

基于遗传算法的神经网络等价模型构建^①鲜开军^{②*} 丁新虎^{*} 朱城超^{**} 朱钟华^{**} 徐东伟^{③**}

(* 中国人民解放军 61478 部队 乌鲁木齐 830001)

(** 浙江工业大学网络空间安全研究院 杭州 310023)

摘 要 神经网络模型结构作为深度学习的重要组成部分,在很大程度上决定着深度学习的性能表现。而目前基于深度学习的应用,大部分都由经典的网络模型修改而来。由于无法获得原神经网络模型结构,本文根据原模型的输入输出数据以及经典的神经网络模型结构,构建了原模型的预测模型。该方法主要通过对预测模型的结构参数进行编码,并利用遗传算法(GA)进行选择、交叉、变异操作,从而构建出原模型的等价模型。对于同一输入数据,等价模型和原模型的输出基本保持一致。本文提出的构建方法在图像分类、信号调制类型分类和网络链路预测领域均取得了较好的效果。

关键词 神经网络;深度学习;遗传算法(GA);等价模型

0 引 言

近年来,深度学习的研究成果已经成功应用在图像识别^[1]、语音识别^[2]和自然语言处理^[3]等诸多领域,具有巨大的发展潜力和社会价值。作为深度学习的重要组成部分,神经网络模型结构在很大程度上决定着深度学习的性能表现^[4]。例如,卷积神经网络(convolutional neural network, CNN)从提出到现在,已经涌现出了许多经典的网络模型^[5],如 AlexNet^[6]、VGGNet^[7]、GoogLeNet^[8]、ResNet^[9]、DenseNet^[10]等,这些网络模型成功地应用在了许多领域,尤其是在图像识别领域^[11]。作为深度学习领域中另一类重要的模型,循环神经网络(recurrent neural network, RNN)擅长处理时间序列数据,基于 RNN 而提出的长短期记忆网络(long short-term memory, LSTM)^[12]和门控循环单元(gated recurrent unit, GRU)^[13]等变体结构,在文本、音频和视频等领域也取得了较好的效果^[14]。

神经网络模型的结构对于模型来说非常重要,因此获取神经网络模型结构并对其进行分析具有重要意义。但是当得到一个黑盒模型时,其模型结构无法得知,而构建等价模型可以模拟出一个等价模型来帮助分析黑盒模型。因此等价模型构建具有实际意义。

遗传算法^[15](genetic algorithm, GA)是模拟达尔文生物进化论的自然选择和遗传学机理的计算模型,是一种通过模拟自然进化过程搜索最优解的方法^[16]。该算法无需准确描述问题的全部特征,且具有较强的通用性和鲁棒性,能较快适应问题和环境的变化,在对较为复杂的组合优化问题求解时,通常能较快地获得较好的优化结果。

构建神经网络等价模型的本质就是一种较为复杂的优化组合问题,且神经网络的可解释性理论研究还处于完善过程中,特别是对于深度神经网络,更是难以找到合适的数学工具对其表达和训练能力进行建模^[17]。

针对上述问题,本文提出了基于遗传算法构建

① 国家自然科学基金(61903334, 61973273)资助项目。

② 男,1974年生,硕士,高级工程师;研究方向:计算机软件应用;E-mail: ten_01@sohu.com。

③ 通信作者, E-mail: dongweixu@zjut.edu.cn。

(收稿日期:2020-09-20)

神经网络等价模型的方法,在原模型的输入输出数据以及经典神经网络模型结构的基础上,将模型的结构参数进行编码,并充分利用遗传算法搜索最优解的能力,构建了原模型的等价模型。

1 整体方案设计

1.1 整体方案

整体设计方案的流程图如图 1 所示。首先提取神经网络模型的结构参数,再选择合适的编码方式将提取的参数进行编码,并根据编码结果生成初始种群,然后通过遗传算法的选择算子、交叉算子、变异算子对种群进行迭代优化,最终选取最优个体作为最终构建的等价模型。

