学校代码：10284
分 类 号：TP183
密　　级：公开
U D C：004.8
学　　号：MG21330071

南京大学

硕 士 学 位 论 文

论 文 题 目　基于前向传播的神经网络可解释性算法研究

作 者 姓 名　张凌茗

专 业 名 称　计算机科学与技术

研 究 方 向　神经网络可解释性

导 师 姓 名　申富饶 教授

2024 年 5 月 27 日

答辩委员会主席 戴新宇 教授

评　　阅　　人 武港山 教授

徐明华 教授

论文答辩日期　2024 年 5 月 16 日

研究生签名:

导师签名:

Interpretable Methods for Neural Networks based on Forward Propagation

by

Zhang Lingming

Supervised by

Professor Shen Furao

A dissertation submitted to

the graduate school of Nanjing University

in partial fulfilment of the requirements for the degree of

MASTER

in

Computer Science and Technology

Department of Computer Science and Technology

Nanjing University

May 27, 2024

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目：基于前向传播的神经网络可解释性算法研究

计算机科学与技术 专业 2021 级硕士生姓名：张凌茗

指导教师（姓名、职称）：申富饶 教授

摘　要

随着深度学习的迅速发展和大数据技术的普及，神经网络在生产和生活中扮演着日益重要的角色。与此同时，对神经网络可靠性和透明度的需求也日益增加，因此神经网络可解释性成为了一个备受关注的领域。神经网络可解释性工作通过深入解析神经网络的内部结构，探索其中的运行规律，以此向用户展示神经网络的决策依据。这项工作一方面有助于提高神经网络的可信度和透明度，使其更好地应用于实际生产和生活中；另一方面，它还可以帮助研究人员优化神经网络，设计出性能更好的模型。

本文主要针对神经网络可解释性中特征可视化与特征归因领域进行描述与回顾，并基于神经网络前向传播法则，设计一种前向相关性分数传播方法。该方法能够有效地实现特征可视化和特征归因，并且具有较高的灵活性，可扩展到各种类型的神经网络中。本文的主要工作有：

1. 本文提出了一种基于前向传播的前向相关性分数传播（Forward Relevance Propagation，FRP）方法。该方法通过制定相关性分数传播规则，为神经网络中的所有神经元分配了与输入尺寸相匹配的解释结果，以实现特征可视化。同时，在通道级别上进行解释结果的融合，可以得到具有良好语义信息的特征解释。
2. 本文将前向相关性分数传播方法转换为特征归因方法，并推广到卷积神经网络以及循环神经网络中以生成显著性图。该方法可以为输入样本生成目标类别的显著性图，突出显示与目标类别分数高度相关的区域，并具有目标类别定位功能。
3. 本文设计并实现了一套神经网络可解释系统。该系统允许用户自主上传网

络模型，并实时在线进行解释功能，为用户进一步了解自己的模型提供了便利。系统采用基于前向传播的前向相关性分数传播方法和特征归因方法对神经网络及样本进行解释和分析。

实验结果表明，本文提出的方法能够生成具有有效语义信息的输出结果，并且有效地解释和分析神经网络的决策过程。同时，该算法在神经网络可解释性系统中的应用为用户理解神经网络模型提供了便利，同时也验证了算法的有效性和实用性。

关键词：神经网络；可解释性；可视化；机器学习

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Interpretable Methods for Neural Networks based on Forward Propagation

SPECIALIZATION: Computer Science and Technology

POSTGRADUATE: Zhang Lingming

MENTOR: Professor Shen Furao

ABSTRACT

With the rapid development of deep learning and the widespread adoption of big data technology, neural networks are playing an increasingly important role in production and daily life. Meanwhile, there is a growing demand for the reliability and transparency of neural networks, making neural network interpretability a highly sought-after field. Neural network interpretability endeavors to delve into the internal structure of neural networks, exploring their operational principles to elucidate the decision-making process to users. This effort not only enhances the credibility and transparency of neural networks for better application in practical production and daily life but also assists researchers in optimizing neural networks and designing models with improved performance.

The main focus of this paper is to describe and review the fields of feature visualization and feature attribution in neural network interpretability. Based on the principles of neural network forward propagation, a forward relevance score propagation method is designed. This method effectively achieves feature visualization and feature attribution, with high flexibility that can be extended to various types of neural networks. The main contributions of this paper are as follows:

1. We propose a forward relevance score propagation method based on forward propagation (Forward Relevance Propagation, FRP). This method formulates rules for relevance score propagation, assigning explanation results matching the input size to all neurons in the neural network to achieve feature visualization. Simultane-

ously, by integrating the interpretation results at the channel level, meaningful feature explanations with good semantic information can be obtained.

2. We transform the forward relevance score propagation method into a feature attribution technique and extends its application to convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to generate saliency maps. This method is capable of producing saliency maps for input samples with respect to the target class, thereby highlighting regions highly correlated with the target class score and possessing target class localization capabilities.
3. We design and implement a neural network interpretability system. The system allows users to upload network models independently and perform real-time online interpretation, providing users with convenience in further understanding their models. The system utilizes forward relevance score propagation based on forward propagation and feature attribution methods to interpret and analyze neural networks and samples.

The experimental results demonstrate that the proposed method can generate output results with effective semantic information and effectively interpret and analyze the decision-making process of neural networks. Additionally, the application of this algorithm in the neural network interpretability system provides users with convenience in understanding neural network models, thus validating the effectiveness and practicality of the algorithm.

KEYWORDS: Neural Networks; Interpretability; Visualization; Machine Learning

目　　录

插图目录

表格目录

第一章　绪论

本章节主要阐述神经网络可解释性的研究背景以及意义，介绍了目前神经网络在可解释性方面的相关进展与工作，并对本文的研究内容与创新点进行说明，最后对文章结构进行了总结。

1.1　研究背景及意义

随着数字化时代的来临，人工智能（Artificial Intelligence，AI）在海量数据与丰富算力的加持下迅速发展，并且正在改变当下的生活和工作方式。归因于图形处理器（Graphics Processing Unit，GPU）的广泛应用和专用硬件的开发，机器学习（Machine Learning）算法尤其是深度学习（Deep Learning）逐渐成为各项人工智能任务中的主流解决方案。从感知机模型[1]（Perceptron）和反向传播学习[2]（Back Propagation）理论的提出，到模型预训练与微调范式[3]（Pretrain & Fine Tune）的流行；从卷积神经网络（Convolutional Neural Network，CNN）在图像识别竞赛 ILSVRC 上超越传统方法[4]，到当下大语言模型[5-8]（Large Language Model，LLM）统一自然语言处理（Natural Language Processing，NLP）各项下游任务，神经网络（Neural Network）作为各种算法的底层模型，推动深度学习技术在计算机视觉（Computer Vision，CV）、自然语言处理和图学习等领域蓬勃发展，并对现实生活生产产生极大影响。从汽车智能驾驶技术[9-10]到智能医疗诊断[11-12]，从智能语音助手[13-14]到金融风险管理[15-16]，深度学习逐渐成为人们使用并依赖的工具，用以实现各种任务与决策，因此引发的安全问题与隐私问题同样值得关注。

相较于支持向量机[17]（Support Vector Machine，SVM）、决策树[18-19]（Decision Tree）等具有较好可解释性的传统机器学习模型，深度学习和神经网络因其内部复杂的非线性、非单调以及非多项式函数操作，经常被批评其“黑盒”特性和缺

乏可解释性。神经网络的可解释性问题已成为科研界和工业界高度关注的研究议题。一方面，神经网络的信任度和可靠性在涉及安全与健康的关键领域显得尤为重要，比如智能驾驶和医疗诊断等领域，用户对模型决策结果的信任是不可或缺的；另一方面，从法律和伦理角度，人类对模型算法的透明度和可解释性提出了要求。例如，2018 年欧洲议会通过的通用数据保护条例（General Data Protection Regulation，GDPR）中，就特别引入了关于自动化决策的条款，明确规定数据主体有权获得涉及自动化决策的相关解释信息，这突显了提高模型可解释性的迫切需求。

神经网络的可解释性研究致力于以人类可理解的方式解释神经网络模型。美国国家标准与技术研究院（National Institute of Standards and Technology, NIST）于 2020 年 8 月发布关于可解释人工智能的四项原则[20]：可证明性（解释结果可以被证据证明）、可用性（解释结果能够被用户理解并对用户有意义）、准确性（解释结果必须准确反应模型运行机制）和限制性（解释结果能识别出不适合其自身运行的情况）。可解释性研究源起进一步了解神经网络模型的意愿，致力于揭开神经网络内部工作原理与训练特性，帮助人类解决系统可靠性问题以及消除偏见和幻觉，并通过优化神经网络结构与训练过程从而提升模型性能。

1.2 研究现状

神经网络作为机器学习领域中的一种重要模型，其可解释性也源于机器学习可解释性研究。目前，主流学术还未对可解释性概念有统一的数学定义。Miller[21]曾给出可解释性的非数学定义：可解释性是人类能够理解决策原因的程度。即当人类越容易理解模型做出的决策或者预测，那该模型的可解释性则越高。在模型的研究阶段，模型在任务评估上的表现以及性能更受关注；而当模型接触现实世界，会带来更多的问题：模型决策或预测的原因是何、决策或预测过程是否有足够的现实依据、决策或预测过程是否存在道德问题与偏见等。具有优秀可解释性的模型则在上述问题中能够给出合理合法合规的解答。

在神经网络对数据进行处理时，输入数据与神经网络权重进行乘法运算和非线性变换，这一过程可能涉及上百万次数学运算。由于这么多的乘法和非线性运算，导致输入与输出之间的映射难以被完整表述。传统的机器学习可解释

