

基于可视化的卷积神经网络优化方法研究

王悦, 李京

(中国科学技术大学计算机科学与技术学院, 安徽合肥 230027)

摘要: 随着计算机算力的提升, 深度学习的应用范围越来越广, 深度学习模型的设计和调优变得困难, 对于复杂模型, 只对一层网络进行调整可能就导致差异显著的结果。众多研究者往往根据历史经验调参, 进行了大量试错, 耗费了大量的时间和精力。为此根据卷积神经网络模型的数据特征, 提出一种基于可视化的辅助调参的方法。通过可视化手段剖析卷积神经网络内部数据, 分析其代表的信息, 从而快速定位模型故障, 实现有针对性地调参, 降低了研究者在调参时的工作难度, 提升了工作效率。

关键词: 卷积神经网络; 调参; 可视化; 层次聚类法; 核密度估计; 生成对抗网络

中图分类号: TP183 **文献标识码:** A **doi:** 10.3969/j.issn.0253-2778.2020.07.013

引用格式: 王悦, 李京. 基于可视化的卷积神经网络优化方法研究[J]. 中国科学技术大学学报, 2020, 50(7): 959-967.

WANG Yue, LI Jing. Research on optimization method of convolutional neural network based on visualization[J]. Journal of University of Science and Technology of China, 2020, 50(7): 959-967.

Research on optimization method of convolutional neural network based on visualization

WANG Yue, LI Jing

(School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

Abstract: With the lifting force computer calculation, the application range of the depth of learning more and more widely. However, the design and tuning of deep learning models is very difficult. For complex models, adjusting only one layer of the network may lead to very different results. Many researchers usually adjust their parameters based on past experience, make a lot of trial and error, and wasting a lot of time and energy. Based on the data characteristics of the convolutional neural network model, this paper proposes a method of auxiliary parameter adjustment based on visualization. Analyze the internal data of the convolutional neural network by visualization and analyze the information represented by it, so as to quickly locate the model fault, realize targeted parameter adjustment, reduce the difficulty of researchers in parameter adjustment, and improve work efficiency.

Key words: convolutional neural networks; parameter tuning; visualization; hierarchical clustering; kernel density estimation; generative adversarial networks

0 引言

深度学习^[1]作为能够自主提取数据特征的算法被广泛应用于各领域, 包括图像处理、自然语言处理、语音识别、药物合成等。在图像领域中, 以卷积神经网络^[2]为代表的一系列模型架构, 在计算机视觉领域发挥了巨大的作用, 包括图像识别、目标检测、图像生成、目标分割等。随着应用场景的拓展, 越来越多的研究人员针对实际情况将深度学习应用到工业界。到目前为止, 尽管已经出现了许多著名的模型, 并且在相应数据集上有着优异的表现,

但还是不能满足实际需求, 因此研究人员需要移植现有模型, 并针对自身实际应用场景, 进行修改优化, 甚至需要重新设计网络模型, 逐步摸索建立起有效的神经网络模型。

在进行模型调优的过程中, 损失函数、准确率等数值有助于判断模型的效果, 但是无法推断出导致模型不理想的原因。研究人员往往根据丰富的经验, 进行大量的尝试, 寻找合适的网络模型, 但是需要耗费大量的时间开销以及计算资源的消耗。

He等^[3]在使用卷积神经网络识别图像领域提出了关于 BatchSize、学习率、数据增强等方面的处

收稿日期: 2020-05-24; 修回日期: 2020-06-24

基金项目: 中国科学院战略性先导科技专项(XDA/B19020102)资助。

作者简介: 王悦, 男, 1994年生, 硕士。研究方向: 机器学习。E-mail: cahhbwy@mail.ustc.edu.cn

通讯作者: 李京, 博士教授。E-mail: lj@ustc.edu.cn

理技巧,将 ResNet50 的 Top1 准确率从 75.3% 提升至 79.29%。关于可视化实现, Krizhevsky 等^[4]对卷积核进行可视化, Zeiler 等^[5]使用反卷积可视化提取的特征, Yosinski^[6]设计了 deep-visualization-toolbox 工具可视化卷积神经网络中间层输出; Liu 等^[7]提出一个交互式的可视化分析系统 CNNVis, 采用了构建 DAG 填充矩阵和边捆绑的方法, 帮助研究人员理解分析卷积神经网络, 并设计了 BaseCNN, 使用 CNNVis 改进模型, 使其在 CIFAR10 数据集的分类错误率从 11.32% 降低至 9.43%。

在图像领域的模型中, 常见网络层有卷积层、反卷积层^[8]、激活层^[9]、批归一化层^[10]、池化层^[11]、全连接层、Dropout 层^[12]、Reshape 层等, 其中包含大量参数的模型为卷积层、反卷积层、全连接层, 卷积层和反卷积层具有相似的结构。卷积与反卷积的参数包含 kernel(卷积核)和 bias(偏置项)两部分, kernel 为 4 维结构, 依次为 $height_{kernel}$ (卷积核高度)、 $width_{kernel}$ (卷积核宽度)、 $channel_{in}$ (输入图像通道数)、 $channel_{out}$ (输出图像通道数)。沿着 width 和 height 维度可直接查看卷积核的特征; kernel 为一维结构, 长度等于 $channel_{out}$ 。全连接层参数包含 weight(权重)和 bias(偏置项), weight 为二维结构, 依次为 $feature_{in}$ (输入特征长度)、 $feature_{out}$ (输出特征长度), 整体可直接转化为图像查看。

卷积神经网络等处理图像的模型, 其中间层输出数据具有共性, 主要分成两种: 一种为 4 维结构, 依次为 batch(样本数)、height(图像高度)、width(图像宽度)、channel(图像通道数), 其中 height 和 width 都包含了图像的空域特征, 适合直接转化为图像的方式查看; 另一种为二维结构, 依次为 batch、feature, 整体可转化为二维图像直接展示。

