参数。针对 Block B 可以探索其第二超参数。

在本申请实施例中，结构层包括多个第一残差网络和多个第二残差网络，超参数值集合包括多个第一超参数值子集合和多个第二超参数值子集合，多个第一残差网络与多个第一超参数值子集合一一对应，多个第二残差网络与多个第二超参数值子集合一一对应，其中，

10 S1，从第一超参数值子集合中确定第一残差网络的第一超参数所对应的第一超参数值包括：从多个第一超参数值子集合中的每个第一超参数值子集合中确定目标第一残差网络的第一超参数所对应的第一超参数值，其中，目标第一残差网络为与每个第一超参数值子集合对应的第一残差网络；

S2，从第二超参数值子集合中确定第二残差网络的第二超参数所对应的第二超参数值包括：从多个第二超参数值子集合中的每个第二超参数值子集合中确定目标第二残差网络的第二超参数所对应的第二超参数值，其中，目标第二残差网络为与每个第二超参数值子集合对应的第二残差网络。

在本申请实施例中，ResNet50 的残差网络包括多个上述 Block A 和多个上述 Block B。如图 7 所示为抽象后的 ResNet50 网络结构。Layer1 是 ResNet50 的输入层。Layer2、Layer4、Layer6 和 Layer8 是 Block A 结构。Layer3、Layer5、Layer7 和 Layer9 是 Block B 结构，其中， $\times 2$ 表示两块 Block B 结构串联。后续层的 Block B $\times 3$ ，Block B $\times 5$ 表示 Block B 结构串联 3 块或 5 块组成一个大的结构。Layer10 是输出层。

在本申请实施例中，每个第一超参数值子集合相当于每个第一超参数的探索空间。每个第二超参数值子集合相当于每个第二超参数的探索空间。

在本申请实施例中，将第一神经网络模型中的部分超参数的超参数值更新为超参数值集合中对应的目标超参数值包括以下至少之一：

S1，将部分第一残差网络的第一超参数所对应的第一超参数值更新为该部分一残差网络所对应的第三超参数值；

5 其中，第三超参数值为第一超参数值子集合中的超参数值；

S2，将部分第二残差网络的第二超参数所对应的第二超参数值更新为该部分第二残差网络所对应的第二超参数值子集合中的第四超参数值。

其中，第四超参数值为第二超参数值子集合中的超参数值。

在本申请实施例中，可以但不限于使用贝叶斯优化算法、网格搜索和
10 随机搜索等等方式对超参数进行更新。

在本申请一些实施例中，以使用贝叶斯优化算法为例。对已经存在的样本点通过高斯过程回归（假设超参数间符合联合高斯分布）计算前面 n 个点的后验概率分布，得到每一个超参数在每一个取值点的期望均值和方差，其中均值代表这个点最终的期望效果，均值越大表示模型最终指标越大，方差表示这个点的效果不确定性，方差越大表示这个点不确定是否可能取得最大值非常值得去探索。可以但不限于使用智能神经网络（NNI, Neural Network Intelligence）机器学习包提供的贝叶斯优化算法。如图 8 所示为贝叶斯优化算法中配置的神经网络搜索空间定义文件。贝叶斯优化算法会根据如图 8 所示为的定义文件推荐出一组神经网络的结构参数。例如
15 layer2/layer4/layer6/layer8 的卷积核尺寸（也即卷积核大小）分别为[16, 16, 64, 128]，layer3/layer5/layer7/layer9 中 Block B 重复的次数分别为[2, 3, 5, 4]。服务器使用这两组参数生成相应的神经网络。

在本申请实施例中，结构层包括第一残差网络和第二残差网络，第一超参数包括第一残差网络所包括的卷积核中第一个卷积核的卷积核尺寸，
25 第二超参数包括第二残差网络的串联数量，其中，从超参数值集合中确定

第一神经网络模型中的超参数所对应的超参数值包括:

S1, 从第三超参数值子集合中确定第一残差网络中第一个卷积核的卷积核尺寸为目标卷积核尺寸值;