方法仍然可行，但更多的工作将着眼于设计更符合神经网络架构的可解释方法，以更好地理解神经网络内部结构的设计。

在神经网络可解释性研究中，仍然可以使用机器学习可解释性领域中的模型未知方法（Model-Agnostic Methods）来解释神经网络，该方法也被称为黑盒解释方法。该类方法的巨大优势在于其灵活性，适用于所有的机器学习模型，包括神经网络。在考虑多种类型模型的可解释性比较时，该方法更容易实现统一的评估标准。甚至在考虑模型安全性的情况下，模型未知方法仍然可以对不可见的模型进行解释与评估。模型未知方法分为两类：全局模型未知方法（Global Model-Agnostic Methods）和局部模型未知方法（Local Model-Agnostic Methods）。关注模型对整体输入解释与否的是模型未知方法分类的标准。全局模型未知方法关注整个模型的平均表现，通常将模型解释为基于数据分布的期望值预测。局部模型未知方法则更加关注单个实例的预测解释，尝试建立单个输入与输出之间的因果联系。

常见的全局模型未知方法有：部分依赖图（The Partial Dependence Plot，PDP）、特征交互作用（Feature Interaction）、函数分解（Functional Decomposition）和全局代理模型（Global Surrogate Models）等。部分依赖图[22-23]揭示了部分特征对模型预测结果的边际影响，它可以解释模型预测结果和被解释特征之间是线性、单调还是更为复杂的关系。特征交互[24-25]作用考虑输入特征在模型决策过程中可能发生的相互作用，这种作用导致整体模型的预测输出不能直接等价于单个特征预测输出的总和。函数分解方法[26-27]希望用简单函数的线性组合代替复杂非线性函数。全局代理模型[28]希望通过训练与黑盒模型相近的可解释性模型作为代理模型，以此对黑盒模型进行解释。

常见的局部模型未知方法包括：个体条件期望（Individual Conditional Expectation，ICE）、反事实解释（Counterfactual Explanation）、局部代理模型（Local Interpretable Model-agnostic Explanations，LIME）和夏普利值计算（Shapely Values）等。个体条件期望方法[29-30]显示了对于单个示例，当特征发生变化时其预测是如何随之发生变化的。反事实解释[31-32]尝试为个体示例的特征值与预测建立因果关系，并得到简单解释：因为输入特征值而导致预测输出。局部代理模型[33-34]则在局部的数据域中训练与黑盒模型相似的代理模型，以此来解释黑盒模型的决策过程。夏普利值[35-36]则基于博弈论中的 Shapely 价值方法，将实例的

输入特征视为参与“游戏”的“玩家”，以此计算输出值的贡献值如何分配至输入特征。

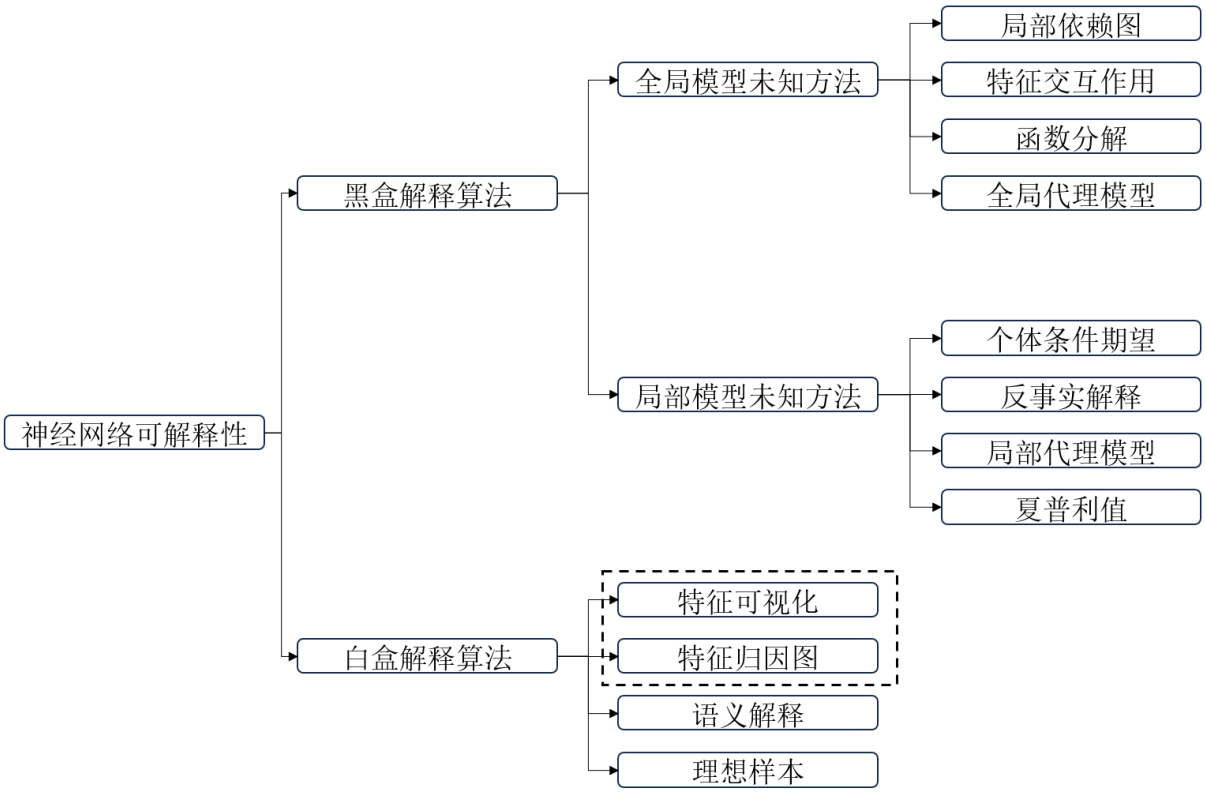


图 1.1 可解释性算法分类

然而，黑盒解释方法始终未能考虑神经网络独特的架构，难以挖掘神经网络内部深层次的规律与规则，同时还存在计算量大、计算困难等缺点。为了更好地理解神经网络内部的运作规律，可解释方法开始融入网络的结构特征，通过揭示神经网络内部隐藏层的特征与概念，达到对神经网络的解释作用。该类方法被称为白盒解释方法，常见的方法有特征可视化（Learned Features Visualization）、特征归因（Features Attribution）、语义解释（Concepts Explanation）和理想样本（Ideal Sample）等。特征可视化[37-38]方法旨在将神经网络内部的网络权重或特征向量进行可视化，从视觉语义上对其进行解释。特征归因方法[39-40]计算输入样本对输出预测的贡献值，也称作显著性图（Saliency Maps）。语义解释[41-43]尝试在概念（Concepts）尺度上对神经网络的内部特征进行解释。理想样本[44]希望找到对于网络激活程度最高的样本，作为网络使用的示例。

传统的机器学习方法在设计之初并未充分考虑神经网络的内部结构，难以对神经网络内部的特征处理作出合理的解释，更多将神经网络当作黑盒模型，对

其整体的功能与决策进行端到端的解释。而结合神经网络结构的解释方法则存在一些缺陷，如计算量大、需要改变网络结构等。此外，大多数解释方法与网络应用的任务密切相关，缺乏统一的可解释方法来适应不同的神经网络架构。

神经网络可解释性算法分类如图1.1所示。在上述神经网络可解释算法中，本文主要聚焦于特征可视化以及特征归因。当前的特征可视化方法更多地关注于将网络内部的权重参数进行表征，解释结果往往是高维特征，难以在语义层面上理解。而当前的特征归因方法则大多依赖于网络结果进行反向传播或梯度计算，因此计算量较大。本篇论文考虑在全连接神经网络、卷积神经网络和长短期记忆网络等异构网络结构中设计通用的前向传播规则，以获得网络内部的特征可视化和特征归因方法。

1.3 本文研究内容

本文在研究特征可视化的基础上，结合特征归因方法，通过设计与网络结构相符合的前向传播规则，计算输入在每个神经元节点的可解释特征，并在通道维度上提取可解释特征的语义信息，对神经网络进行神经元粒度的特征解释。并通过对输出神经元的特征解释，以此获取输入样本关于预测输出的归因图，即贡献度。这种可解释算法同时考虑被解释网络的结构特性与计算量，可在不同的网络中较快得到任意指定神经元级别的可解释特征，并计算对应的显著性图，此外，该算法还可以被集成到实际的神经网络可解释性系统中。本文的主要内容以及创新点如下：

1. 本文提出了一种基于前向传播的特征可视化方法。该算法通过自定义前向传播法则，在神经元粒度上生成与输入样本相同尺寸的特征向量，并在通道粒度上对特征向量进行语义解释。本文设计实验验证了语义解释更好的通道对网络的性能有更高的影响。
2. 本文提出了一种基于前向传播的特征归因方法。该算法可在分类任务上对全连接网络、卷积神经网络以及长短期记忆神经网络的输入样本构建特定输出类别的贡献分数，生成对应的显著性图像。算法基于单次前向传播规则，有效地增加了显著性图像的生成质量。
3. 本文搭建了一个神经网络可解释性系统。该系统使用本文提供的特征可视

化与特征归因方法对用户上传的神经网络以及输入样本进行特征可视化以及解释，方便用户对神经网络内部运行机制与规律有更深的理解。

1.4 论文结构纲要

本文主要研究神经网络特征可视化方法与特征归因方法，提出一种基于前向传播神经元粒度的特征可视化方法，并基于该算法提出一种通用的神经网络特征归因方法，最后基于这两种算法搭建了一个神经网络可解释性系统。

全文一共分为六章，其结构如图1.2所示：第一章内容为绪论，主要介绍神经网络可解释性的研究背景以及研究意义；第二章内容介绍了特征可视化以及特征归因领域中目前主流的方法；第三章内容介绍了一种基于前向传播规则的特征可视化方法，可生成神经元粒度的任何神经元节点的可解释性特征向量；第四章内容介绍了基于前向传播规则的特征归因方法，对不同网络结构的输入输出生成显著性图像；第五章内容介绍了基于本文提出的算法构建的神经网络可解释性系统；第六章内容总结全文，并对未来工作进行了展望。