本文提出一种易于实现、能够快速分析模型优缺点的方法。根据卷积神经网络模型中数据的特征, 选择可视化^[13]方法, 结合聚类分析和核密度估计分析, 定位不良模型中的故障, 实现高效调优神经网络模型。本文主要研究步骤为可视化方案设计、基于聚类的特征提取水平评估、基于统计的模型参数分析、针对 TensorFlow^[14]模型的可视化实现、基于可视化结果的模型分析与调优。主要研究对象为图像领域中的使用生成对抗网络生成 CelebA^[15]人脸数据。本文创新点如下:

(I) 使用层次聚类法^[16], 对模型中间层提取到的特征进行分析, 达到评估该层效果的目的。

(II) 使用核密度估计法^[17], 对模型参数分析其分布规律, 辅助判断模型的优劣。

(III) 实现了简单易行的可视化方法, 并根据可视化结果, 有针对性地调优神经网络。

1 基于可视化的模型评估方法

本节将针对两种中间层输出、两类网络层参数, 介绍结合层次聚类和核密度估计的可视化分析方法, 并介绍根据可视化结果分析模型故障的方法。

1.1 中间层输出可视化

1.1.1 4 维结构数据可视化

4 维结构数据在 height 和 width 维度上包含空域特征, 因此可直接将数据转化成图像进行展示。沿 batch 和 channel 维度进行切片, 即可得到 $batch \times channel$ 个 $height \times width$ 大小的图像。将这些图像横向为 channel、纵向为 batch 排列, 将得到 4 维结构数据的可视化结果, 如图 1 所示。

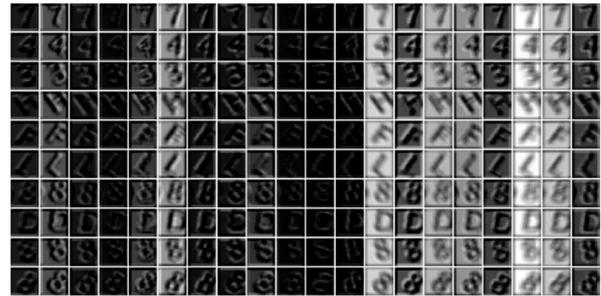


图 1 中间层输出 4 维结构数据可视化

Fig. 1 Visualization of 4-dimensional structure data output by the hidden layer

不同 channel 表示不同的特征, 从图 1 可看出, 不同列之间整体相似但是细节上有明显差异, 不同列则对应不同的特征。本文采用构建层次聚类树的方法, 度量特征之间的相似程度。特征图像的大小相同, 并且整体相似, 需要一种算法能够同时度量绝对数值大小和相对的数值变化, 因此本文选择余弦距离^[18]的计算方法。余弦距离的计算公式为

$$distance(A, B) = 1 - \frac{\sum_{i=1}^n \sum_{j=1}^m A_{i,j} \times B_{i,j}}{\sqrt{\sum_{j=1}^m (A_{i,j})^2} \times \sqrt{\sum_{j=1}^m (B_{i,j})^2}} \quad (1)$$

式中, 图像 A、B 的大小均为 n 行 m 列, $A_{i,j}$ 表示图像 A 的第 (i, j) 个像素, $B_{i,j}$ 表示图像 B 的第 (i, j) 个像素。余弦距离的取值范围为 $0 \sim 2$, 数值越小表示二者越相似。考虑到某些特征图可能全为 0, 计算过程中会导致式(1)分母为 0, 这种特征图实际上是模型特征提取无效的结果, 因此对这种特征图与其他特征图的距离定义为最大值 2。

按照层次聚类法构建层次聚类树, 聚类过程中 linkage 选择 single 单链法, 得到图 2 可视化结果。两个通道在层次聚类树中合并的越早, 说明这两个通道的特征相似程度越高, 如图 2 中 17 和 18 通道; 两个通道在层次聚类树中合并的越晚, 说明两个通道的特征差异性越大, 如 10 通道和其他通道。不同通道的数据分布也不同, 本文采用核密度估计, 绘制概率密度分布图, 分析不同通道的数值分布。如图 3 所示, 每条曲线表示一个通道的数值分布。

1.1.2 二维结构数据可视化

二维结构的数据一般情况下表达的是较高抽象程度的特征, 每个样本都由一个特征向量表示, 对其进行直接可视化, 横向为 feature, 纵向为 batch, 用热力图的方式进行渲染, 得到图 4 所示结果。与 1.1.1 节类似, 不同 feature 之间可用层次聚类法度量相似程度, 绘制层次聚类树进行分析。

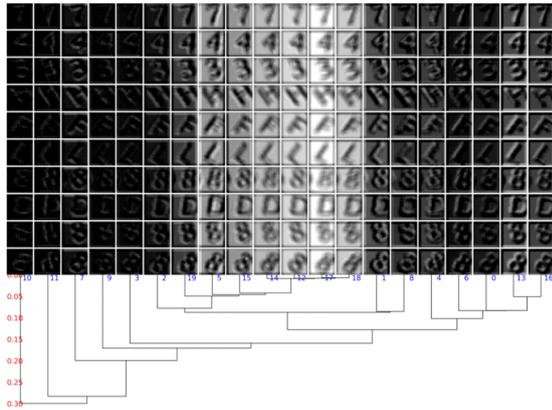


图 2 中间层输出 4 维结构数据结合层次聚类的可视化
 Fig. 2 Visualization of 4-dimensional structure data combined with visualization of hierarchical clustering trees output by the hidden layer