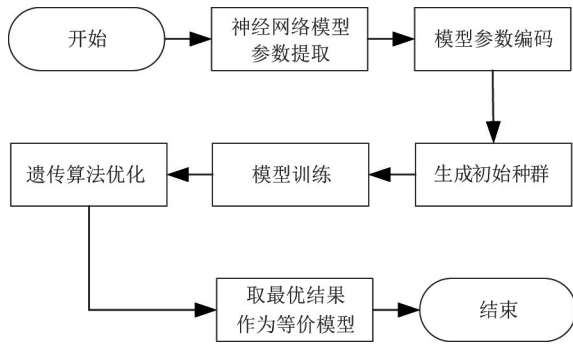


图 1 等价模型构建流程图

1.2 模型结构参数提取

模型的结构参数与模型结构相关,通过优化结构参数可以使得最终构建的等价模型结构类似于原模型,因此首先需要确定所需优化的结构参数。

本文以经典的卷积神经网络结构进行说明。卷积神经网络主要由输入层、卷积层、池化层和输出层组成,而卷积层为最核心的结构。在卷积层中,上一层的特征图被卷积核卷积后,通过激活函数得到输出特征图^[18]。假设卷积层的输入特征图的大小为 $H \times W \times N$,其中 $H \times W$ 表示输入特征图的尺寸, N 是输入特征的通道数, $K \times K \times N$ 表示卷积核尺寸,用 M 表示卷积核数量与输出特征图的通道数^[19],卷积操作如图 2 所示。

卷积神经网络的最后通常由多层全连接网络组成^[18],通过全连接层计算,将特征数据映射到样本

标签空间,再对其进行分类,全连接层中的核心操作是矩阵向量乘积,如图 3 所示。

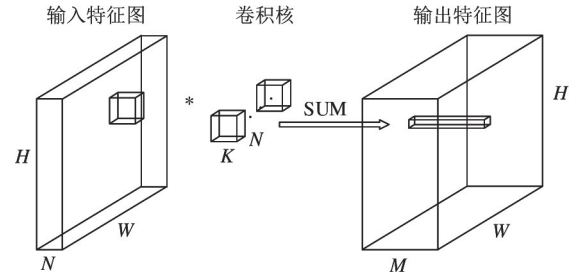


图 2 卷积计算过程

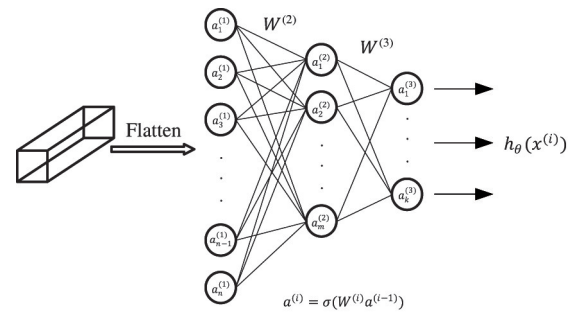


图 3 全连接结构

通过分析可知,卷积神经网络的模型结构与卷积核的尺寸、个数和全连接层的单元个数相关。通过改变这些参数,可以生成不同的模型结构。

1.3 模型参数编码

编码是把一个问题的可行解从其解空间通过相应的转换方式,转换到遗传算法所能处理的搜索空间^[16]。在遗传算法编码方式的选择问题上,Holland^[20]建议采用二进制编码。虽然还有格雷码编码^[21]、浮点编码、树结构编码^[22]、参数动态编码^[23]等多种编码方式,但是考虑到本文所提取出的模型结构参数均为正整数,且编解码方式越简洁,就越方便后续的交叉变异等遗传操作,因此本文选用二进制编码方式。

二进制编码仅使用 $\{0, 1\}$ 2 个符号,将每个个体的参数编码为二进制符号串,式(1)体现了一个二进制串 $(b_n, \dots, b_1, b_0)_2$ 与十进制数 $(a)_{10}$ 相互转换过程。

$$(b_n \cdots b_1 b_0)_2 = \left(\sum_{i=0}^n b_i \cdot 2^i \right)_{10} = (a)_{10} \quad (1)$$

本文以经典网络结构 AlexNet^[6] 为例,提取了 AlexNet 的模型结构参数作为等价模型构建时的优化对象,并利用二进制编码方式对其进行编码。表 1

展示了 AlexNet 原始网络的结构。选择卷积核尺寸、个数和全连接层单元个数作为编码对象,结果如表 2 所示。

表 1 AlexNet 网络结构

层名称	核尺寸	卷积核数/全连接单元数	步长	输出大小
Input				244 × 244 × 3
Conv1	11 × 11 × 3	96	4	55 × 55 × 96
Pool1	3 × 3 × 96	96	2	27 × 27 × 96
Conv2	5 × 5 × 9	256	1	27 × 27 × 96
Pool2	3 × 3 × 96	256	2	13 × 13 × 256
Conv3	3 × 3 × 256	384	1	13 × 13 × 384
Conv4	3 × 3 × 384	384	1	13 × 13 × 384
Conv5	3 × 3 × 384	256	1	13 × 13 × 256
Pool3	3 × 3 × 256	256	2	6 × 6 × 256
FC1	3 × 3 × 96	4096		4096
FC2		4096		4096
FC3		1000		1000
Softmax				ClassNum