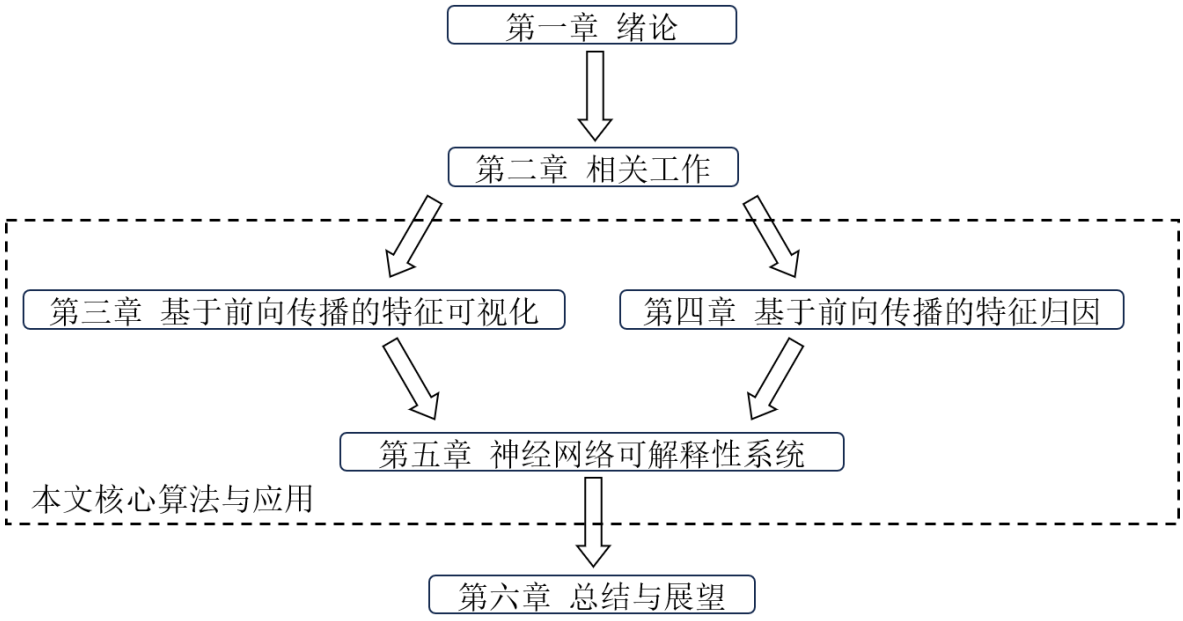


图 1.2 文章结构

第二章　相关工作

特征可视化和特征归因是神经网络可解释性工作中具有代表性的两种解释方法。它们的特点是易于可视化，能够在视觉上增加使用者对神经网络决策的理解程度。本章首先介绍了神经网络架构的基本原理，并对当前主流的特征可视化和特征归因方法进行了介绍和回顾，最后对本文讨论的任务进行了说明。

2.1　神经网络基本原理

2.1.1　神经元结构与神经网络

神经网络的基本结构是神经元，其设计思想源自生物中神经细胞的传播过程。在生物神经元中，神经细胞通过突触与其他神经元相互连接。其细胞体接收来自其他神经元的信号输入，并进行合并加工，然后再通过轴突末端的突触将信号输出传递给其他神经元。在输入信号的合并加工过程中，如果多个输入信号的总和未超过神经元固有的被激活的边界值（称为阈值），那么该神经元的细胞体将忽略接收到的信号，不做任何反应。基于这种输入信号的整合加工和激活输出的模式，研究者可以设计出人工神经元的雏形。

1943 年，心理学家 Warren McCulloch 和数学家 Walter Pitts 合作提出了第一个人工神经元模型，称为 MP 神经元[45]。MP 神经元的设计基于生物神经元功能的理解，模拟了生物神经细胞对其接收到的所有其他神经细胞的刺激输入进行整合处理以及激活输出的过程。因此，MP 神经元可视作一个接受其他神经元输入并根据输入产生输出的简单逻辑门。

MP 神经元的结构如图2.1所示，其数学表达式为：

$$Y = f\left(\sum_{i=1}^{N} x_i w_i + b\right). \tag{2.1}$$

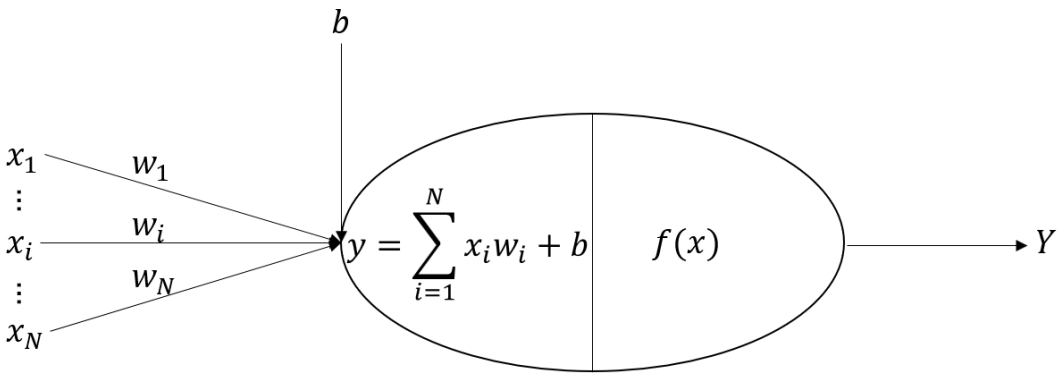


图 2.1 MP 神经元结构

MP 神经元模型由两部分组成，多个输入的整合部分和输出的激发部分。整合部分是一个多输入逻辑门,每个输入的连接有其独特的权重,其中 $x_i\,(1 \leqslant i \leqslant N)$ 为 MP 神经元接受的输入，$w_i\,(1 \leqslant i \leqslant N)$ 为连接的权重值，b 为神经元固定的偏置值。整合部分将所有输入进行加权求和，并加上偏置得到一个总输入，本质上是对所有的输入做线性组合，如公式2.2所示：

$$y = \sum_{i=1}^{N} x_i w_i + b. \tag{2.2}$$

整合部分的输出会作为激发部分的输入。激发部分的主体是激活函数 $f\,(x)$，激活函数决定了神经元是否被激活。MP 神经元模型最初采用的激活函数为 sgn 函数，即公式2.3

$$f\,(x) = \operatorname{sgn}(x) = \begin{cases} -1, & \text{if } x \leqslant 0, \\ 1, & \text{if } x > 0. \end{cases} \tag{2.3}$$

将多个神经元进行规定模式的连接，即可得到多层感知机（Multilayer Perceptron，MLP），即简单的人工神经网络（Artificial Neural Network，ANN）。多层感知机引入了隐藏层，具有以任意精度逼近输入数据和输出数据之间任意非线性关系的能力[46]。多层感知机通过多个非线性变换将输入数据映射到高维空间中，并从中学习非线性关系，拟合更为复杂的分布。

多层感知机由输入层、若干个隐藏层和输出层组成。每一层由多个神经元组成，层间的神经元互不相连，只与前一层和后一层的神经元进行连接。其中输入层同样允许有多个输入，并可以同时输出多个输出。多层感知机的结构如

图2.2所示，其中 $x\,(1 \leqslant i \leqslant N)$ 为 N 个输入，$Y\,(1 \leqslant i \leqslant M)$ 为 M 个输出。

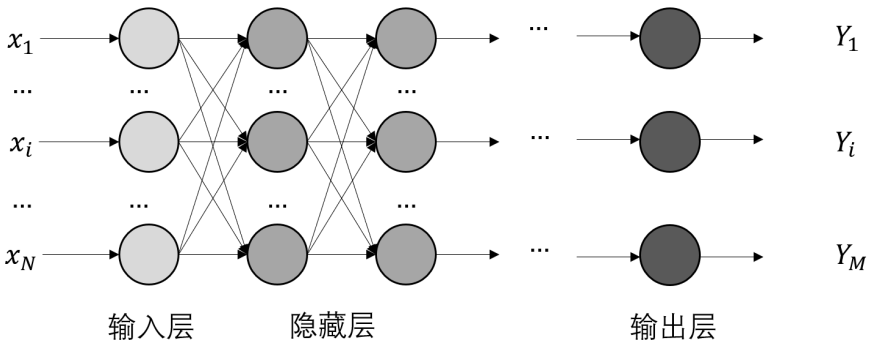


图 2.2 多层感知机模型

多层感知机采用的激活函数更为丰富，除了 sgn 函数之外还有 Sigmoid 函数、Tanh 函数和 ReLU 函数等等。其中 Sigmoid 函数的数学形式为：

$$f\,(x) = \text{Sigmoid}\,(x) = \frac{1}{1+e^{-x}}, \tag{2.4}$$

其函数图像如图2.3所示。

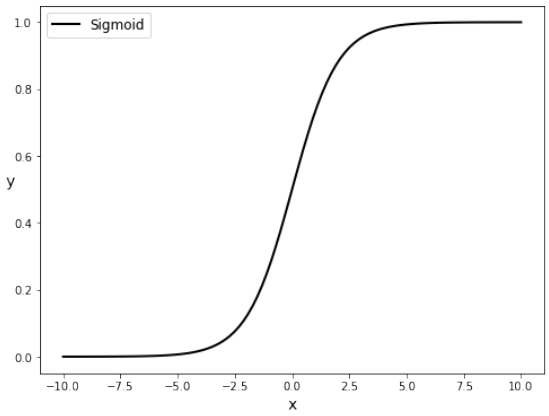


图 2.3 Sigmoid 函数图像

Sigmoid 函数可以接受负无穷到正无穷的输入，并将输出映射至 $(0, 1)$ 之间，这使得它适用于表示概率值。同时，它是一个连续且可微的函数，这使得它能够与基于梯度的优化算法（如反向传播）一起使用，实现神经网络的训练。然而，Sigmoid 函数存在梯度消失的问题[47-48]。在不饱和区，其梯度较小，当多层感知机的层数增加时，梯度容易消失，导致训练失效。此外，Sigmoid 函数还具有收敛缓慢和计算量大等缺点。