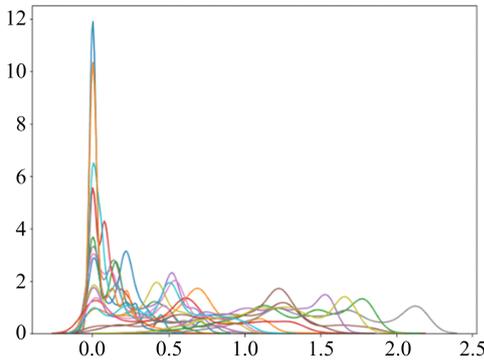


图 3 中间层输出 4 维结构数据不同通道数值分布
 Fig. 3 The distribution of 4-dimensional structure data of different channels output by the hidden layer

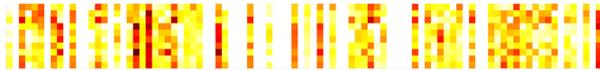


图 4 中间层输出 2 维结构数据可视化(部分)
 Fig. 4 Visualization of 2-dimensional structure data output by the hidden layer (part)

通常, feature 维度数值较大, 运用这种直接可视化的方法得到的图像不方便查看, 对中间层二维结构数据的分析, 更加注重不同 feature 的数值分布. 运用核密度估计法, 绘制不同通道的概率密度曲线, 如图 5 所示.

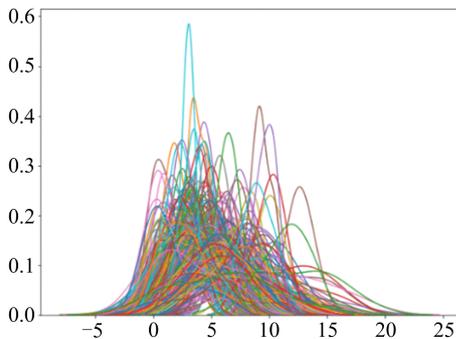


图 5 中间层输出二维结构不同特征数值分布
 Fig. 5 The distribution of 2-dimensional structure data of different features output by the hidden layer

1.2 参数可视化

1.2.1 卷积、反卷积层参数可视化

卷积、反卷积层的参数可视化, 沿 $channel_{in}$ 和 $channel_{out}$ 维度切片, 得到 $channel_{in} \times channel_{out}$ 个 $height_{kernel} \times width_{kernel}$ 大小的卷积单元, 将每个卷积单元数据缩放到 0~1 之间, 按照横向为 $channel_{in}$ 、纵向为 $channel_{out}$ 排列, 并将 bias 拼接到行尾, 进行可视化, 得到图 6 所示结果. 图 6 第一列至倒数第二列, 每个小方格表示一个卷积单元, 最后一列表示偏置项的大小.

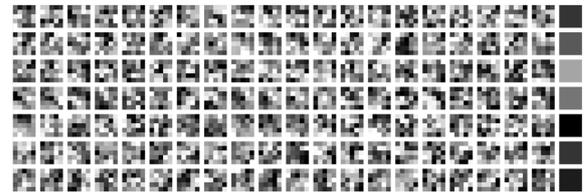


图 6 卷积、反卷积参数可视化(部分)
 Fig. 6 Visualization of convolution and deconvolution parameters

沿 $channel_{out}$ 分析卷积核的相似性, 绘制层次聚类树, 得到图 7 所示结果.

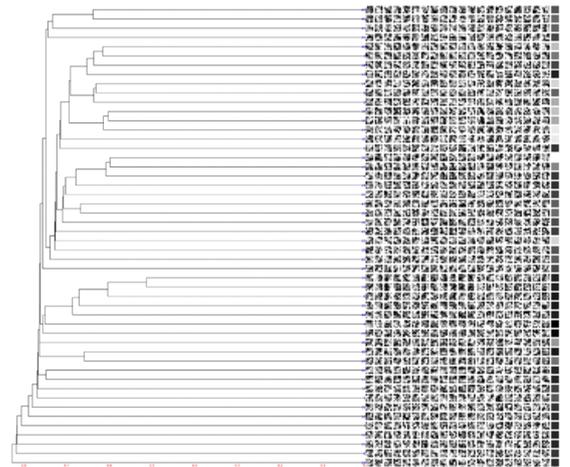


图 7 卷积、反卷积参数结合层次聚类树的可视化
 Fig. 7 Visualization of convolution and deconvolution parameters combined with visualization of hierarchical clustering trees

对卷积核沿 $channel_{out}$ 维度统计数值分布, 运用核密度估计绘制概率密度分布图, 得到图 8 所示结果.

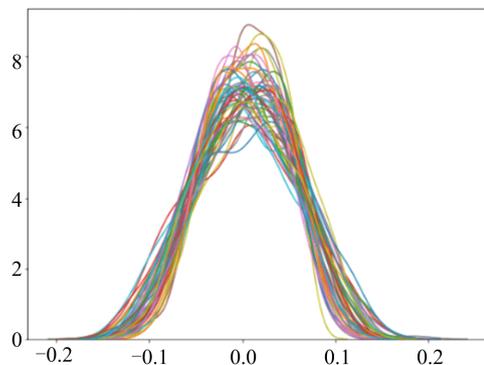


图 8 卷积、反卷积参数沿 $channel_{out}$ 维度分布
 Fig. 8 The distribution of convolution and deconvolution parameters along the $channel_{out}$ dimension

1.2.2 全连接层参数可视化

全连接层参数数量巨大,将权重 $weight$ 直接转化成图像,横向为 $feature_{in}$,纵向为 $feature_{out}$,并且将 $bias$ 拼接至行尾,用热力图的方式进行渲染,得到图 9 所示结果.