S2, 从第四超参数值子集合中确定第二残差网络的串联数量为目标数量, 其中, 超参数值集合包括第三超参数值子集合和第四超参数值子集合。

在本申请实施例中, 根据上述抽象的 ResNet50 网络结构, 可以把搜索空间缩小到 Layer2 到 Layer9。Layer1 和 Layer10 不需要探索。其次可以发现一个网络的卷积核尺寸是影响其执行性能和精度的关键, 因此针对 Block A 的结构可以只探索其卷积核的尺寸。针对 Block B 主要探索其重复结构的数量。探索空间可以通过如下方式确定:

对于 Layer2, Layer4, Layer6, Layer8, 搜索其第一个的卷积层的卷积核尺寸, 因为第一个卷积核的尺寸就会影响到后续的卷积核尺寸。例如: 对于 Layer2 使用的是 Block A 结构, 可以探索其内部第一个卷积层的卷积核尺寸, ResNet50 使用的默认值是 $1 \times 1 \times 64 \times 64$, 可以探索最后一个数值, 探索的空间可以为 [16, 32, 64, 128] (相当于上述第一超参数值子集合), 因此对于 Layer2 的第一个卷积层的卷积核就有如下的组合: $1 \times 1 \times 64 \times 16$, $1 \times 1 \times 64 \times 32$, $1 \times 1 \times 64 \times 64$, $1 \times 1 \times 64 \times 128$ 。其它层的卷积核的参数也可以类似使用 4 个不同的整型参数作为探索项。

对于 Layer3, Layer5, Layer7, Layer9 这四层因为是对 Block B 的重复, 可以探索每个 Layer 重复多少次 Block B 合适, 例如: 对于 Layer3, ResNet50 默认重复了 2 个 Block B 结构, 可以将其搜索空间定义为 [1, 2, 3, 4, 5, 6, 7] (相当于上述第二超参数值子集合)。针对后面的 Layer5, Layer7, Layer9 也可以设置其探索空间为 [1, 2, 3, 4, 5, 6, 7]。

在本申请实施例中, 根据以上搜索空间的定义整个探索空间的范围大概是: $44 \times 74 = 614,656$ 。这个探索空间相对于全新搜索神经网络结构时动

辄数十亿的探索空间来说，降低了 4 个数量级，对于硬件成本的开销大大降低。

在本申请实施例中，获取对第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度包括：

5 S1，使用从目标应用中获取的第一训练样本集合训练第一神经网络模型，得到第二神经网络模型；

 S2，通过第二神经网络模型对从目标应用中获取的验证样本集合中的每个验证样本进行检测，得到验证样本集合中每个验证样本所对应的检测结果，其中，验证样本集合包括从目标应用中获取的多个验证样本和每个
10 验证样本所对应的目标检测结果；

 S3，将匹配数量在多个验证样本的总数量中的占比确定为第一模型精确度，其中，匹配数量用于指示验证样本所对应的检测结果与验证样本所对应的目标检测结果匹配的验证样本的数量。

在本申请实施例中，上述贝叶斯优化算法中需要一个对该神经网络的
15 评估值（即模型的精确度），依据这个评估值才能继续生成优化后的神经网络结构。借助于该评估值可以判断神经网络是否满足需求。首先把训练的样本划分出一部分作为神经网络架构搜索的评估使用。其次，这部分样本也可以划分成训练集和验证集。生成的神经网络模型都会在该训练集上面进行监督学习训练，然后在验证集上面得到一个模型的精确度。使用该精
20 确度作为对神经网络架构的评估值。贝叶斯优化算法会在生成一个神经网络架构后，收到一个对该神经网络架构的评估值。然后贝叶斯优化算法根据该评估值去生成一个更新的神经网络架构。

在本申请实施例中，在将第一神经网络模型中的部分超参数的超参数
 值更新为超参数值集合中对应的目标超参数值，得到第三神经网络模型之
25 后，还包括：

S1, 使用从目标应用中获取的第二训练样本集合对第三神经网络模型进行训练, 得到第四神经网络模型。

本申请实施例中的目标应用可以是游戏应用, 此时第二训练样本集包括真实玩家控制游戏角色进行场景数据以及真实玩家的操作数据, 场景数据包括: 场景尺寸、真实玩家控制的游戏角色的位置、与真实玩家控制的游戏角色进行战斗的游戏角色位置、真实玩家各技能的当前冷却时间等; 操作数据包括: 技能使用方式、摇杆移动角度, 真实玩家控制的游戏角色是否跳跃等。