ReLU 函数是现代神经网络更偏向选择的激活函数。其数学表达式为：

$$f(x) = \mathrm{ReLU}(x) = \mathrm{Max}(0, x), \tag{2.5}$$

其函数图像如图2.4所示。

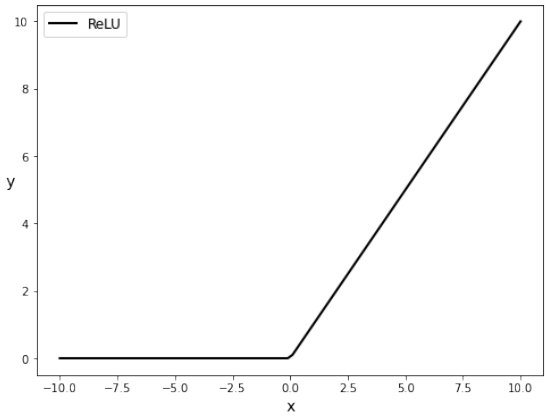


图 2.4 ReLU 函数图像

ReLU 函数的计算非常简单，只需要比较输入与 0 的大小，这使得其更容易受到大参数量神经网络的青睐。同时，ReLU 具有较高的稀疏性，非激活区和非饱和区都相对宽广，在梯度计算上也便于求导，具有极好的计算效率。其固定的导数值也有效地解决了梯度消失问题。然而，ReLU 激活函数会导致神经网络对于输入小于等于 0 的情况表现为不激活，这可能导致某些不合适的初始化，会使得某些神经元节点始终处于输出为 0 的状态，从而无法通过样本学习更新权重[49]。

神经元的激活函数为神经网络提供了强大的表达能力的理论基础。线性变换的组合只能得到线性变换，而激活函数大多为非线性函数。因此，神经元激活函数的非线性性质允许神经网络学习和表示更加复杂的函数关系。这扩展了神经网络的表示能力，使其能够捕捉和学习输入数据中的复杂模型和特征。

同时，激活函数的非线性特性引入了神经网络的不透明性。大量的非线性操作使得建立输入和输出的直接映射变得困难。输入数据在隐藏层中被转换为难以理解的高维特征，从而使得输入到输出的间接转换过程也变得难以理解。为了探究神经网络的决策功能，结合特定的任务和场景将使解释工作更易于理解。

2.1.2 图像分类任务与卷积神经网络

在神经网络应用中，图像分类（Image Classification）是最经典的任务之一。这是计算机视觉领域中的一个重要任务，旨在将输入的图像分到预定义的类别中。在图像分类任务中，模型需要学习从输入图像到输出类别的映射关系，以便对未知图像进行正确分类。

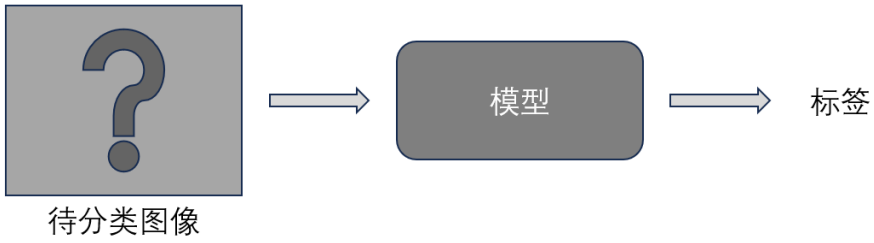


图 2.5 图像分类任务

传统的计算机视觉技术对于图像分类任务的处理一般包括三个步骤：

1. 特征处理：传统的图像分类方法通常会从图像中提取出代表图像特征的数值特征，例如颜色直方图、纹理特征、形状描述符等。这些特征能够描述图像的视觉属性，帮助区分不同的图像类别。
2. 特征选择：在特征提取后，可能会对特征进行选择或降维，以减少特征的维度和计算复杂度，同时保留最具代表性的特征信息。特征选择的方法包括过滤式、包裹式和嵌入式等。
3. 模型训练：对于选择的特征，使用机器学习模型如支持向量机、决策树等算法训练模型，使得模型利用选择的特征来学习不同图像类别的边界，从而实现图像分类任务。

在传统的图像分类方法中，特征处理的方法中，无论是基础的颜色直方图，或者是纹理特征（Texture Features）还是尺度不变特征变换[50]（Scale-Invariant Feature Transform，SIFT）以及方向梯度直方图[51]（Histogram of Oriented Gradients，HOG）等方法都有着较为容易理解的先验知识，提取的特征也是可解释性较强的特征。模型也常使用可解释性较好的机器学习模型，如支持向量机、决策树等。整个对图像分类处理的决策过程都容易被人类所理解并信任。

在 2012 年的 ImageNet 大规模视觉识别挑战赛[52]（ImageNet Large Scale Visual Recognition Challenge，ILSVRC）图像分类任务中神经网络开始崭露头角。该

赛事使用的数据库 ImageNet 是一个包含超过 1 400 万张图像和超过 2 万个类别标签的庞大图像数据库。当年由来自多伦多大学的 Alex Krizhevsky、Ilya Sutskever 和 Geoffrey Hinton 提出的名为 AlexNet 的深度卷积神经网络[4]（Deep Convolutional Neural Network，DCNN）模型以远超传统算法的优势赢得了图像分类任务的冠军。

深度卷积神经网络是在多层感知机上进行两类改动得到的深度学习模型。第一个改动是层数的增加，使神经网络的深度变深，让输入经过的特征变换更多。第二个改动则是新增了名为卷积层（Convolutional Layer）的神经元的连接方式。在多层感知机中，神经元不与同层的神经元进行连接，但会与前一层以及后一层的所有神经元进行连接，故这类连接方式的神经网络也被称作全连接神经网络（Fully Connected Neural Network，FCN）。而卷积神经网络在基于图像知识的平移等变性以及局部相关性两个归纳偏置的基础上，设计了卷积层这一连接结构。

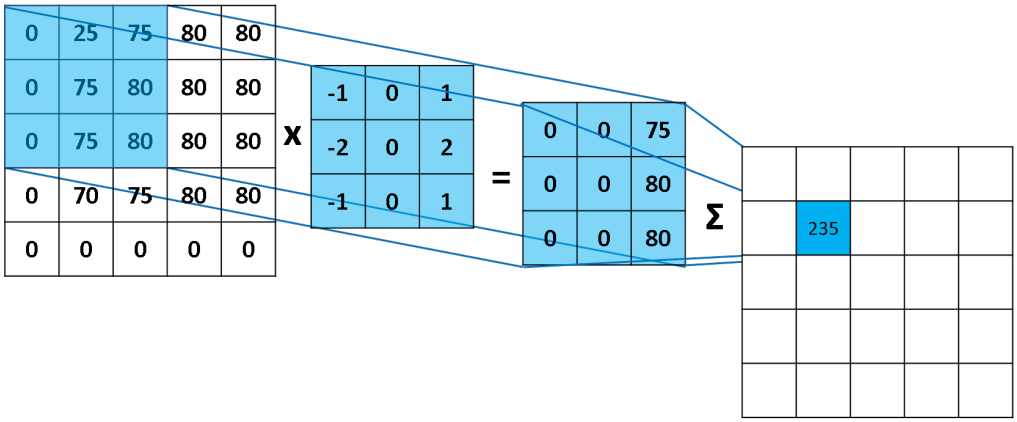


图 2.6 卷积计算过程

根据局部相关性，卷积层中的神经元只与相邻层位置相近的神经元进行连接。并且根据平移等变性原理，同一层的所有连接使用相同的连接权重，以参数共享的方式减少总的参数量。卷积层实质上是进行了参数共享以及局部连接的神经网络层，是全连接层的一种简化形式。卷积层的计算过程如图2.6所示。

在图像分类任务中，卷积神经网络能够实现端到端的标签概率输出，无需手动进行特征提取与选择。从可解释性的角度来看，卷积操作可以被视为使用滤波器来提取特征的过程。随着网络层数的增加，感受野也相应增加，从而提取的特征范围也随之扩大。最终，通过全连接层对特征进行线性组合，输出每个类别

的概率值。然而，图像输入后转换的特征向量维度较高，难以直接理解其含义。每个神经元的作用以及决策依据也不够清晰，这使得对分类结果的可信度也难以提高。

2.1.3 文本分类任务与循环神经网络

与计算机视觉领域相似，文本分类任务是自然语言处理中的一个基础且具有代表性的任务。在自然语言处理领域，许多任务都可以通过文本分类的方式来解决，例如垃圾文本识别、涉黄涉暴文本识别、意图识别、文本匹配和命名实体识别等。

在 Transformer[53]为基础的神经网络成为自然语言处理领域主流之前，文本分类任务的处理方法主要分为两大类：基于规则和基于统计与机器学习。基于规则的方法相对简单直接，通过手动设计的规则集将文本分配到不同的类别或标签中。这种方法通常适用于特定领域或任务，其中领域知识可以转化为一系列规则。基于规则的文本分类方法简单易懂，可解释性强，并且易于调整。然而，它也面临着规则设计的主观性、对领域知识的依赖性以及适用性受限等挑战。

基于统计与机器学习的方法通常首先使用词向量化作为文本数据的预处理工作，例如词袋模型（Bag-of-Words Model）、TF-IDF 方法[54]和词嵌入[55]（Word Embedding）等。然后，通过机器学习模型进行学习与分类：朴素贝叶斯分类器[56]（Naive Bayes Classifier）基于贝叶斯定理对文本特征的条件概率分布进行建模；支持向量机寻找向量化的文本特征超平面进行二元分类；K-最近邻算法[57]（K-Nearest Neighbors，KNN）使用文本之间的相似度来确定最近邻居，并根据类别进行分类。

根据文本序列特征，神经网络将固定的连接方式改为循环连接，形成了循环神经网络[2]（Recurrent Neural Network，RNN）。循环神经网络专门用于处理序列信息，其具有记忆功能，能够考虑序列中前面的信息来影响后续的输出。循环神经网络的基本结构包括一个或多个循环单元，这些单元通过时间步连接起来，每个时间步都接收当前输入和前一个时间步的隐藏状态作为输入，然后产生当前时间步的隐藏状态和输出。这种结构使得循环神经网络能够对任意长度的序列进行处理，同时利用之前的信息来影响当前的输出。