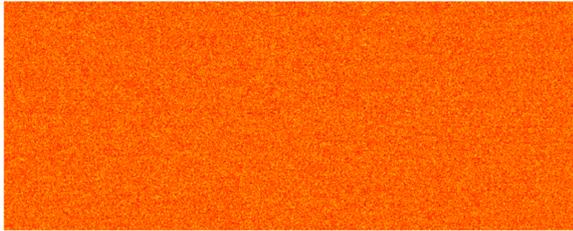


图 9 全连接层参数可视化

Fig. 9 Visualization of fully connected layer parameters

沿 $feature_{out}$ 分析参数的相似程度,绘制层次聚类树,得到图 10 所示结果.

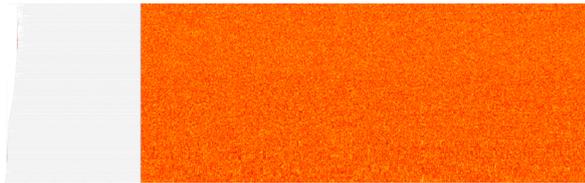


图 10 全连接层参数结合层次聚类树的可视化

Fig. 10 Visualization of fully connected layer parameters combined with visualization of hierarchical clustering trees

沿 $feature_{out}$ 统计数值分布,得到图 11 所示结果.

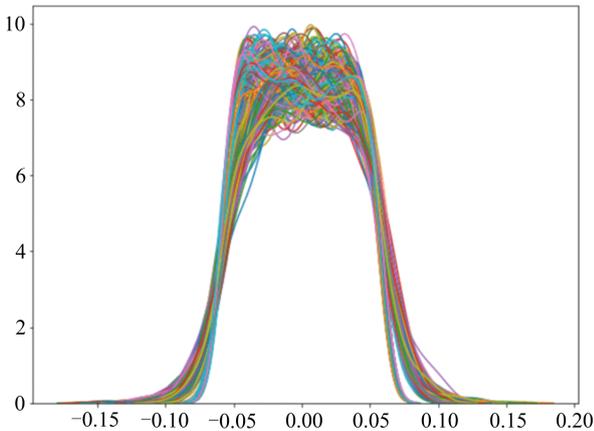


图 11 全连接层参数沿 $feature_{out}$ 维度分布

Fig. 11 The distribution of fully connected layer parameters along the $feature_{out}$ dimension

1.3 可视化分析方法

1.3.1 模型分析顺序

在卷积神经网络模型中,按照网络层与数据输入输出的远近关系,定义网络的抽象层次.越靠近模型输入或输出端,且靠近的输入或输出为图像的网络层抽象层次越低,反之抽象层次越高.即低抽象层的网络层输出与数据集图像的相似程度越高,高抽象层的网络层输出与数据集图像的相似程度越低或者与数据集图像属性(如类别)的相似程度越高.

对模型进行分析时,首先分析低抽象层的中间

层输出,采用结合层次聚类树的可视化方法,分析不同通道特征之间的相似性和差异性,可结合参数分布情况进行分析.其次分析高抽象层的中间层输出,主要分析其参数的分布情况,并用直接可视化辅助分析.再分析卷积层、反卷积层的参数,主要采用结合层次聚类树的可视化方法,分析不同 $channel_{out}$ 的卷积单元的差异性,并用核密度估计辅助分析沿 $channel_{out}$ 的参数分布情况.最后分析全连接参数,在结合层次聚类树的可视化过程中,分析不同 $feature_{out}$ 的差异性以及热力图颜色的均匀程度,在应用核密度估计的分布情况中,分析不同 $feature_{out}$ 的相似程度.

1.3.2 低抽象层输出数据分析方法

对低抽象层的分析重点是结合层次聚类树的可视化分析,良好模型的可视化结果见图 2,其特征明显,没有全为 0 的特征图,并且层次聚类树的分岔点分布均匀,既使得模型包含多种特征,又使得模型能够更精细地提取重要特征.

若模型效果较差,则特征分布图可能出现以下几种情况.①特征图出现大量的黑色块,即全为 0 的特征图;②特征图种类较为单一,体现在层次聚类树的分岔点多在接近 0 的位置;③特征图与数据集图像差别较大,这是因为特征提取时,策略激进,使得特征图相比原图变化剧烈,这种情况将导致特征的丢失.图 12 是劣质模型的可视化,包含大量黑色特征图,并且特征图的变化与原图相比差距较大,有效特征图之间的差异性不足.

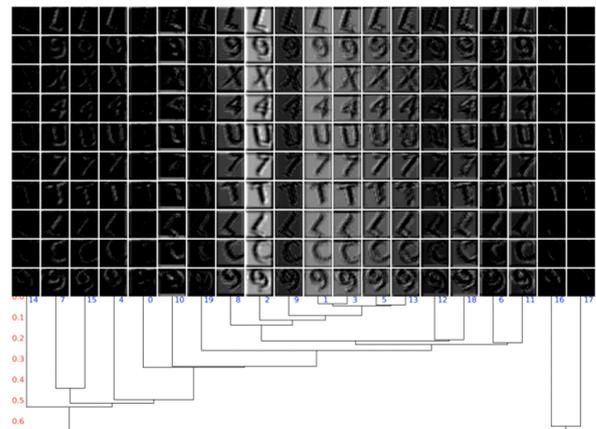


图 12 劣质模型低抽象层四维输出可视化

Fig. 12 Visualization of the four-dimensional output of the low abstraction layer of the inferior model

此外,从数值分布也可判断低抽象层的效果.图 2 为良好模型的低抽象层输出分布,良好模型的不同 $channel$ 的数值分布差异性较大;劣质模型的可视化,如图 13 所示,表明可能存在差异性较小、数值多集中在 0 附近等现象.

若可视化结果出现特征单一的情况,则需要修改网络层提取特征的方法,比如添加批归一化层;若可视化结果出现大量黑色特征图,则需要检查是否是输入数据存在问题,检查本层及本层之前网络层,可能需要修改激活函数,调整网络结构;若可视化结果特征图与输入差别大,则需要扩大网络容量.