表 2 AlexNet 编码结果

层名称	核尺寸 $K \times K$	编码结果	卷积核数/全连接单元数	编码结果
Conv1	11	$(1011)_2$	96	$(0110\ 0000)_2$
Pool1	3	$(0011)_2$	96	$(0110\ 0000)_2$
Conv2	5	$(0101)_2$	256	$(0001\ 1000\ 0000)_2$
Pool2	3	$(0011)_2$	256	$(0001\ 1000\ 0000)_2$
Conv3	3	$(0011)_2$	384	$(0001\ 1000\ 0000)_2$
Conv4	3	$(0011)_2$	384	$(0001\ 1000\ 0000)_2$
Conv5	3	$(0011)_2$	256	$(0001\ 0000\ 0000)_2$
Pool3	3	$(0011)_2$	256	$(0001\ 0000\ 0000)_2$
FC1			4096	$(0001\ 0000\ 0000\ 0000)_2$
FC2			4096	$(0001\ 0000\ 0000\ 0000)_2$
FC3			1000	$(0011\ 1110\ 1000)_2$

1.4 适应度函数选择

适应度函数作为描述个体性能的主要评价指标,依据函数值的大小对个体的优劣进行区分。因此适应度函数的构造至关重要,这对遗传算法的迭代速度以及全局最优解有着直接影响。

本文通过将神经网络模型训练时的损失函数与种群个体的适应度函数建立映射关系,实现种群进化过程中对优化目标函数的寻优。

而神经网络模型的训练过程,就是减小损失函

数值,直到其收敛至基本稳定的过程。多分类问题一般采用交叉熵损失函数^[23],其数学表达式为

$$H(\theta) = -\frac{1}{m} \sum_{i=1}^m [\mathbf{y}^{(i)} \log h_{\theta}(x^{(i)}) + (1 - \mathbf{y}^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] \quad (2)$$

其中, m 为样本数量, $\mathbf{y}^{(i)}$ 为第 i 个样本的 one-hot 向量, $h_{\theta}(x^{(i)})$ 为对第 i 个样本的预测概率结果。

本文在训练等价模型时,采用的数据标签为原模型的预测结果。一般而言,分类问题的最终结果

$y^{(i)}$ 为 one-hot 向量,但从模型的角度而言,模型最终的输出值其实是对各个类别的预测概率值。因此为了避免精度损失,在训练等价模型时采用相对熵^[24]作为其损失函数,其数学表达式如下:

$$H(\theta) = \frac{1}{m} \sum_{i=1}^m \left[y_{\text{pre}}^{(i)} \log \frac{h_{\theta}(x^{(i)})}{y_{\text{pre}}^{(i)}} \right] \quad (3)$$

其中, $y_{\text{pre}}^{(i)}$ 为原模型对于第 i 个样本的预测概率结果, $h_{\theta}(x^{(i)})$ 为构建的等价模型对第 i 个样本的预测概率结果。

对比式(2)与式(3),式(3)使用原模型的预测值 $y_{\text{pre}}^{(i)}$ 替换式(2)中的 $y^{(i)}$,这避免了 $y^{(i)}$ 仅考虑原模型预测结果的最大分量而造成的精度损失情况,有利于等价模型在训练阶段尽可能地学到原模型更多的细节信息,力求其预测值 $h_{\theta}(x^{(i)})$ 的概率分布尽可能逼近 $y_{\text{pre}}^{(i)}$ 。

但作为种群进化依据的适应度函数,其函数值越大表示个体的适应性越好,即构建的等价模型越接近原模型。因此选取模型收敛到一定程度时相对熵的倒数作为最终个体的适应度,用于后续遗传算法对个体进行选择的依据,其数学表达式如下:

$$\text{fitness}(k, i) = \frac{1}{H_{ki}(\theta)} \quad (4)$$

其中, (k, i) 代表第 k 轮迭代中第 i 个个体,而 $H_{ki}(\theta)$ 为对应模型的损失函数值。

1.5 模型训练

当确定神经网络模型结构参数,并对结构参数进行编码后,通过将参数进行不同组合即可产生相应的种群。对种群中每个个体代表的模型进行训练后,根据适应度函数的结果进行后续遗传算法的优化。

本文对于模型的训练过程考虑了训练数据和训练方式。

关于训练数据方面,考虑到原模型最终的输出值是对各个类别预测的概率值,但在转换为预测结果时会经过 one-hot 编码,而 one-hot 编码只关注预测结果中最大的分量,将其置为 1,而将其他类别预测结果置为 0,作为分类问题的最终结果。构建等价模型的目的是为了使构建的模型结构尽可能与原模型一致,因此为了避免 one-hot 编码所造成的转换