循环神经网络通过利用隐藏状态来保存之前的信息。然而，隐藏状态本身也

是一个高维特征向量，其含义难以直接理解。对于整个序列而言，相同的参数如何作用于不同的输入，并进行信息处理和状态提取，也是一个值得探究的问题。

2.2 神经网络可解释性

本节将在图像分类任务场景下介绍神经网络可解释性相关工作，重点介绍特征可视化与特征归因两种可解释方法。

2.2.1 特征可视化

特征可视化旨在揭示神经网络学习到的特征。卷积神经网络直接使用原始形式（像素）作为输入，而非其他传统机器学习使用提取的具有明显语义信息的特征（如边缘、条纹）。在原始图像经过许多卷积层之后，网络所捕捉的特征越来越复杂。最后将变换后的图像信息经过全连接层转变为分类结果或预测分数。卷积神经网络从原始图像像素中学习抽象特征和概念，通过激活最大化[44]（Activation Maximization，AM）方法可以将其学习的特征可视化。激活最大化的方法即找到最大化目标解释单元激活的输入。此处的单元可以是单个神经元、单个卷积核、整个通道或者整个层。其数学表示形式为：

$$input' = \arg\max_{input} h\left(input\right). \tag{2.6}$$

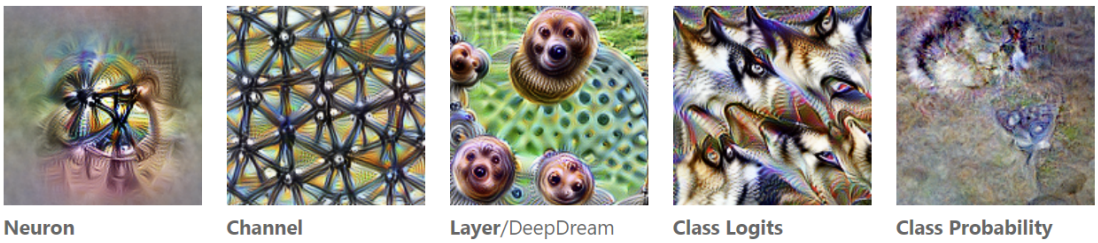


图 2.7 不同类别单元的激活最大化结果[37]

在图像分类任务中，可以使用该方法寻找出能使某类别分数最高的图像，作为该网络对某类别的最佳输入样本。激活最大化通常使用优化的方法来寻找最佳样本。首先随机生成输入，以目标解释单元的激活输出为优化目标，不断调整输入，直至激活输出达到最大值。同样地，也可以通过最小化目标值达成激活最

小化方法寻找目标解释单元的负反应输入样本。

图2.7是各种目标解释单元的激活最大化示例。其中从左到右分别是神经元、通道、隐藏层、分类分数（Softmax 处理前）、分类概率（Softmax 处理后）。

激活最大化能够非常直观地找到每个目标解释单元学习到了何种特征，并且告诉人类神经网络的每一层都在对输入做何种特征提取。图2.8展示了在 ImageNet 上训练的 GoogLeNet[58]各个层次的激活最大化效果。

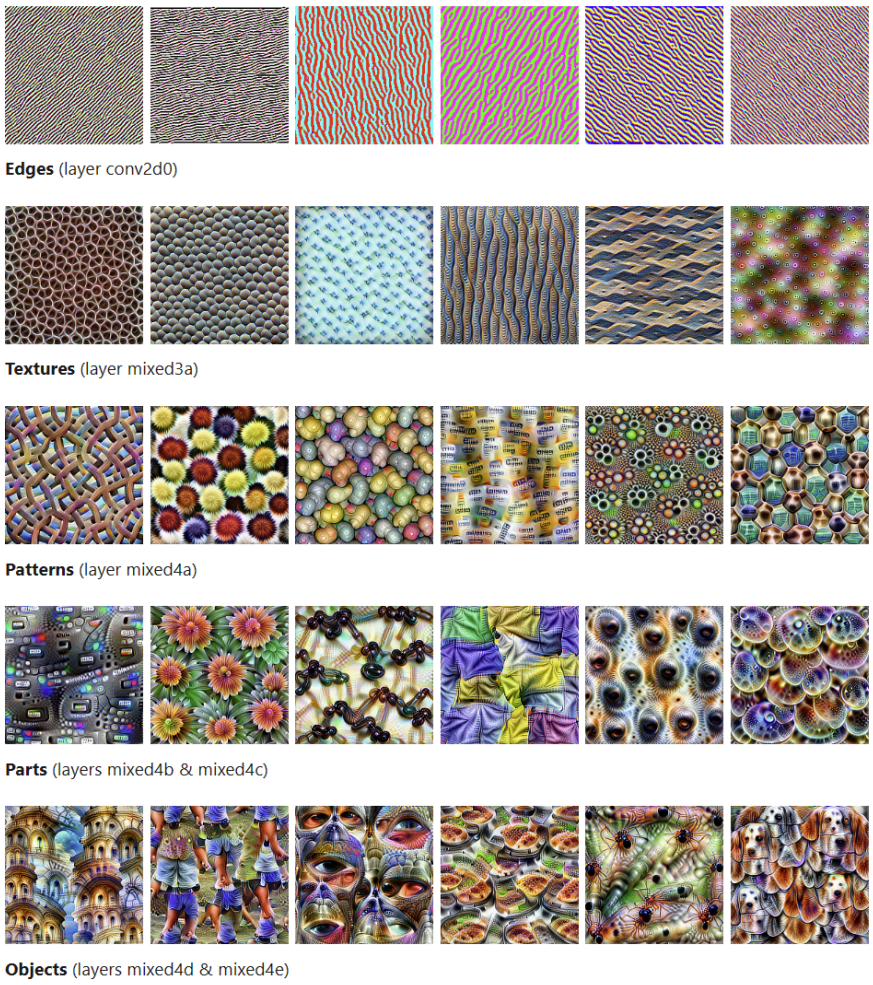


图 2.8 GoogLeNet 中各网络层的激活最大化效果[37]

在 GoogLeNet 中第一个卷积层学习到像边缘和简单纹理这样的特征；后续的卷积层学习到更复杂的纹理和图案等特征；最后的卷积层学习到像物体或物体部分这样的特征；全连接层学会将高层特征的激活与要预测的个别类别联系起来。激活最大化的结果是以随机输入为起点优化而来，当然也可以使用训练集或测试集中的样本逐个测试目标解释单元的激活程度，以此挑选出最佳样本。

激活最大化主要对用以图像分类的卷积神经网络进行特征可视化。然而，从

技术上讲，也可以为全连接神经网络（用于处理表格数据）或者循环神经网络（用于处理文本数据）中的神经元找到最大程度激活的输入样本。例如，在信用违约预测中，输入可能包括之前的信贷数量、手机合约数量、地址以及其他数十个特征，神经元学到的特征将是这些特征之间的某种组合。对于循环神经网络而言，更好的方法是可视化网络学到的内容：Karpathy 等人[59] 表明循环神经网络确实学习到了可解释的特征。他们训练了一个字符级别的模型，从先前的字符中预测序列中的下一个字符。当开括号“(”出现时，某个神经元被高度激活，而当匹配的闭括号“)”出现时，则停止激活；其他神经元在行尾激活；还有一些神经元在 URL 中激活。与卷积神经网络的特征可视化不同之处在于，这些示例不是通过优化得到的，而是通过研究训练数据中的神经元激活情况得出的。

2.2.2 特征归因

特征归因是一种建立输入输出归因关系的可解释性算法，其主要目标是寻找最能影响分类分数的输入特征。在图像分类任务中，特征归因方法突出展示了对分类结果影响最大的像素点。实际上，特征归因方法有许多别称，如敏感性图、显著性图、像素归因图、特征相关性以及特征贡献等。值得一提的是，在模型未知方法中，SHAP 和 LIME 等方法也属于特征归因方法的范畴，但这一类方法并未考虑神经网络的内部结构，因此在此暂不做详细介绍。

神经网络可解释性中的大量工作都集中于特征归因上，对此进行简单分类：

1. 基于扰动和遮挡：直接通过处理输入的部分特征以生成解释；
2. 基于梯度的方法：通过计算输出相对于输入特征的梯度用作每个输入特征的贡献度或显著性分数；
3. 基于反向传播的方法：通过定义反向传播的规则，将输出反向传播至每个输入特征以此作为显著性分数。

基于扰动的方法[60]通过对输入样本部分信息增加扰动并观察输出结果的变化情况，从而确定输入样本中扰动部分对输出结果的影响，即对于一个神经网络 $f(x)$，输入样本的哪些区域对输出值具有较大贡献。

具体地，考虑输入 $x \in R^d$，分类器 f，输出结果为 $f(x) \in R^n$，其中类别 c

的 Softmax 分数为 $f^c(x)$，扰动的掩码为 M，输入对于类别 c 的重要分数为：

$$score_x^c = f^c(x) - f^c(M \times x). \tag{2.7}$$

更具体地，考虑如图2.9所示的简单双输入网络，首先经过一次前向计算得到网络的输出 Out。

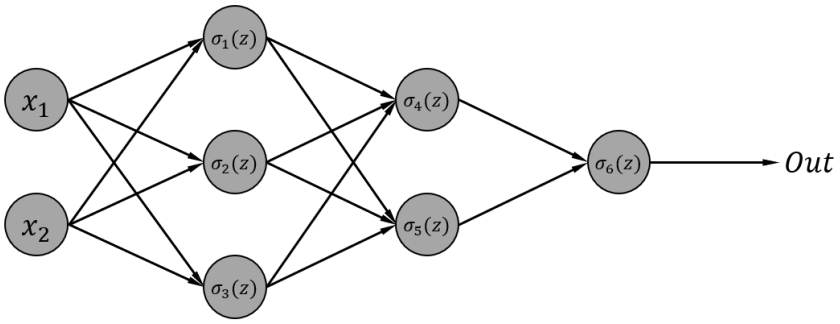


图 2.9 双输入网络示例

考虑第一个输入，对其进行扰动得到新的输入 x_1，并再进行一次前向计算得到网络的输出 Out_1。考虑第二个输入，对其进行扰动得到新的输入 x_2，并再进行一次前向计算得到网络的输出 Out_2。

在此基础上即可得到输入对于此输出的分数分别为 $score_1 = Out - Out_1$ 和 $score_2 = Out - Out_2$。比较每个输入的分数大小，分数越大表示该输入对结果的影响越大，分数越小则影响越小。对于完整的图像输入，对每个输入进行扰动并得到对应的分数，便可以得到整个输入图像的显著性图。图2.10展示了在 ImageNet 数据集上的解释结果，左图为输入原图、中间为随机扰动后的输入图像，右图表示解释算法生成的显著图，通过显著图能够展示网络关注了整个物体的大致轮廓。但基于扰动的方法会产生大量的噪声，使得显著图同样会产生大量噪声。