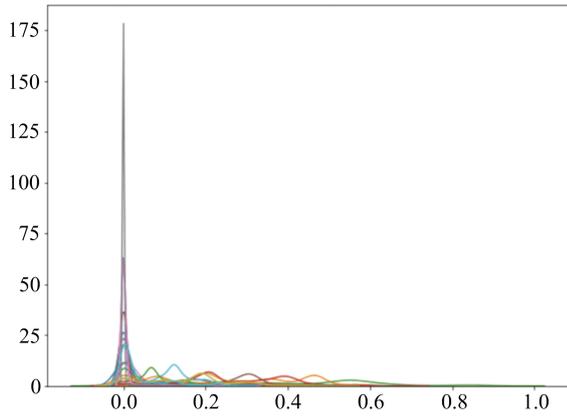


图 13 劣质模型低抽象层输出分布

Fig. 13 Distribution of the four-dimensional output of the low abstraction layer of the inferior model

1.3.3 高抽象层输出数据分析方法

高抽象层的直接可视化往往效果不好,因此更加关注其数值的分布情况. 正常情况的抽象层见图 5,数值分布在不同 feature 上差异较大,并且不会在 0 附近集中很多数据,而劣质模型的数值分布可能会大量的集中在 0 附近,如图 14 所示.

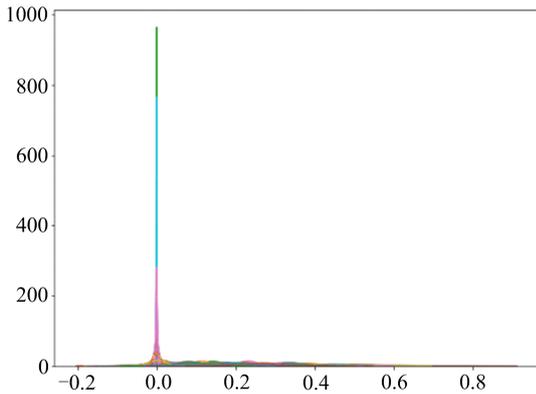


图 14 劣质模型高抽象层输出分布

Fig. 14 Distribution of the four-dimensional output of the high abstraction layer of the inferior model

高抽象层可视化结果出现大量集中在 0 附近时,需要检查是否是激活函数导致出现大量死节点,通常检查本层及本层之前的网络层.

1.3.4 结合层次聚类树的可视化参数分析方法

在训练过程中,良好模型对特征的提取范围覆盖面大并且有倾向性,结合层次聚类树的可视化结果见图 7,层次聚类树的分叉点数值较大,同时有部分卷积单元相似性相对较高. 在劣质模型中,如图 15 所示. 层次聚类树的分叉点数值较大且集中,意味着模型无法提取有效特征,卷积单元在优化算法作用下差异性越来越大.

同样地,优质模型的全连接层参数见图 10,而劣质模型的全连接层参数如图 16 所示. 在层次聚类树的分叉点上,优质模型和劣质模型尽管分叉点数值都较大,但是优质模型的分叉点分布要比劣质模型广. 同时,优质模型的热力图颜色均匀,噪声感弱,劣质模型的热力图颜色较为杂乱,噪声感强烈.

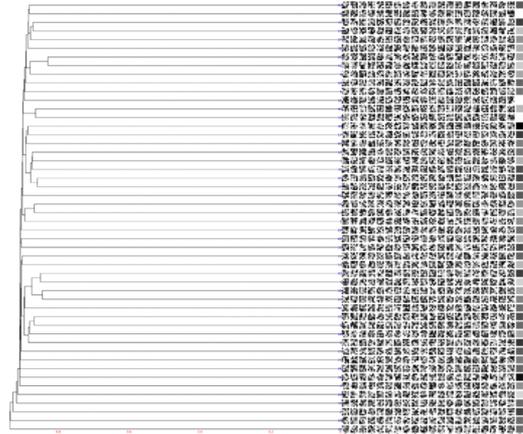


图 15 劣质模型卷积核参数可视化

Fig. 15 Visualization of the convolutional kernel of the inferior model

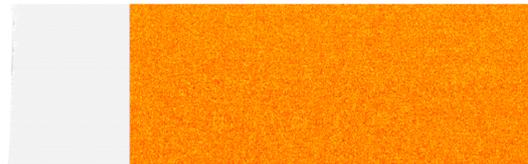


图 16 劣质模型全连接层参数可视化

Fig. 16 Visualization of the dense kernel of the inferior model

1.3.5 基于核密度估计的参数分析方法

在参数分布上,优质模型与劣质模型区别不明显. 如图 8 和 11 所示,优质模型的参数分布多为正太分布或者均匀分布,劣质模型的分布多为均匀分布,并且分布区间相对较大,如图 17 所示. 由此可见,对参数的分析只能作为辅助判断方法.

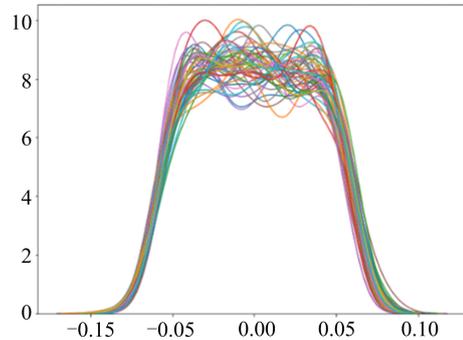


图 17 劣质模型参数分布可视化

Fig. 17 Distribution of parameters of the inferior model

在网络模型中,参数的分布区间呈两端大、中间小,并且向输出端偏移的趋势. 例如,网络一共 7 层,则 1、2、3 层的参数分布接近正太分布,并且分布区间较大,4、5、6 层分布接近均匀,分布区间较小,第 7 层参数分布比第 4、5、6 层略大,呈现均匀分布. 即越靠近输入,参数分布越接近正太分布;越靠近网络两端,参数分布区间越大.