损失,在训练等价模型时,直接采用原模型的预测概率值进行损失函数的计算,数学表达式如式(3)所示。

关于训练方式方面,考虑到神经网络模型本身训练所占时长,为了提高训练效率,在对种群中个体对应的模型训练时并不一定要等到损失函数收敛到最优状态,而是可以设定固定训练步数。该举措虽然使模型没有收敛到最优,但加快了遗传迭代速度,且训练步数并不是造成模型结构差异的因素,训练步数为无关变量,对于后续的选择算子进行个体的选择并未造成影响。为了提高最终构建的等价模型精度,可以将通过遗传算法迭代后最终选择的个体进行再训练,直至最优收敛状态,作为构建的等价模型。总体训练流程如图 4 所示。

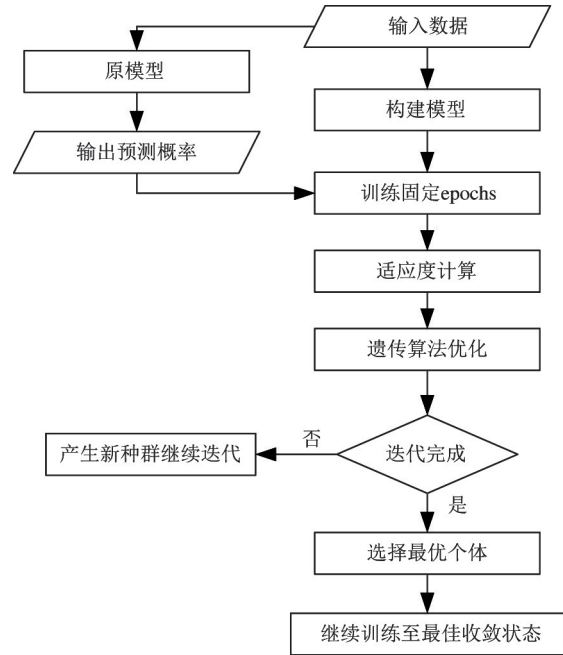


图4 模型训练方式

1.6 遗传算法优化

遗传算法的操作算子包括选择、交叉和变异 3 种基本形式,它们是遗传算法强大搜索能力的核心,是模拟自然选择遗传过程中发生的繁殖、杂交和突变现象的主要载体^[16]。等价模型构建中的处理流程如图 5 所示。

1.6.1 选择算子

选择算子体现适者生存的原理,依据适应度对优质个体进行挑选而抛弃劣质个体,其主要作用是

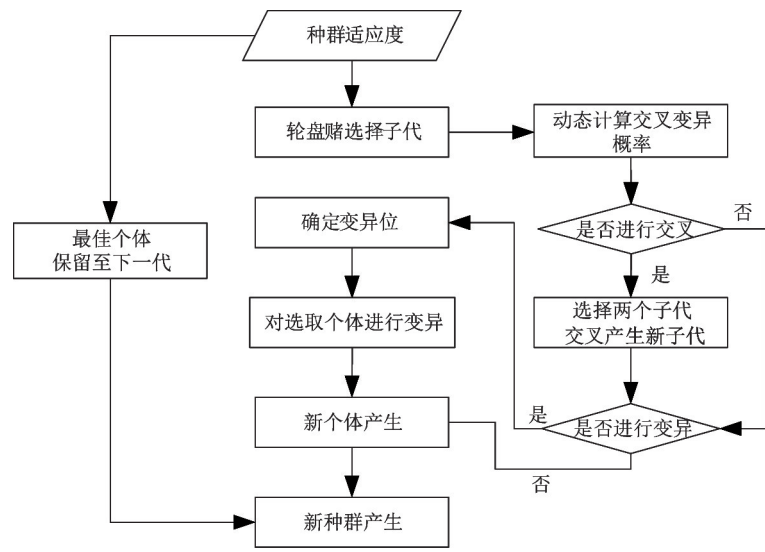


图 5 遗传算法优化流程

避免基因缺失,并提高全局收敛性和计算效率^[16]。常用的选择算子有轮盘赌选择^[25]、线性排序选择^[26]、最优保留选择和锦标赛选择^[27]等。

本文首先采用最优保留选择方式对种群进行选择,直接将种群中适应度最高的个体结构完整地复制到下一代群体中。然后再按照个体的适应度计算出每个个体被选中的概率,如式(5)所示,依据计算出个体被选中的概率值,进行轮盘赌选择,从而完成选择过程。

$$P_{\text{select}}(a_i^k) = \frac{\text{fitness}(a_i^k)}{\sum_{i=1}^M \text{fitness}(a_i^k)} \quad (5)$$