此外，对于网络的每一个输入，都要进行一次前向传播，并得到其干扰之后的结果。在基于扰动的方法中，一个重要的问题是使用什么样的方法对输入样本进行扰动，也就是使用输入样本的哪些变体进行研究，常见的方法有恒定值扰动、噪声扰动和模糊扰动等，在此不做展开。

基于梯度的方法通常有三个步骤：

1. 选取输入样本 I_0，并进行前向推理，得到输出结果 S_c；

图 2.10 基于扰动的解释算法显著图结果

2. 计算输出结果对于每个输入特征对应的梯度，即：

$$E_{grad}\left(I_0\right) = \frac{\partial S_c}{\partial I}|_{I=I_0}, \tag{2.8}$$

3. 将计算得到的梯度进行可视化。

基于梯度的方法期望用一个线性表达式去近似神经网络决策过程，对于类别 c 的分数 S_c，希望以线性回归模型去近似分数的获取过程：

$$S_c\left(I\right) \approx w^T I + b. \tag{2.9}$$

其中：

$$w = \frac{\partial S_c}{\partial I}|_{I=I_0}. \tag{2.10}$$

梯度类激活映射[61]（Grad-Class Activation Mapping，Grad-CAM）是基于梯度方法的一个代表，是类激活映射[62]（Class Activation Mapping，CAM）的“梯度版本”。在介绍梯度类激活映射之前，本章将先简单介绍类激活映射。

卷积神经网络中的卷积操作可以当作图像处理中的滤波器，对图像的局部特征进行提取与处理。每一层的特征图与原始输入图像存在空间上的对应关系，随着层数的增加，感受野也随着增加。故经验地认为，较浅的卷积层在提取较为低级的语义特征如边缘、纹理等，而较深的卷积层则是处理高级的语义特征，与对象有着更强的关系。

类激活映射方法需要在最后的卷积层后面添加一个全局平均池化层（Global Average Pooling，GAP）。如图2.11所示，通过全局平均池化去捕捉每一个通道的

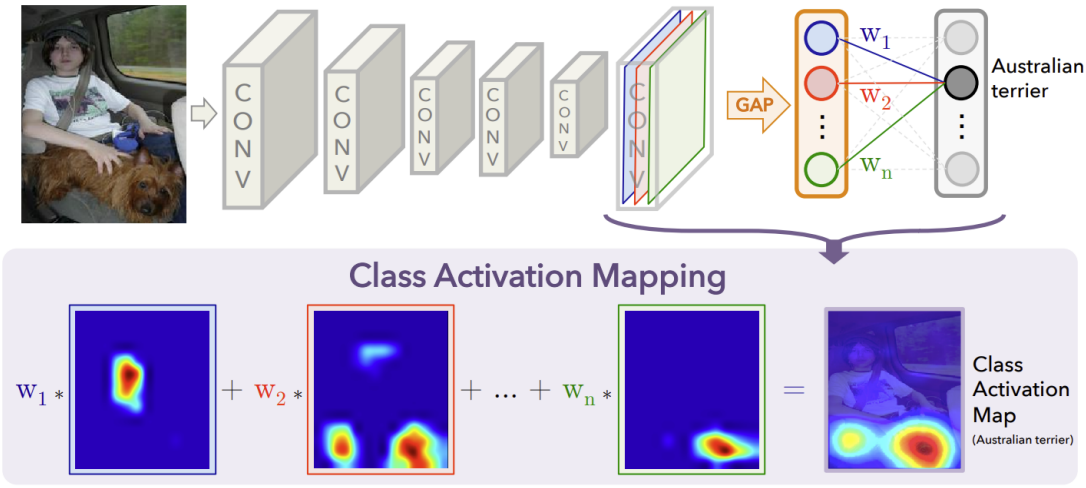


图 2.11 CAM 过程[62]

特征，再与输出层进行全连接，其中 W_1 至 W_n 为全局平均池化与输出（即目标类别）的权重，该权重可以直接当作特征图对目标类别 S_c 的贡献程度，对特征图进行加权求和即可得到 CAM 结果，计算每个位置的显著性分数的公式为：

$$M_c(x,y) = \sum_k w_k^c f_k(x,y), \tag{2.11}$$

其中 $f_k(x,y)$ 为输入像素点位置 (x,y) 在最后一层特征图的值。

但类激活映射方法严重依赖于全局平均池化层，需要对网络结构进行改动甚至重新调整参数。为此，梯度类激活映射希望用梯度的方法去表示特征图的权重，从而去除对全局平均池化层的依赖。即通过计算的方法获取 w_k^c：

$$w_k^c = \frac{1}{Z}\sum_i \sum_j -\frac{\partial y^c}{\partial f_{ij}^k}. \tag{2.12}$$

除了权重的计算不同，梯度类激活映射的其他步骤与类激活映射相同，但梯度类激活映射摆脱了全局平均池化的限制，并且可以扩展到任意特征层而非局限于最后一个卷积层。图2.12展示了梯度类激活映射所生成的显著性图，从左到右分别为原图、显著性图和原图与显著性图叠加图。可以从显著性图看到网络对于目标类别的关注点即显著性图中高亮部分，多数与目标对象重叠，表明网络从这些关键区域提取特征对目标进行识别。

基于反向传播的方法则致力于构建合理的反向传播法则，将输出结果反向

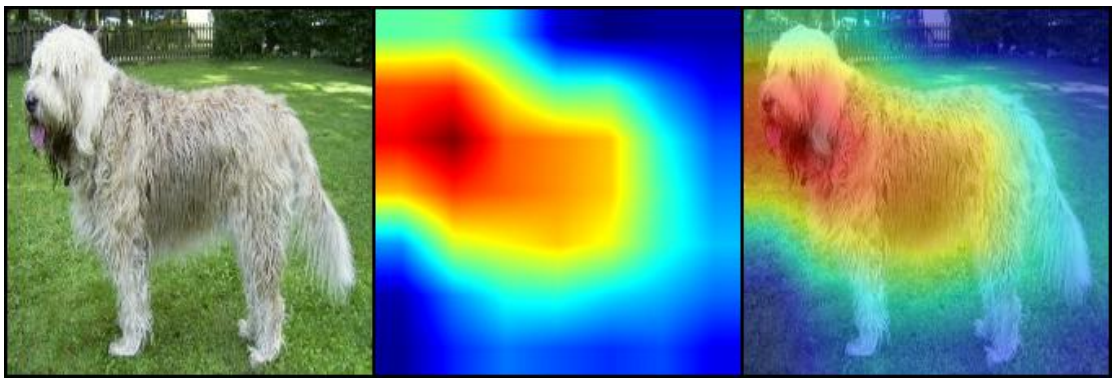


图 2.12 Grad-CAM 显著图示例

传播至输入空间，如反卷积网络[60]（Deconv Net）、相关性分数逐层传播[63]（Layer-wise Relevance Propagation，LRP）等方法。反向传播方法可以基于卷积设计反卷积思路，将目标输出通过反卷积计算前向到输入层；相关性分数则重新设计反向传播法则，将目标输出逐层反向传播到输入层

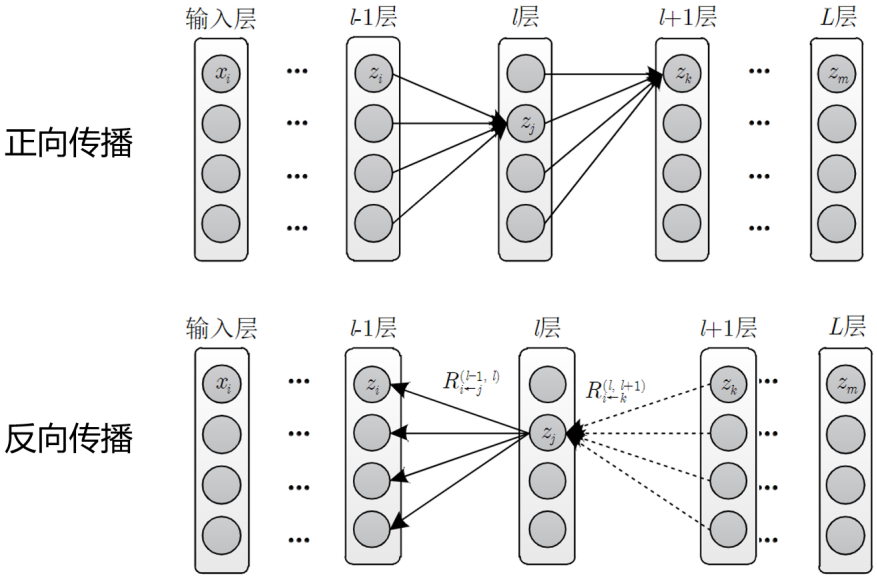


图 2.13 正向传播与反向传播

相关性分数逐层传播方法设计了一整套反向传播的规则，记 R 为相关性分数。

规则一：

- x_i 对输出结果 $f_c(x)$ 贡献为正，则 $R(x_i) > 0$。
- x_i 对输出结果 $f_c(x)$ 贡献为负，则 $R(x_i) < 0$。