2 实验及结果分析

2.1 实验内容

为了便于使用可视化优化模型的方法,本文选择模型较为复杂的深度卷积生成对抗网络^[19]进行

实验.生成对抗网络模型包含生成器和判别器,生成器的高抽象层位于网络输入端,低抽象层位于网络输出端,判别器与之相反,高抽象层位于网络输出端,低抽象层位于网络输入端.生成对象为 CelebA 数据集中的对其采样后的人脸数据.

CelebA 数据集中包含 202599 张人脸数据,使用其 img_align_celeba.zip 中对齐并剪裁成统一尺寸的数据,数据为 218×178 的 RGB 三通道 256 色真彩图.进行预处理缩放并剪裁成 160×160 的数据,并将数据范围从 $0 \sim 255$ 缩放至 $-1 \sim 1$,数据类型转化为浮点类型.

最初情况下,生成器模型和判别器模型见表 1 和表 2,判别器输入为 160×160 的 3 通道图像,生成器输入为长 256 的随机标准高斯分布的一维向量,损失函数为二元交叉熵,优化器为 RMSprop,学习率为判别器 0.0003,生成器 0.0007, batch size 为 64.

表 1 生成器模型 v1
Tab.1 Generator v1

网络层	激活函数
1 全连接到 2560	ReLU
2 全连接到 25600	ReLU
3 重排列成 $20 \times 20 \times 64$ 的三维张量	
4 5×5 反卷积,步长 2,个数 32	ReLU
5 5×5 反卷积,步长 2,个数 16	ReLU
6 5×5 反卷积,步长 2,个数 3	ReLU

表 2 判别器模型 v1
Tab.2 Discriminator v1

网络层	激活函数
1 5×5 卷积,步长 2,个数 16	LeakyReLU
2 5×5 卷积,步长 2,个数 32	LeakyReLU
3 5×5 卷积,步长 2,个数 64	LeakyReLU
4 重排列成长 25600 的一维向量	
5 全连接到 160	LeakyReLU
6 全连接到 1	sigmoid

2.2 方法验证

首先进行 15000 次训练,生成图像如图 18 所示.此时生成对抗网络是能够工作的,但是效果极差,只能看到人脸的轮廓,损失函数数值波动情况正常.



图 18 模型 v1 生成图片

Fig.18 Generated image of model v1

对模型进行可视化分析.首先可视化生成器的中间层输出,如图 19 所示(依次为生成器 3、4、5 层输出).图 19 表明中间层无法正常生成信息,并且大部分数值为 0.其次分析前向两层全连接分布,如图 20 所示(生成器第 1 层输出整体分布见左图,生成器第 2 层输出整体分布见右图),由图 20 可知,第

1、2 层存在数量众多的数值为 0 的神经元.这种结果将导致后续网络层无法进一步提取这两层输出的特征.

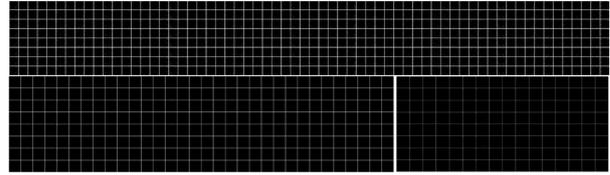


图 19 生成器输出可视化

Fig.19 Visualization of generator output

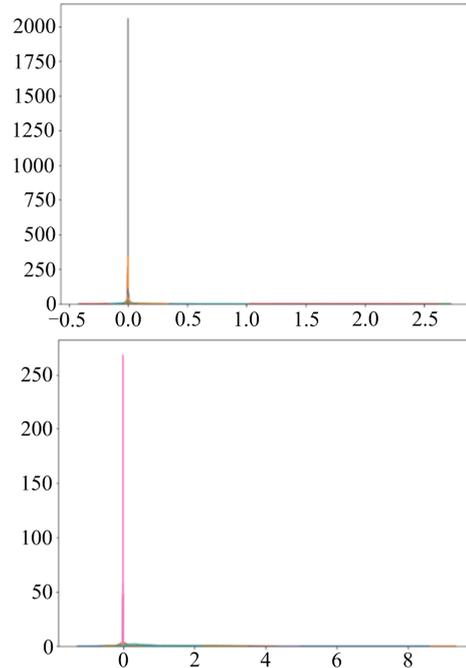


图 20 生成器输出分布

Fig.20 Distribution of generator output

全连接层激活函数为 ReLU,这会产生大量数值为 0 的输出,因此删除一层全连接,得到 v2 版本生成器.

表 3 生成器模型 v2
Tab.3 Generator v2

网络层	激活函数
1 全连接到 25600	ReLU
2 重排列成 $20 \times 20 \times 64$ 的 3 维张量	
3 5×5 反卷积,步长 2,个数 32	ReLU
4 5×5 反卷积,步长 2,个数 16	ReLU
5 5×5 反卷积,步长 2,个数 3	ReLU

在其他参数相同的情况下,经过训练,生成器生成图 21 所示结果.v2 版生成对抗网络模型正常工作,但生成器的生成效果不佳.相比 v1 版本,检查在对生成器进行修改后,生成器的中间输出结果和全连接的输出分布是否有改善.由图 22 可知,修改后的第 1 层全连接层输出数据分布情况正常.由图 23 可知,可视化生成器第 2、3、4 层输出(依次输出),修改后的生成器可正常提取特征.