步骤 S1 在实现时, 使用第二训练样本集包括的场景数据为输入, 并使用第二训练样本集包括的操作数据为输出训练第三神经网络模型。

在训练第三神经网络模型时, 首先初始化包括对应场景数据样本的输入、对应操作数据的输出、以及第三神经网络模型参数的损失函数; 在第三神经网络模型每次迭代训练过程中, 将所述第二训练样本集中的场景数据输入至所述第三神经网络模型, 得到预测操作数据; 并根据从所述训练样本集合中选取的操作数据和所述预测操作数据代入所述损失函数, 求解损失函数取得最小值时对应的第三神经网络模型参数, 根据所求解的第三神经网络模型参数更新第三神经网络模型参数。

S2, 使用第四神经网络模型对目标应用中的待检测数据进行检测。

在本申请实施例中, 生成的第三神经网络模型可以但不限于用于目标应用中的数据进行检测, 比如执行图像检测、图像分割、模仿学习等任务。

在本申请实施例中, 输入层包括卷积层、批量归一化层和激活层, 输出层包括全连接层和损失函数层, 结构层包括四层第一残差网络和四层第二残差网络, 其中, 第一神经网络模型是输入层、具有超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型包括:

输入层、具有超参数值的结构层、和输出层通过串联连接方式进行连

接，得到第一神经网络模型，其中，卷积层、批量归一化层和激活层串联连接，全连接层和损失函数层串联连接，第一层第一残差网络、第一层第二残差网络、第二层第一残差网络、第二层第二残差网络、第三层第一残差网络、第三层第二残差网络、第四层第一残差网络、第四层第二残差网络串联连接，结构层具有的超参数值包括：四层第一残差网络中每一层第一残差网络的卷积核尺寸，四层第二残差网络中每一层串联的第二残差网络的数量。

在本申请实施例中，在上述 ResNet50 的网络架构中，输入层 Layer1 是 ResNet50 的第一个卷积层输入，其可以包括卷积层、批量归一化层（BatchNormalization 层）和激活层。输出层 Layer10 可以包括全连接层和损失函数层（softmax 层）。

在本申请实施例中，结构层包括第一层第一残差网络 Layer2，第一层第二残差网络 Layer3，第二层第一残差网络 Layer4，第二层第二残差网络 Layer5，第三层第一残差网络 Layer6，第三层第二残差网络 Layer7，第四层第一残差网络 Layer8，第四层第二残差网络 Layer9。

需要说明的是，对于前述的各方法实施例，为了简单描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本申请实施例并不受所描述的动作顺序的限制，因为依据本申请实施例，某些步骤可以采用其他顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例所涉及的动作和模块并不一定是本申请实施例所必须的。

通过以上的实施方式的描述，本领域的技术人员可以清楚地了解到根据上述实施例的方法可借助软件加必需的通用硬件平台的方式来实现，当然也可以通过硬件，但很多情况下前者是更佳的实施方式。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分可

以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质（如ROM/RAM、磁碟、光盘）中，包括若干指令用以使得一台终端设备（可以是手机，计算机，服务器，或者网络设备等等）执行本申请各个实施例的方法。

5 根据本申请实施例的另一个方面，还提供了一种用于实施上述神经网络模型的生成方法的神经网络模型的生成装置，如图9所示，该装置包括：

确定模块82，配置为从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值，其中，第一神经网络模型至少是输入层、具有超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型；
10

获取模块84，配置为获取对第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度，其中，第一模型精确度用于指示使用第二神经网络模型对验证样本集合进行检测的精确度；

更新模块86，配置为在第一模型精确度低于目标精确度的情况下，将
15 第一神经网络模型中的部分超参数的超参数值更新为超参数值集合中对应的目标超参数值，得到第三神经网络模型，其中，使用第四神经网络模型对验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度，所述第四神经网络模型是对所述第三神经网络模型进行训练得到的。