规则二：输入空间所有特征的贡献值总和等于输出结果 $f_c(x)$，即：

$$f(x) = \sum_{d=1}^{V} R_d. \tag{2.13}$$

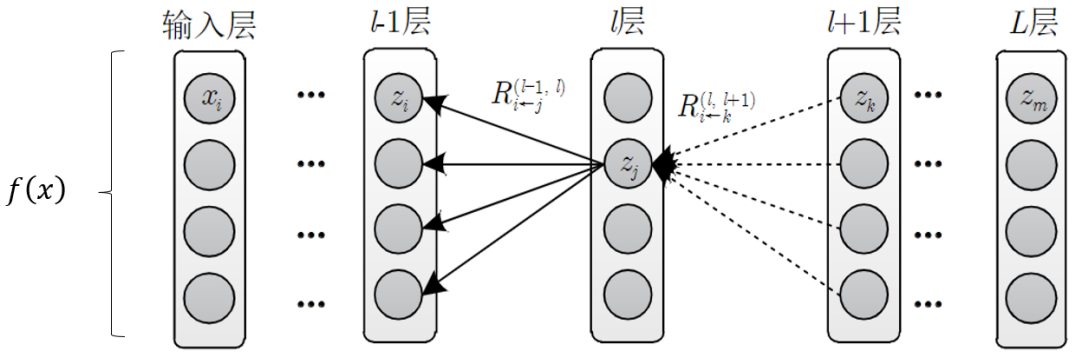


图 2.14 LRP 规则二

规则三：层级相关性分值守恒，即：

$$f(x) = \cdots = \sum_{d=1}^{V(l+1)} R_d^{(l+1)} = \sum_{d=1}^{V(l)} R_d^{(l)} = \sum_{d=1}^{V(l-1)} R_d^{(l-1)} = \cdots = \sum_{d=1}^{V(1)} R_d^{(1)}. \tag{2.14}$$

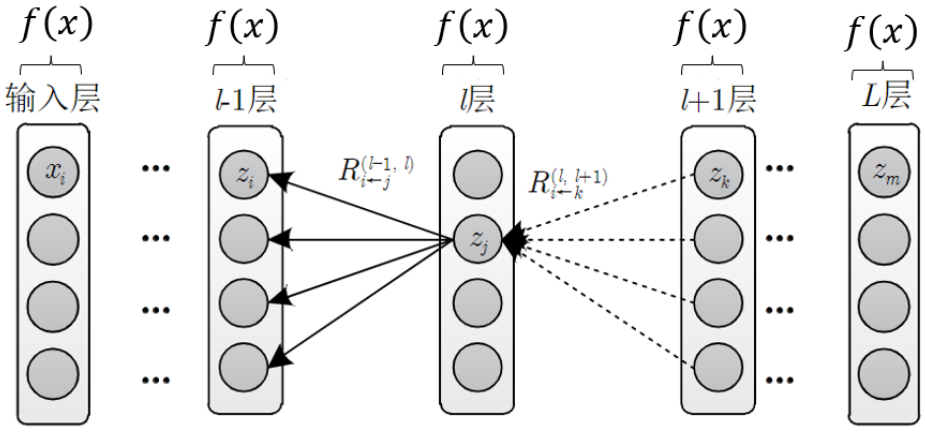


图 2.15 LRP 规则三

规则四：第 l 层某个神经元分解出的相关性值等于其流向第 $l-1$ 层中所有神经元的相关性值之和。

规则五：第 l 层某个神经元流入的相关性值等于第 $l+1$ 层中所有流向该神经元的相关性值之和。

$$R_j^{(l)} = \sum_{i=1}^{V(l-1)} R_{i\leftarrow j}^{(l-1,l)} = \sum_{k=1}^{V(l+1)} R_{j\leftarrow k}^{(l,l+1)}. \tag{2.15}$$

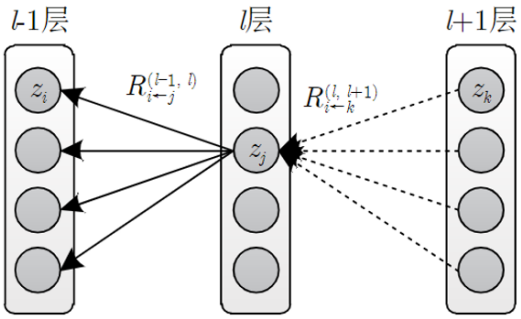


图 2.16 LRP 规则四和规则五

基于这样的规则，很容易反向得到所有的输入关于输出结果的相关性分数，图2.17展示了 LRP 算法的一些输出示例。LRP 算法相较于基于梯度的方法，计算更为简单，但仍需进行正向推理和反向传播两次计算。这种方法其实可以扩展到任意神经元的激活值反向传播至输入空间。

图 2.17 LRP 算法得到的结果

2.3 本章小结

本章主要介绍了神经网络的基本原理，以及神经网络可解释性中特征可视化和特征归因两种方法。

对于神经网络原理，本章简单介绍了神经元的构成以及简单的神经网络，之后针对具体任务介绍了不同的神经网络；对于神经网络可解释方法，本章针对特征可视化与特征归因两种方法进行了介绍，并对一些经典方法做了较为详细的说明与分析。

目前，针对神经网络可解释性的方法层出不穷，尤其在特征可视化和特征归因方面。由于篇幅限制，本文未能对所有方法进行详细介绍。大多数方法都存在一些局限性和不足之处：基于扰动和遮挡的方法缺乏对网络结构的考虑；基于梯度的方法不稳定，而且计算梯度需要的计算量较大；基于反向传播的方法需要进行至少两次推理计算。尽管如此，这些方法都为后文介绍的本文方法提供了基础，为解决神经网络可解释性问题提供了重要思路。

第三章　基于前向传播的特征可视化可解释算法

前文介绍了当前特征可视化的主流方法，本文在这些主流方法的基础上，提出一种基于前向传播的特征可视化方法，用以生成神经元粒度的特征图像，并在通道维度上对特征图像进行融合，得到了具有语义信息的解释结果。最后通过聚类方法以及剪枝优化方法验证了该解释结果的正确性。

3.1　特征可视化与结果

前文介绍过基于激活最大化方法对目标解释单元生成对应理想样本的情况，并在图2.8中给出了算法结果示例。可以看出这种基于优化的方法能够生成含有一定语义信息的解释结果，能够成功显示目标解释单位对特定图像或特定模式的输入具有较高的激活程度。特别地，该方法对卷积神经网络的解释结果，符合工程师的经验假设：神经网络在浅层网络中关注细节上的特征如边缘、纹理等，在较深的层中随着感受野的增加进而关注与对象更为相关的特征，并出现特征的组合。但这类方法仍然存在缺陷：

1. 这些方法只能说明特定模式或图案会导致目标单元激活程度最高，并不能直接验证网络中该目标单元确实学习到了某种语义信息；
2. 解释结果是基于全局的网络权重进行优化得到的，因此对于具体的输入样本，仍然无法确定网络对其做出决策的依据。

理想中的特征可视化方法应该至少包含以下特点：

- 特征可视化结果应尽量与输入样本处于同一向量空间，便于与输入样本直接进行对比相似；

- 特征可视化目标解释单元应当不受限；
- 特征可视化结果应尽量解释输入样本与目标解释单元之间的关系。

以线性回归模型为例，其数学表达式为：

$$y = w_1 x_1 + w_2 x_2 + b, \tag{3.1}$$

其中 (x_1, x_2) 为输入样本，(w_1, w_2, b) 为模型的参数。以输出结点为目标解释单元，则优秀的特征可视化方法得到结果首先应该是与输入同一个维度，即

$$V \in \mathcal{R}^2. \tag{3.2}$$

其次该结果应该能得到输入与输出之间的关系，即

$$V = F\left(y, \left(x_1, x_2\right)\right), \tag{3.3}$$

其中 F 表示输入与输出的某种关系。

3.2 基于前向传播的特征可视化可解释算法设计

针对上一节提出的特征可视化需求，本文提出了一种基于前向传播的特征可视化方法。从前文内容中可以得知，相关性分数逐层传播算法需要经历两次推理过程：第一次前向传播得到所有神经元的输出值，第二次反向传播得到相关性分数。为了解决这个问题，本文重新定义了前向传播规则，使得在一次前向传播中即可得到相关性分数。

3.2.1 FRP

前向相关性分数传播方法（Forward Relevance Propagation，FRP）源自对神经元结构的思考。每个神经元都分为两部分，一部分负责接收整合，另一部分负责激活输出。对于接收整合部分，对所有的输入进行加权求和的过程，是可解释性较强的线性部分。对于该部分算法设计，希望每个目标解释单元产生的解释结

果同样是满足神经网络的整合部分，即可以通过加权求和的方式进行前向传播，同时希望其能够始终保持与输入样本相同的维度与尺寸，使得其更易于解释与理解。

以将 ReLU 作为激活函数的全连接神经网络第一层第一个神经元作为示例，考虑输入样本 $X \in \mathcal{R}^N$，其输出值的计算方式为：

$$Out_1^1 = \mathrm{Max}\left(\sum_{i=1}^{N} w_{1i}^1 x_i + b, 0\right), \tag{3.4}$$

其中 w_{1i}^1 是第一层第一个神经元与输入第 i 个特征的连接权重。

前向相关性分数传播方法在以第一层第一个神经元为目标解释单元时，为保证解释结果能够与输入样本处于同一向量空间，选择逐点相乘的方法，这样做的目的是对任意尺寸的输入样本都可以使解释结果保持同样的尺寸，其计算方法为：

$$V_1^1 = \left(w_1^1 \odot X + W_b\right) \times |\mathrm{sgn}\left(Out_1^1\right)|, \tag{3.5}$$

其中 $\odot$ 表示矩阵的 Hadamard 乘积。对应神经元中的激活输出部分，算法使用符号函数 sgn 来让解释结果的输出与神经元的激活保持一致，即当神经元激活时，解释结果不变；神经元不激活时，解释结果置为全零。其中 sgn 函数的表达式为：

$$\mathrm{sgn}(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{if } x = 0, \\ -1, & \text{if } x < 0. \end{cases} \tag{3.6}$$

从前文中可以知道 w_1 即第一个神经元的连接权重矩阵与输入样本尺寸相同，这里将 w_1 与 x 逐点相乘，并保持了输入的原始尺寸。W_b 是一个偏置矩阵，大小与输入尺寸一样，每个元素值相同，为 b/N。

$$\left(W_b\right)_{x,y} = \frac{b}{N}. \tag{3.7}$$

对于任意层神经元，算法希望解释结果如同特征值在神经网络一样进行前向传播，于是借助网络本身的权重，使解释结果如同输入一样在网络中进行前

向传播，则有：

$$V_i^l = \left(\sum_{j=1}^{(l-1)} w_{ij}^{(l)} V_j^{l-1} + W_{b_i^l} \right) \times |\mathrm{sgn}\left(Out_i^l\right)| \text{ where } l \neq 1. \tag{3.8}$$