生成对抗网络的判别器起到区分真实图片和伪造图片的作用,生成器的生成效果下限由判别器



图 21 模型 v2 生成图片

Fig. 21 Generated image of model v2

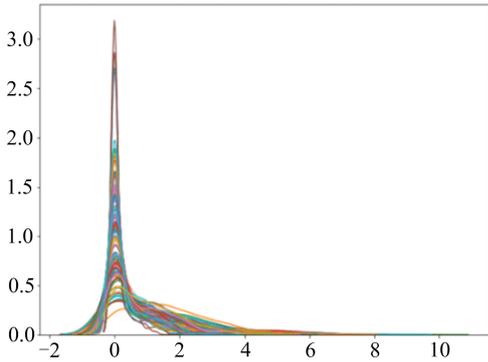


图 22 生成器第 1 层输出分布

Fig. 22 Distribution of generator layer 1 output

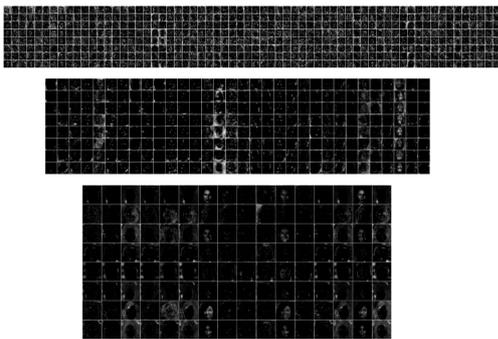


图 23 生成器输出可视化

Fig. 23 Visualization of generator output

的判别能力决定. 如果判别器能够区分非常优质的伪造图片和真实图片, 则生成器生成的图片质量较好; 如果判别器仅能区分非常劣质的伪造图片和真实图片, 则生成器在能够生成劣质的伪造图片后, 不再有优化空间, 无法继续优化以生成更加优质的图像. 从结果中看, 生成的图片效果不佳, 因此推断, 判别器可能存在问题, 接下来可视化分析判别器模型. 可视化结果如图 24 所示, 上中下分别对应判别器 1、2、3 层输出; 判别器中间层参数分布如图 25 所示, 从上到下、从左到右依次为判别器 1、2、3、5 层参数分布.

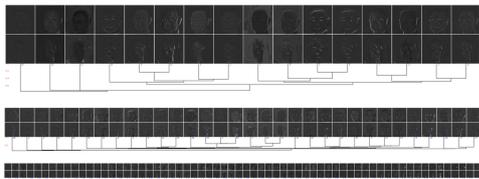


图 24 判别器中间层输出

Fig. 24 Visualization of discriminator hidden layer output

由图 24 可看出, 判别器前三层网络层侧重提取纹理细节, 对其他特征的提取不到位, 因此生成器

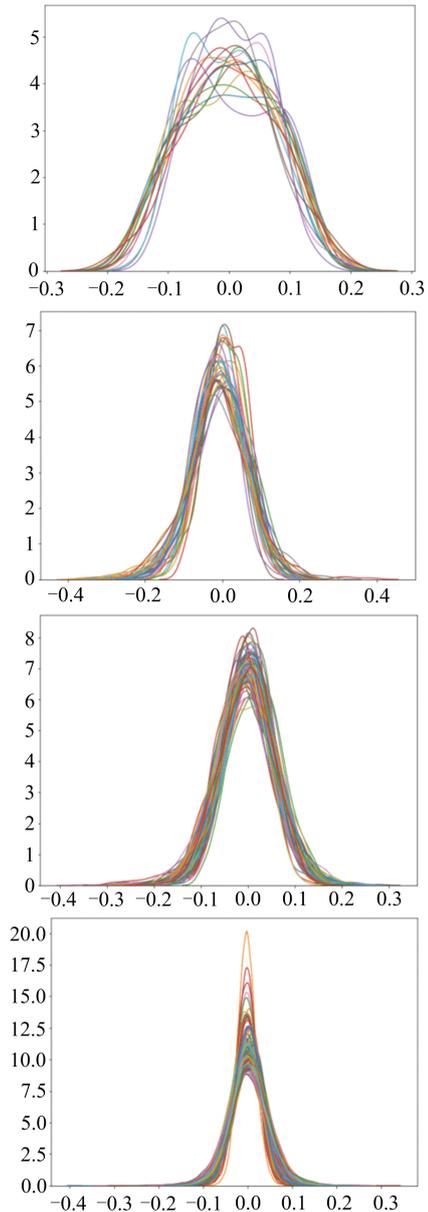


图 25 判别器网络层参数分布

Fig. 25 Distribution of discriminator layer parameters

的图像细节较好, 整体较差. 由图 25 可知, 越接近高抽象层, 不同输出神经元或输出通道的参数分布越相似, 因此推测, 判别器对于高抽象特征的提取能力存在不足. 为了平衡模型对不同类型特征的提取能力, 对判别器增加批归一化层, v3 版判别器见表 4.

保持优化器、学习率、batch size 等参数不变, 经过相同次数的训练后, 生成器的生成结果如图 26 所示. 由图 26 可知, 此时的模型已经比前两个版本有了显著改善.

对比 v2 版模型, 需要可视化判别器前 3 层输出, 如图 27 所示, 其中上中下分别对应判别器 1、2、3 层输出. 由图 27 可看出, 判别器对特征的提取有了明显改进, 特征提取的效果明显提升、多样性增加.