其中, a_i^k 表示第 k 轮迭代中第 i 个个体,而 $P_{\text{select}}(a_i^k)$ 为在 k 轮迭代中第 i 个个体被选中的概率。

1.6.2 交叉算子

在遗传算法中,交叉算子模拟的是自然界生物

遗传基因交换重组的过程,通过交叉能使个体之间的遗传物质进行交换从而产生新的个体,一方面保存了种群中的优良基因,另一方面也增加了种群的多样性。

本文采用二进制编码方式,而适用于二进制编码个体的交叉算子有单点交叉、两点交叉、多点交叉、均匀交叉和算数交叉等^[28]。在交叉过程中,选择合适的交叉概率 P_c 更有利于算法的寻优与收敛, P_c 过小则会在迭代过程中导致新个体产生缓慢,致使优化过程减缓, P_c 过大会导致最优个体丢失等情况出现,无法收敛^[29]。

综合考虑下,本文采取两点交叉法作为交叉算子,在个体的二进制编码串中随机设置两个交叉点,然后将这两个交叉点之间的内容进行交换。表 3 给出了一个具体的实例进行说明,对于选择出的两个个体的全连接层进行交叉,得到两个新的个体。

表 3 两点交叉法

原个体 全连接单元数	编码结果	交叉点	交叉编码结果	新个体 全连接单元数
128	(1000 0000) ₂	5-1	(1001 1110) ₂	158
222	(1101 1110) ₂		(1100 0000) ₂	192

而对于交叉概率 P_c 的计算,则采用文献[29]中提出的自适应调节概率计算方式,如式(6)所示。其中 p_{c1} 与 p_{c2} 为设定的最大与最小交叉概率值, f' 为

进行交叉操作的两个个体中适应度较大的个体, f_{avg} 与 f_{max} 分别为该次迭代中种群的平均适应度与最大适应度。如果进行交叉操作的个体适应度小于平均

适应度,则表示此时处于遗传算法寻优的前期过程,种群的适应度分布较为分散,选择更小的交叉概率有利于保留种群中表现较好的个体;如果进行交叉的个体适应度大于平均适应度时,则表示此时处于遗传算法寻优的后期过程,种群的适应度分布较为集中,选择更大的交叉概率,能在不破坏表现较好个体的前提下,提高新个体的产生概率,可以在一定程度上避免陷入局部最优的困境。

$$P_c = \begin{cases} \frac{p_{c1} + p_{c2}}{2} + \frac{p_{c1} - p_{c2}}{2} \times \sin\left(\frac{f' - f_{avg}}{f_{max} - f_{avg}} \times \frac{\pi}{2}\right) & f' \geq f_{avg} \\ p_{c1} & f' < f_{avg} \end{cases} \quad (6)$$

1.6.3 变异算子

变异体现了遗传过程中的突变现象,在迭代优化过程中能对个体失去的或未开发的遗传物质进行一定程度的恢复,防止种群过早收敛^[16]。

变异运算就是将个体的编码串中某些基因座上的基因值用该基因座上的其他等位基因来替换,从而形成新的个体。常用的变异方式有基本位变异、

均匀变异、边界变异、高斯近似变异等^[30]。同时在变异过程中,变异概率 P_m 的取值对于遗传算法的寻优过程与最终收敛结果有着同样重要的影响, P_m 过小不容易诞生新的个体,而 P_m 过大则会影响遗传算法的稳定性。

本文采用的编码方式为二进制编码,因此选择基本位变异方式即可。首先计算变异概率 P_m ,当结果满足变异条件后;再通过计算个体编码的变异位点,对其进行取反操作完成变异。表 4 给出了一个具体的实例进行说明,对于一个全连接层单元数量做变异操作,生成新的变异个体。

计算变异概率 P_m 的方式选用文献[29]中提出的自适应变异概率方法,其计算方式如式(7)所示,其中 P_{m1} 与 P_{m2} 为设定的最大最小变异概率值, f 为当前个体的适应度值, f_{avg} 与 f_{max} 分别为该次迭代中种群的平均适应度与最大适应度。如果个体的适应度大于平均适应度,则表示此时处于遗传算法寻优的后期过程,种群趋于稳定,个体的表现均较为理想,因此加大变异概率更有利于全新个体的产生。

$$P_m = \begin{cases} \frac{p_{m1} + p_{m2}}{2} + \frac{p_{m1} - p_{m2}}{2} \times \sin\left(\frac{f - f_{avg}}{f_{max} - f_{avg}} \times \frac{\pi}{2}\right) & f \geq f_{avg} \\ p_{m1} & f < f_{avg} \end{cases} \quad (7)$$

表 4 基本位变异

原个体 全连接单元数	编码结果	变异位	变异编码结果	新个体 全连接单元数
1000	(0011 1110 1000) ₂	6	(0011 1100 1000) ₂	968