在本申请一些实施例中，结构层包括第一残差网络和第二残差网络，
20 其中，确定模块包括：

第一确定单元，配置为从第一超参数值子集合中确定第一残差网络的第一超参数所对应的第一超参数值，其中，第一残差网络所具有的多个超参数包括第一超参数；

第二确定单元，配置为从第二超参数值子集合中确定第二残差网络的第二超参数所对应的第二超参数值，其中，第二残差网络所具有的多个超
25

参数包括第二超参数，超参数值集合包括第一超参数值子集合和第二超参数值子集合。

在本申请一些实施例中，结构层包括多个第一残差网络和多个第二残差网络，超参数值集合包括多个第一超参数值子集合和多个第二超参数值子集合，多个第一残差网络与多个第一超参数值子集合一一对应，多个第二残差网络与多个第二超参数值子集合一一对应，其中，

第一确定单元配置为：从多个第一超参数值子集合中的每个第一超参数值子集合中确定目标第一残差网络的第一超参数所对应的第一超参数值，其中，目标第一残差网络为与每个第一超参数值子集合对应的第一残差网络；

第二确定单元配置为：从多个第二超参数值子集合中的每个第二超参数值子集合中确定目标第二残差网络的第二超参数所对应的第二超参数值，其中，目标第二残差网络为与每个第二超参数值子集合对应的第二残差网络。

在本申请一些实施例中，更新模块包括以下至少之一：

第一更新单元，配置为将部分第一残差网络的第一超参数所对应的第一超参数值更新为部分第一残差网络所对应的第三超参数值，其中，所述第三超参数值为第一超参数值子集合中的超参数值；

第二更新单元，配置为将部分第二残差网络的第二超参数所对应的第二超参数值更新为部分第二残差网络所对应的第四超参数值，其中，所述第四超参数值为第二超参数值子集合中的超参数值。

在本申请一些实施例中，结构层包括第一残差网络和第二残差网络，第一超参数包括第一残差网络所包括的卷积核中第一个卷积核的卷积核尺寸，第二超参数包括第二残差网络的串联数量，其中，确定模块包括：

第三确定单元，配置为从第三超参数值子集合中确定第一残差网络中

第一个卷积核的卷积核尺寸为目标卷积核尺寸值;

第四确定单元,配置为从第四超参数值子集合中确定第二残差网络的串联数量为目标数量,其中,超参数值集合包括第三超参数值子集合和第四超参数值子集合。

5 在本申请一些实施例中,获取模块包括:

训练单元,配置为使用从目标应用中获取的第一训练样本集合训练第一神经网络模型,得到第二神经网络模型;

检测单元,配置为通过第二神经网络模型对从目标应用中获取的验证样本集合中的每个验证样本进行检测,得到验证样本集合中每个验证样本所对应的检测结果,其中,验证样本集合包括从目标应用中获取的多个验证样本和每个验证样本所对应的目标检测结果;

10

第五确定单元,配置为将匹配数量在多个验证样本的总数量中的占比确定为第一模型精确度,其中,匹配数量用于指示验证样本所对应的检测结果与验证样本所对应的目标检测结果匹配的验证样本的数量。

15 在本申请一些实施例中,上述装置还包括:

训练模块,配置为在将第一神经网络模型中的部分超参数的超参数值更新为超参数值集合中对应的目标超参数值,得到第三神经网络模型之后,使用从目标应用中获取的第二训练样本集合对第三神经网络模型进行训练,得到第四神经网络模型;

20 检测模块,配置为使用第四神经网络模型对目标应用中的待检测数据进行检测。

在本申请一些实施例中,第二训练样本集包括场景数据和所述场景数据对应的操作数据,对应地,训练模块包括:

初始化单元,用于初始化包括输入层、输出层、结构层以及所述第三神经网络模型参数的损失函数;