表 4 判别器模型 v3
Tab. 4 Generator v3

	网络层	归一化层	激活函数
1	5×5 卷积, 步长 2, 个数 16	批归一化	Leaky ReLU
2	5×5 卷积, 步长 2, 个数 32	批归一化	Leaky ReLU
3	5×5 卷积, 步长 2, 个数 64	批归一化	Leaky ReLU
4	重排列成长 25600 的一维向量		
5	全连接到 160	批归一化	Leaky ReLU
6	全连接到 1		sigmoid



图 26 模型 v3 生成图片

Fig. 26 Generated image of model v3

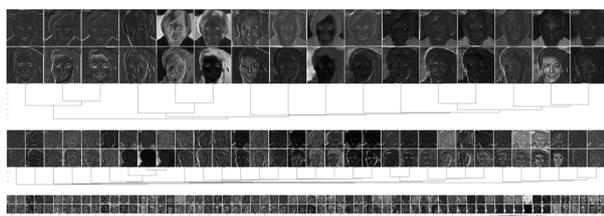


图 27 判别器中间层输出

Fig. 27 Visualization of discriminator hidden layer output

2.3 实验结果及分析

实验中, 3 个模型的迭代过程展示了可视化在优化模型时起到的指导性作用, 表明通过本文设计的可视化方法可快速定位故障的位置。

首先从 v1 到 v2 版本的修正, 通过可视化生成器第 3、4、5 层输出, 快速定位到可能产生故障的位置, 即生成器第 1、2 层, 又通过分析第 1、2 层输出分布, 判断出两层 ReLU 激活的全连接层将导致大量神经元输出为 0, 修改方法可选择修改激活函数或者删除一层网络; 经过尝试, 后者优化效果更明显, 并且可减少参数量。

在 v2 到 v3 版本的改进中, 结合生成对抗网络的特性, 即判别器决定了生成器生成图片质量的下限。通过可视化第 1、2、3 层输出, 分析不同通道特征的相似程度, 得出特征单一化的结论, 并可视化第 1、2、3、5 层的参数分布, 判定模型对高抽象层特征的提取能力不足, 因此加入了批归一化操作进行优化。

最终生成的图像基本满足要求, 若想得到效果更好的生成模型, 则需要对各种超参数进行进一步的微调, 调整过程中仍可通过可视化去判断调整方向的可行性。

综上所述, 本文设计的可视化方法以及分析思路, 有助于定位模型故障点, 从而改进模型。熟练掌握可视化分析方法可大大节约模型调优的时间精力, 有利于模型的开发与应用。

本文使用 TensorFlow 2.1 版本的框架进行开

发, 所采用的可视化方法具有良好的可移植性。经过查阅资料, 在主流常用框架下, 如 PyTorch、Keras、theano, 均可移植成功。

3 结论

深度学习模型开发本身是一件枯燥且耗费精力的事情, 如果没有足够的经验和技巧, 在调参调优模型的过程中必然会耗费大量的时间和精力, 易陷入玄学调参的境地。即使是复现其他论文模型, 由于种种原因, 都可能导致复现失败。

本文则设计了结合层次聚类、核密度估计的可视化方法, 并且提出了根据可视化进行调优卷积神经网络模型的方法, 从而识别模型故障点, 进行快速调优。后续的研究工作将针对两个方面展开, 一方面继续挖掘模型内部数据代表的信息, 并拓展到其他领域的神经网络模型; 另一方面结合训练过程, 动态地分析模型内部数据的时序变化。

参考文献 (References)

- [1] GOODFELLOW I, BENGIO Y, COURVILLE A. Deep Learning[M]. MIT press, 2016.
- [2] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing Systems. 2012: 1097-1105.
- [3] HE T, ZHANG Z, ZHANG H, et al. Bag of tricks for image classification with convolutional neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 558-567.
- [4] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing Systems. 2012: 1097-1105.
- [5] ZEILER M D, FERGUS R. Visualizing and understanding convolutional networks [C]// Computer Vision - ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8689. Springer, Cham, 2014: 818-833.
- [6] YOSINSKI J, CLUNE J, NGUYEN A, et al. Understanding neural networks through deep visualization [EB/OL]. (2015-06-22) [2020-04-24]. <https://arxiv.org/abs/1506.06579>.
- [7] LIU M, SHI J, LI Z, et al. Towards better analysis of deep convolutional neural networks [J]. IEEE

- Transactions on Visualization and Computer Graphics, 2016, 23(1): 91-100.
- [8] DUMOULIN V, VISIN F. A guide to convolution arithmetic for deep learning [EB/OL]. (2018-01-11) [2020-04-24]. <https://arxiv.org/abs/1603.07285>.
- [9] SIBI P, JONES S A, SIDDARTH P. Analysis of different activation functions using back propagation neural networks [J]. Journal of Theoretical and Applied Information Technology, 2013, 47 (3): 1264-1268.
- [10] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift [C]// Proceedings of the 32nd International Conference on International Conference on Machine Learning , 2015: 448-456; doi: 10.5555/3045118.3045167.
- [11] YU D, WANG H, CHEN P, et al. Mixed pooling for convolutional neural networks [C]//International conference on rough sets and knowledge technology. Springer, Cham, 2014: 364-375.
- [12] DAHL G E, SAINATH T N, HINTON G E. Improving deep neural networks for LVCSR using rectified linear units and dropout [C]//2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013: 8609-8613.
- [13] SPENCE R. Information Visualization [M]. New York; Addison-Wesley, 2001.
- [14] ABADI M, BARHAM P, CHEN J, et al. Tensorflow: A system for large-scale machine learning [C]//12th USENIX Symposium on Operating Systems Design and Implementation (OSDI' 16). 2016: 265-283.
- [15] LIU Z, LUO P, WANG X, et al. Large-scale celebfaces attributes (celeba) dataset [J]. Retrieved August, 2018, 15: 2018.
- [16] JOHNSON S C. Hierarchical clustering schemes[J]. Psychometrika, 1967, 32(3): 241-254.
- [17] BOTEV Z I, GROTOWSKI J F, KROESE D P. Kernel density estimation via diffusion[J]. The Annals of Statistics, 2010, 38(5): 2916-2957.
- [18] NGUYEN H V, BAI L. Cosine similarity metric learning for face verification[C]//Asian conference on computer vision. Springer, Berlin, Heidelberg, 2010: 709-720.
- [19] RADFORD A, METZ L, CHINTALA S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. 2015 arXiv:1511.06434.