而当确定变异概率 P_m 后,对于变异位的确定再做进一步考虑。由于神经网络对于结构的敏感性,当变异位确定为编码的高位时,变异结果对于模型结构的变动较为剧烈,容易造成整体结构的不稳定。因此本文提出通过结合编码的位置信息来确定变异位,使得随着编码位的增长而被选中为变异位的概率减小,计算方式如式(8)所示。

$$Loc = \max\{p_l\} = \max\left\{\frac{1}{1+l} \times \text{random}(0,1)\right\} \\ l = 1, 2, \dots, L \quad (8)$$

其中 p_l 表示编码中第 l 位被选中的概率, $\text{random}(0,1)$

表示生成 0~1 之间的随机数, L 表示编码的总位数。

2 实验部分

根据本文所提出的基于遗传算法的神经网络等价模型构建方法,在图像分类、通信信号调制类型分类、网络链路预测领域^[31]进行了模型构建实验。本文就网络链路领域的模型构建结果进行展示说明。

网络的链路预测问题主要是通过利用网络结构等信息,预测网络中缺失的链路或未来可能出现的链路。在图深度学习^[32]中,链路预测问题主要通过图自编码器模型^[33]解决。本文利用 Apache 数据中

程序员之间的 Email 通讯数据,构建一个 3000 个节点左右的 Email 通讯网络,并使用链路预测模型对该通讯网络进行链路预测。

表 5 展示了多个链路预测模型及其对应的模型结构参数。

确定上述模型的可提取结构参数后,根据原模型的参数值,对模型参数的编码范围进行确定,并在

参数范围内随机生成种群并对其进行编码,如表 6 所示。

实验中设定遗传算法迭代次数为 30 次,种群规模为 20,最小交叉概率 p_{c1} 和最大交叉概率 p_{c2} 分别为 0.5 和 0.9,最小变异概率 p_{m1} 和最大变异概率 p_{m2} 分别为 0.1 和 0.2,对种群进行迭代优化,各个模型的迭代结果如图 6 所示。

表 5 链路预测模型

模型名称	层细节	可提取参数
图自编码器(GAE)	GCN	第 1 层 GCN 隐藏层维度 $hidden_1$
变分图自编码器(VGAE)		第 2 层 GCN 输出层维度 $hidden_2$
多任务图自编码器(MTGAE)	AutoEncoder	第 1 层编解码层单元数 $encode_1$
		第 1 层编解码层单元数 $encode_2$
对抗图自编码器(ARGA-AE)	GCN, MLP	第 1 层 GCN 隐藏层维度 $hidden_1$
对抗变分图自编码器(ARGA-VAE)		第 2 层 GCN 输出层维度 $hidden_2$
		第 1 层鉴别器隐藏层维度 $hidden_2$ 第 2 层鉴别器隐藏层维度 $hidden_3$

表 6 链路预测模型参数范围

模型名称	参数 1 范围选取	参数 2 范围选取	参数 3 范围选取
GAE	$hidden_1 : [8, 56]$	$hidden_2 : [8, 40]$	—
VGAE			
MTGAE	$encode_1 : [128, 512]$	$encode_2 : [64, 256]$	—
ARGA-AE	$hidden_1 : [8, 56]$	$hidden_2 : [8, 56]$	$hidden_3 : [32, 96]$
ARGA-VAE			