25

输入单元，用于在所述第三神经网络模型每次迭代训练过程中，将所述第二训练样本集中的场景数据输入至所述第三神经网络模型，得到预测操作数据；

损失函数最小化单元，用于根据从所述训练样本集合中选取的操作数据5 和所述预测操作数据代入所述损失函数，求解所述损失函数取得最小值时对应的第三神经网络模型参数；

更新单元，用于根据所求解的第三神经网络模型参数更新所述第三神经网络模型，得到第四神经网络模型。

在本申请一些实施例中，输入层包括卷积层、批量归一化层和激活层，10 输出层包括全连接层和损失函数层，结构层包括四层第一残差网络和四层第二残差网络，其中，第一神经网络模型是输入层、具有超参数值的结构层、和输出层通过目标连接方式进行连接得到的神经网络模型包括：

将所述卷积层、所述批量归一化层和所述激活层进行串联连接，得到所述输入层；

15 将所述全连接层和所述损失函数层进行串联连接，得到所述输出层；

将第一层第一残差网络、第一层第二残差网络、第二层第一残差网络、第二层第二残差网络、第三层第一残差网络、第三层第二残差网络、第四层第一残差网络、第四层第二残差网络进行串联连接，得到所述结构层，所述结构层具有的所述超参数值包括：所述四层第一残差网络中每一层第一20 残差网络的卷积核尺寸，所述四层第二残差网络中每一层串联的第二残差网络的数量；

所述输入层、具有所述超参数值的结构层、和所述输出层通过串联连接方式进行连接，得到所述第一神经网络模型。

本申请实施例的应用环境可以但不限于参照上述实施例中的应用环境。25 本申请实施例提供了用于实施上述实时通信的连接方法的一种应用示

例。

在游戏 AI 领域深度学习得到了广泛的应用。除了大家熟悉的深度强化学习以外，模仿学习也被广泛用于游戏 AI 领域。使用模仿学习的动作克隆方式通过学习人类专家录制的游戏动作策略，然后模仿学习的模型会从保存的专家动作中去匹配自己当前遇到的情形，然后模仿专家采取的动作来应对。

针对相关技术中每款游戏需要开发人员自己设计特定游戏的神经网络架构，上述神经网络模型的生成方法可以但不限于应用于生成用于目标游戏应用的模仿学习的场景中的神经网络模型。在本场景中，提出了一种生成神经网络模型的方法可以针对特定游戏，使用启发式算法和贝叶斯优化算法相结合的方法，实现对神经网络架构的自动探索。通过使用并发的计算资源，可以替代开发人员，实现快速的探索不同的神经网络架构，并找到一个适应特定游戏的优化架构。通过对神经网络架构的探索，可以针对不同的游戏，例如赛车类游戏或者第一人称射击类游戏，使用不同的模仿学习神经网络结构来获取更好的动作效果。

如图 10 所示，上述生成神经网络模型的过程包括如下步骤：

步骤 S901，首先确定模仿学习使用的深度神经网络的探索空间。

这里，神经网络的探索空间定义了可以搜索得到什么样的神经网络，同时也定义了如何描述神经网络结构。神经网络所实现的计算可以抽象成一个无孤立节点的有向无环图（DAG，Directed Acyclic Graph），图的节点代表神经网络的层，边代表数据的流动。每个节点从其前驱节点（有边射入）接收数据，经过计算之后将数据输出到后续节点（有边射出）。理论上说，只要是无孤立节点的 DAG，都是合法的神经网络结构。

从简单的图结构到任意的 DAG 也反映了整个神经网络结构的发展历史。最简单的神经网络是线性链式结构，其对应的图的每个节点最多只有

一个前驱，一个后续，类似于数据结构中的链表。早期的全连接神经网络，卷积神经网络都是这种拓扑结构。Inception、ResNet、DenseNet 中的节点允许有多个前驱，多个后续，从而形成了多分支、跨层连接结构，它们更复杂的图。在神经网络结构中，除了第一个层必须为输入层，最后一个层必须为输出之外，中间的层的类型是可选的，它们代表了各种不同的运算即层的类型。典型有全连接，卷积，反卷积，空洞卷积，池化，激活函数等。