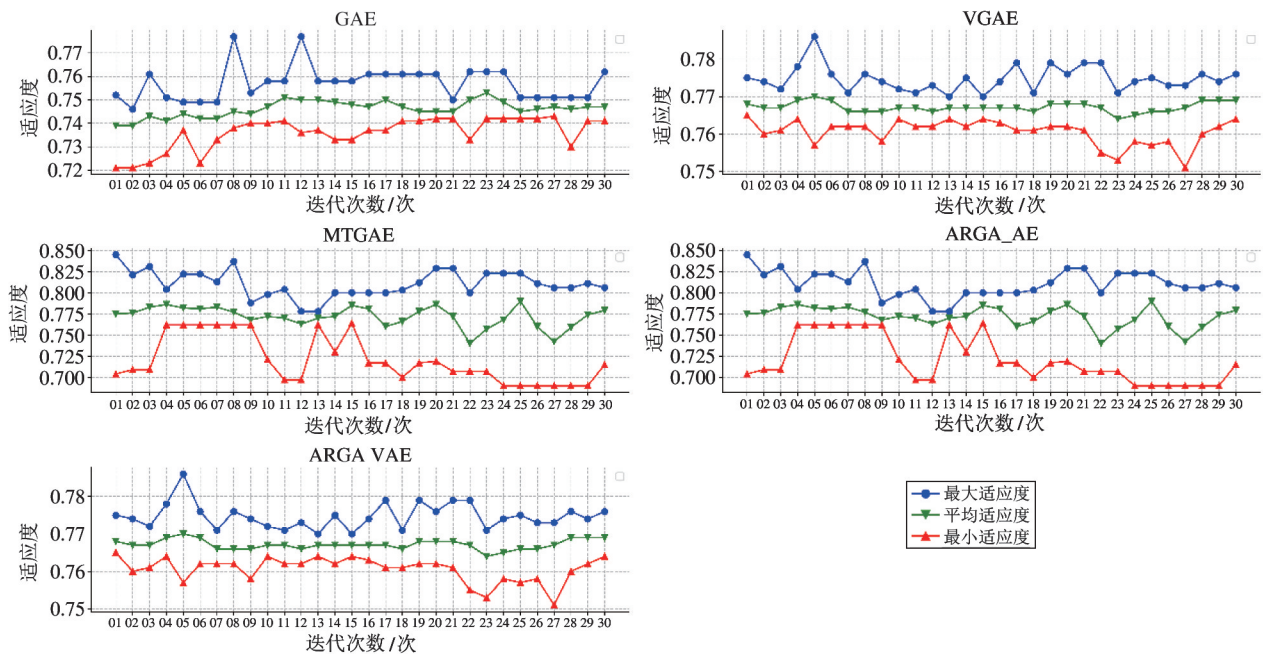


图 6 遗传算法迭代结果

在迭代结束后,取最优参数构建的等价模型进行再训练,直至最优状态,最终的曲线下面积(AUC)与平均精度(AP)结果与原模型对比结果如表7所示。从表中的结果分析可知,构建的等价模

型在结构参数上虽然与原模型存在差别,但是在预测精度方面与原模型基本一致,由此证明了本文提出的等价模型构建方法的有效性。

表7 原模型与等价模型对比

模型	原模型参数	原模型 AUC/AP	等价模型参数	等价模型 AUC/AP
GAE	$hidden_1:32$	0.970/0.974	$hidden_1:40$	0.910/0.961
	$hidden_2:16$		$hidden_2:14$	
VGAE	$hidden_1:32$	0.965/0.972	$hidden_1:33$	0.917/0.965
	$hidden_2:16$		$hidden_2:36$	
MTGAE	$encode_1:256$	0.966/0.366	$encode_1:374$	0.971/0.549
	$encode_1:128$		$encode_1:194$	
ARGA-AE	$hidden_1:32$	0.969/0.974	$hidden_1:40$	0.873/0.953
	$hidden_2:32$		$hidden_2:27$	
	$hidden_3:64$		$hidden_3:65$	
ARGA-VAE	$hidden_1:32$	0.937/0.944	$hidden_1:8$	0.907/0.964
	$hidden_2:32$		$hidden_2:28$	
	$hidden_3:64$		$hidden_3:96$	

3 结 论

在无法获取准确神经网络模型结构的情况下,本文充分依据原模型的输入输出数据与经典神经网络模型结构,并结合遗传算法提出了基于遗传算法的神经网络等价模型构建方法,实现在输入数据相同的情况下,保持原模型的输出数据与等价模型的输出数据基本一致。实验结果表明,本文提出的方法能构建出原神经网络的等价模型,且精度具有一定的保障。然而,此构建方式在原模型预测精度不高的情况下,构建出的等价模型效果也不理想,因此今后的工作将重点放在提升构建的等价模型拟合精度方面。

参考文献

- [1] Cheng F, Zhang H, Fan W, et al. Image recognition technology based on deep learning[J]. *Wireless Personal Communications*, 2018, 102(2): 1917-1933
- [2] Nassif A B, Shahin I, Attali I, et al. Speech recognition using deep neural networks: a systematic review[J]. *IEEE Access*, 2019, 7: 19143-19165
- [3] Chowdhary K R. Natural Language Processing[M]. New Delhi: Springer, 2020: 603-649
- [4] 韩伟,张雄伟,张炜,等. 深度学习中的经典网络模型及训练方法[J]. *军事通信技术*, 2016, 37(1): 90-97
- [5] Gu J, Wang Z, Kuen J, et al. Recent advances in convolutional neural networks[J]. *Pattern Recognition*, 2018, 77: 354-377
- [6] Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge[J]. *International Journal of Computer Vision*, 2015, 115(3): 211-252
- [7] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. *arXiv*:1409.1556, 2014
- [8] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, 2015: 1-9
- [9] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 770-778
- [10] Huang G, Liu Z, Laurens V D M, et al. Densely connected convolutional networks[C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, USA, 2017: 4700-4708
- [11] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. *Communications of the ACM*, 2017, 60(6): 84-90
- [12] Hochreiter S, Schmidhuber J. Long short-term memory

- [J]. *Neural Computation*, 1997, 9(8): 1735-1780
- [13] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation [J]. *arXiv*:1406.1078, 2014
- [14] 杨丽, 吴雨茜, 王俊丽, 等. 循环神经网络研究综述 [J]. *计算机应用*, 2018, 38(S2):1-6, 26
- [15] Whitley D. A genetic algorithm tutorial [J]. *Statistics and Computing*, 1994, 4(2): 65-85
- [16] 马永杰, 云文霞. 遗传算法研究进展 [J]. *计算机应用研究*, 2012, 29(4):1201-1206, 1210
- [17] Hooker S, Erhan D, Kindermans P J, et al. A benchmark for interpretability methods in deep neural networks [C] // *Advances in Neural Information Processing Systems*, Vancouver, Canada, 2019: 9737-9748
- [18] 常亮, 邓小明, 周明全, 等. 图像理解中的卷积神经网络 [J]. *自动化学报*, 2016, 42(9):1300-1312
- [19] 毕鹏程, 罗健欣, 陈卫卫. 轻量化卷积神经网络技术研究 [J]. *计算机工程与应用*, 2019, 55(16):25-35
- [20] Holland J H. *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence* [M]. Cambridge: MIT Press, 1992
- [21] Mei D, Du X, Chen Z. Optimization of dynamic parameters for a traction-type passenger elevator using a dynamic byte coding genetic algorithm [J]. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 2009, 223(3): 595-605
- [22] Gao C, Cai M, Chen H. QoS-aware service composition based on tree-coded genetic algorithm [C] // *The 31st Annual International Computer Software and Applications Conference*, Beijing, China, 2007: 361-367
- [23] De Boer P T, Kroese D P, Mannor S, et al. A tutorial on the cross-entropy method [J]. *Annals of Operations Research*, 2005, 134(1): 19-67
- [24] Robertson J C, Tallman E W, Whiteman C H. Forecasting using relative entropy [J]. *Journal of Money, Credit and Banking*, 2005, 37(3): 383-401
- [25] Yu F, Fu X, Li H, et al. Improved roulette wheel selection-based genetic algorithm for TSP [C] // *2016 International Conference on Network and Information Systems for Computers*, Wuhan, China, 2016: 151-154
- [26] Baker J E. Adaptive selection methods for genetic algorithms [C] // *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, Pittsburgh, USA, 1985: 104-111
- [27] Goldberg D E, Deb K. A comparative analysis of selection schemes used in genetic algorithms [J]. *Foundations of Genetic Algorithms*, 1991, 1: 69-93
- [28] 李书全, 孙雪, 孙德辉, 等. 遗传算法中的交叉算子的述评 [J]. *计算机工程与应用*, 2012, 48(1):36-39
- [29] 徐力, 刘云华, 王启富. 自适应遗传算法在机器人路径规划的应用 [J]. *计算机工程与应用*, 2020, 56(18): 36-41
- [30] Hong T P, Wang H S, Chen W C. Simultaneously applying multiple mutation operators in genetic algorithms [J]. *Journal of Heuristics*, 2000, 6(4): 439-455
- [31] Liben-Nowell D, Kleinberg J. The link-prediction problem for social networks [J]. *Journal of the American Society for Information Science and Technology*, 2007, 58(7): 1019-1031
- [32] Zhang Z, Cui P, Zhu W. Deep learning on graphs: a survey [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, doi:10.1109/TKDE
- [33] Kipf T N, Welling M. Variational graph auto-encoders [J]. *arXiv*:1611.07308, 2016

Neural network equivalent model construction based on genetic algorithm

Xian Kaijun*, Ding Xinhua*, Zhu Chengchao**, Zhu Zhonghua**, Xu Dongwei**

(* Unit 61478 of PLA, Urumqi 830001)

(** Institute of Cyberspace Security, Zhejiang University of Technology, Hangzhou 310023)

Abstract

As an important part of deep learning, the structure of neural network model determines the performance of deep learning to a large extent. At present, most applications based on deep learning are modified from classic network models. Since the structure of original neural network model cannot be obtained, this paper constructs a prediction model of the original model based on the input and output data of the original model and the classic structure of neural network model. This method mainly encodes the structural parameters of the prediction model, and uses genetic algorithm (GA) to perform selection, crossover, and mutation operations to construct an equivalent model of the original model. For the same input data, the output of the equivalent model and the original model are basically the same. The construction method proposed in this paper has achieved good results in the fields of image classification, signal modulation type classification and network link prediction.

Key words: neural network, deep learning, genetic algorithm (GA), equivalent model