步骤 S902，在神经网络架构的探索空间中，执行贝叶斯优化算法来获取一个神经网络架构。

10 这里，如果基于神经网络架构的探索空间完全从 0 开始探索一个神经网络，其探索空间是非常巨大，需要消耗巨量的硬件资源才能够找到一个适应某种游戏的神经网络结构。这种方法虽然节约了人力的投入，但是对硬件资源消耗巨大，同时也非常费时。因此在本申请实施例中使用 ResNet50 作为神经网络架构基础，然后在此基础上，通过贝叶斯优化算法去探索
15 适合每个游戏的神经网络结构。

步骤 S903，对步骤 S902 生成的神经网络架构进行验证来获得一个对该网络的精确度的评价值。

步骤 S904，检查该网络的评估结果是否满足要求，如果不满足则继续执行步骤 S902 至 S904。

20 步骤 S905，得到满足需要的神经网络架构后，把该神经网络架构送给模仿学习的训练程序。

步骤 S906，模仿学习的训练程序使用该神经网络架构训练得到一个特定游戏的 AI 模型。

在本申请实施例中，根据以上过程搜索到的一个满足某款游戏需求的
25 神经网络结构如图 11 所示，这个网络结构与原始的 ResNet50 最大的不同

在于该 Layer5, Layer7, Layer9 的 Block B 重复的次数有一定的缩减,带来的好处就是可以缩小网络权重文件的大小,节省计算资源的开销。同时 Layer2 的卷积核尺寸也与原始的不同(在图上没有表示出来), Layer2 的卷积核从原始的 $1 \times 1 \times 64 \times 64$ 变为 $1 \times 1 \times 64 \times 32$, 可以看到卷积核尺寸也缩减了一半,这也可以提升计算效率。

通过上述过程,针对不同的游戏需要可以生成不同的模仿学习网络,使用自动神经网络架构搜索算法来完成对神经网络结构的设计。这使得模仿学习的 AI 使用者不需要深入了解深度学习模型的细节,就能够获得一个相对不错的神经网络模型。此外,在探索神经网络架构时可以使用多台服务器并发探索的方式,与人工探索相比,可以更快速的探索不同的神经网络架构,针对特定游戏找出一个优化的网络结构。另一方面,使用贝叶斯优化算法,可以借助于已经探索过的神经网络架构的先验信息去生成一组更好的神经网络架构,比随机搜索和网格搜索的探索效率更高。使用 ResNet50 作为神经网络架构搜索的基础,把 ResNet50 抽象成不同的模块(Block),通过算法去探索不同 Block 间的组合,以及 Block 之间的输入,降低了探索的空间,使得搜索效率极大提升。

根据本申请实施例的又一个方面,还提供了一种用于实施上述神经网络模型的生成的电子装置,如图 12 所示,该电子装置包括:一个或多个(图中仅示出一个)处理器 1102、存储器 1104、传感器 1106、编码器 1108 以及传输装置 1110,该存储器中存储有计算机程序,该处理器被设置为通过计算机程序执行上述任一项方法实施例中的步骤。

在本申请实施例中,上述电子装置可以位于计算机网络的多个网络设备中的至少一个网络设备。

在本申请实施例中,上述处理器可以被设置为通过计算机程序执行以下步骤:

S1, 从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值, 其中, 所述第一神经网络模型是输入层、具有所述超参数值的结构层、和输出层通过所述目标连接方式进行连接得到的神经网络模型;

S2, 获取对所述第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度, 其中, 所述第一模型精确度用于指示使用所述第二神经网络模型对验证样本集合进行检测的精确度;

S3, 在所述第一模型精确度低于目标精确度的情况下, 将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值, 得到第三神经网络模型, 其中, 使用对所述第三神经网络模型进行训练得到的第四神经网络模型对所述验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度。

本领域普通技术人员可以理解, 图 12 所示的结构仅为示意, 电子装置也可以是智能手机 (如 Android 手机、iOS 手机等)、平板电脑、掌上电脑以及移动互联网设备 (Mobile Internet Devices, MID)、PAD 等终端设备。图 12 其并不对上述电子装置的结构造成限定。例如, 电子装置还可包括比图 12 中所示更多或者更少的组件 (如网络接口、显示装置等), 或者具有与图 12 所示不同的配置。

其中, 存储器 1104 可用于存储软件程序以及模块, 如本申请实施例中的神经网络模型的生成方法和装置对应的程序指令/模块, 处理器 1102 通过运行存储在存储器 1104 内的软件程序以及模块, 从而执行各种功能应用以及数据处理, 即实现上述的目标组件的控制方法。存储器 1104 可包括高速随机存储器, 还可以包括非易失性存储器, 如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在一些实施例中, 存储器 1104 可包括相对于处理器 1102 远程设置的存储器, 这些远程存储器可以通过网络连接至终端。上述网络的实例包括但不限于互联网、企业内部网、局域网、

移动通信网及其组合。

上述的传输装置 1110 配置为经由一个网络接收或者发送数据。上述的网络实例可包括有线网络及无线网络。在一个实例中，传输装置 1110 包括一个网络适配器 (Network Interface Controller, NIC)，其可通过网线与其他
5 网络设备与路由器相连从而可与互联网或局域网进行通讯。在一个实例中，传输装置 1110 为射频 (Radio Frequency, RF) 模块，其配置为通过无线方式与互联网进行通讯。

其中，存储器 1104 配置为存储应用程序。

本申请实施例还提供了一种存储介质，该存储介质中存储有计算机程
10 序，其中，该计算机程序被设置为运行时执行上述任一项方法实施例中的步骤。

在本申请实施例中，上述存储介质可以被设置为存储用于执行以下步骤的计算机程序：

S1，从超参数值集合中确定第一神经网络模型中的超参数所对应的超
15 参数值，其中，所述第一神经网络模型至少是输入层、具有所述超参数值的结构层、和输出层通过所述目标连接方式进行连接得到的神经网络模型；

S2，获取对所述第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度，其中，所述第一模型精确度用于指示使用所述第二神经网络模型对验证样本集合进行检测的精确度；

20 S3，在所述第一模型精确度低于目标精确度的情况下，将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值，得到第三神经网络模型，其中，使用第四神经网络模型对验证样本集合进行检测的第二模型精确度高于或等于所述目标精确度，所述第四神经网络模型是对所述第三神经网络模型进行训练得到的。

25 在本申请一些实施例中，存储介质还被设置为存储用于执行上述实施

例中的方法中所包括的步骤的计算机程序。

在本申请实施例中，本领域普通技术人员可以理解上述实施例的各种方法中的全部或部分步骤是可以通程序来指令终端设备相关的硬件来完成，该程序可以存储于一计算机可读存储介质中，存储介质可以包括：闪存盘、只读存储器(Read-Only Memory, ROM)、随机存取器(Random Access Memory, RAM)、磁盘或光盘等。

上述本申请实施例序号仅仅为了描述，不代表实施例的优劣。

上述实施例中的集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用，可以存储在上述计算机可读的存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来，该计算机软件产品存储在存储介质中，包括若干指令用以使得一台或多台计算机设备（可为个人计算机、服务器或者网络设备等）执行本申请各个实施例所述方法的全部或部分步骤。

在本申请的上述实施例中，对各个实施例的描述都各有侧重，某个实施例中未详述的部分，可以参见其他实施例的相关描述。

在本申请所提供的几个实施例中，应该理解到，所揭露的客户端，可以通过其它的方式实现。其中，以上所描述的装置实施例仅仅是示意性的，例如所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，单元或模块的间接耦合或通信连接，可以是电性或其它的形式。

所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地

方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本申请实施例方案的目的。

另外，在本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

以上所述仅是本申请的具体实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本申请原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也应视为本申请的保护范围。

10 工业实用性

本申请实施例中，从超参数值集合中确定第一神经网络模型中的超参数所对应的超参数值；获取对所述第一神经网络模型进行训练得到的第二神经网络模型的第一模型精确度；在所述第一模型精确度低于目标精确度的情况下，将所述第一神经网络模型中的部分超参数的超参数值更新为所述超参数值集合中对应的目标超参数值，得到第三神经网络模型，从而实现了提高生成神经网络模型的效率的技术效果，进而解决了生成神经网络模型的效率较低的技术问题。