总结一下前向相关性分数传播方法的计算原理：

- 第一层神经元的解释结果 V_i^1 计算方法需要让输入与矩阵权重进行逐点相乘并加上偏置权重，如公式3.5；
- 其余层的神经元的解释结果 V_i^l 则参考正常的神经网络前向传播方式，将前一层的 V_i^{l-1} 与权重参数加权求和，并加上偏置权重，如式子3.8；
- 当激活函数为 ReLU 函数时，若原神经元输出值为零则对应的 V 置为零矩阵。

3.2.2 FRP 的特点分析

前向相关性分数传播方法产生的算法结果满足前文所提出的特征可视化所需具备的特点，以下给出证明。

对于 V_i^l，满足 V_i^l 与输入 X 形状相同，都属于同一个向量空间。对于第一层有：

$$W_b \in \mathcal{R}^N \text{ and } w_1 \odot X \in \mathcal{R}^N, \tag{3.9}$$

对于其他层有：

$$V^{(l-1)} \in \mathcal{R}^N \text{ and } W_{b_i^l} \in \mathcal{R}^N, \tag{3.10}$$

故：

$$V_i^l = \left(\sum_{j=1}^{(l-1)} w_{ij}^{(l)} V_j^{l-1} + W_{b_i^l} \right) \in \mathcal{R}^N. \tag{3.11}$$

前向相关性分数传播方法生成的所有结果都处于与输入样本相同的向量空间中，并且与输入样本保持良好的相关性。这样做使得结果更易于理解和解释，因为它们直接与原始输入数据相关联，从而提供了更清晰的联系和上下文理解，并可以从中得出语义相关的特征与解释。

V_i^l 与输入 X 以及目前解释单元的输出值 Out_i^l 皆有联系。在未被负值过滤

的通路里，V_i^l 是输入 X 的线性变换，并且 V_i^l 中每个元素 $\left(v_i^l\right)_{(m,n)}$ 之和等于对应目标解释单元的输出值：

$$Out_i^l = \sum_m \sum_n \left(v_i^l\right)_{(m,n)}, \tag{3.12}$$

对于第一层神经元的输出值有：

$$Out_i^1 = \sum_{j=1}^{N} w_{ij}^1 x_j + b, \tag{3.13}$$

而对 V_i^1 中每个元素 $\left(v_i^1\right)_{(m,n)}$ 之和有：

$$\sum_m \sum_n \left(v_i^1\right)_{(x,y)} = \left(\sum_m \sum_n x_{(m,n)} \times \left(w_i^1\right)_{(m,n)} + N \times \frac{b}{N} \right) \times \left| \operatorname{sgn}\left(Out_i^1\right) \right| \tag{3.14}$$

$$= \left(\sum_{j=1}^{N} w_{ij}^1 x_j + b \right) \times \left| \operatorname{sgn}\left(Out_i^1\right) \right| \tag{3.15}$$

$$= Out_i^1, \tag{3.16}$$

同样地，对于其他层神经元的输出有：

$$Out_i^l = \sum_{j=1}^{(l-1)} w_{ij}^l Out_j^{(l-1)} + b_i^l, \tag{3.17}$$

同样地，对其解释结果有：

$$\sum_m \sum_n \left(v_i^l\right)_{(x,y)} = \left(\sum_m \sum_n \left(\sum_{j=1}^{(l-1)} w_{ij}^{(l)} \left(V_j^{l-1}\right)_{(m,n)} \right) + N \times \frac{b_i^l}{N} \right) \times \left| \operatorname{sgn}\left(Out_i^l\right) \right| \tag{3.18}$$

$$= \left(\sum_{j=1}^{(l-1)} w_{ij}^{(l)} \left(\sum_m \sum_n \left(V_j^{l-1}\right)_{(m,n)} \right) + b_i^l \right) \times \left| \operatorname{sgn}\left(Out_i^l\right) \right| \tag{3.19}$$

$$= \left(\sum_{j=1}^{(l-1)} w_{ij}^l Out_j^{(l-1)} + b_i^l \right) \times \left| \operatorname{sgn}\left(Out_i^l\right) \right| \tag{3.20}$$

$$= Out_i^l, \tag{3.21}$$

即前向相关性分数传播方法产生的结果是将目标解释单元的输出值按网络本身的权重分至输入空间中。

现在重新看待本章第一节提出的线性回归模型即公式3.1，对于输出结点使用前向相关性分数传播方法产生的解释结果为 $[w_1x_1 + b/2, w_2x_2 + b/2]$，该解释结果首先与输入 $[x_1, x_2]$ 有着相同的维度，且每个位置一一对应。同样地，解释结果中所有元素之和与目标解释单元的输出相等：$y = w_1x_1 + b/2 + w_2x_2 + b/2$，即解释结果是将目标解释单元的输出分散至输入空间中，如图3.1所示。这样得到的解释结果可以从两个方面进行解释：

- 两个输入对输出的贡献分别为 $[w_1x_1 + b/2, w_2x_2 + b/2]$。
- 以输出结点视角看到的输入样本是 $[w_1x_1 + b/2, w_2x_2 + b/2]$。

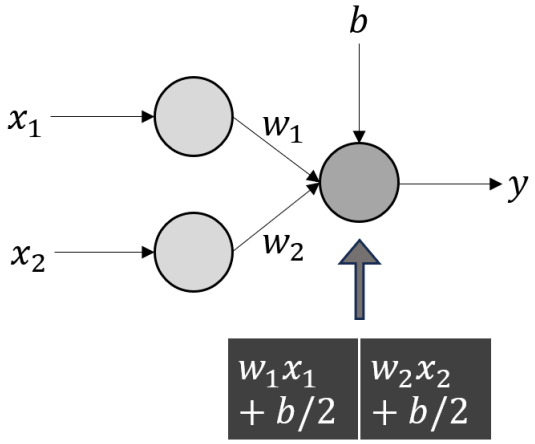


图 3.1 线性回归模型中输出结点看到的输入样本

3.3 实验与分析

为了验证前向相关性分数传播方法的效果，本文在图像分类任务场景下，对相应的神经网络进行特征可视化，对生成的解释结果进行了分析，并设置实验检验了解释结果。

3.3.1 实验设置

数据集

MNIST[64]。MNIST 数据集是一个经典的手写数字识别数据集，被广泛用于测试各种机器学习和深度学习算法的性能。该数据集由美国国家标准与技术研究所（NIST）的员工和高中生手写的 70 000 个数字图像组成，其中包括 60 000 个用于训练模型和 10 000 个用于测试模型的图像。MNIST 数据集中的每个图像都是一个灰度图像，大小为 28x28 像素，表示了一个手写数字（0 到 9 中的一个）。

CIFAR-10[65]。CIFAR-10 数据集是一个用于识别普适物体的小型数据集。它包含 10 个类别的 60 000 个 32x32 彩色图像，每个类别有 6 000 个图像，共有 50 000 个训练图像和 10 000 个测试图像。该数据集由 Alex Krizhevsky, Vinod Nair, 和 Geoffrey Hinton 提出，并用于机器学习和计算机视觉研究。

模型

多层感知机（MLP），使用一个网络结构为 $28 \times 28 \rightarrow 192 \rightarrow 120 \rightarrow 60 \rightarrow 10$ 的多层感知机处理 MNIST 数据集，其中每个隐藏层后面增加一个 ReLU 层作为激活函数。

卷积神经网络（CNN），使用 AlexNet 作为卷积神经网络处理 CIFAR-10 数据集。AlexNet 是一个由 8 层隐藏层组成的深度模型，其中包含 5 个卷积层和 3 个全连接层。同样地，本次使用的 AlexNet 激活函数被替换为 ReLU 函数。

实验环境

本次实验所有程序在课题组的 GPU 服务器上进行，使用的显卡型号为 1080Ti，使用 Python 进行代码编写，并采用 PyTorch 作为深度学习框架，具体配置见表3.1。

表 3.1 实验配置

配置	参数
操作系统	Ubuntu 20.04.3 LTS
GPU	GeForce GTX 1080 Ti
CPU	Intel(R) Xeon(R) CPU E5-2620 v4
Python	3.10.13
PyTorch	2.0.1

3.3.2 特征可视化结果

对于多层感知机在 MNIST 上的表现，本文对每个神经元进行了特征可视化分析，图3.2展示了多层感知机的特征可视化效果，图中的目标解释单元从各层

随机挑选，每张图下面标注 Layer*x*_*y* 表示目标解释单元为网络中第 *x* 层第 *y* 个神经元。其中 Layer2_39 与 Layer4_53 是经过 ReLU 函数之后的神经元，因存在负值而被置零。

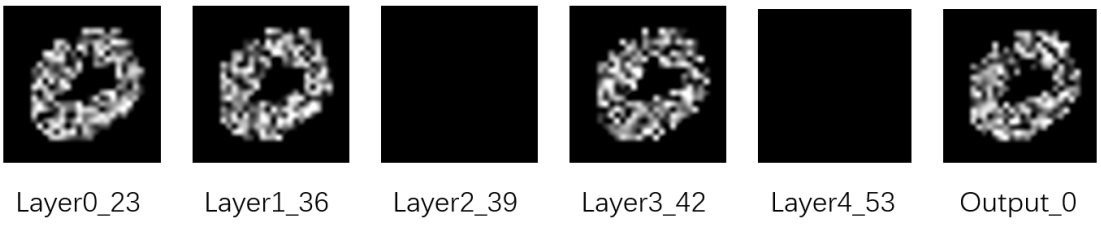


图 3.2 多层感知机对同一输入的可视化结果

对于 MNIST 数据集，多层感知机模型能够轻松完成分类任务，故在网络的初期，神经网络便能够对图像的特征进行识别与区分，网络中多数神经元均具有识别能力，故浅层神经元的解释结果已经能反映出图像的特征，图3.3展示了不同输入下多层感知机的特征可视化效果。

图 3.3 多层感知机对不同输入的可视化结果

本次实验采用的 AlexNet 在 CIFAR-10 上的性能表现如表3.2所示，在训练集上已经达到超过 99% 的准确率，并且在测试集上达到了约 74% 的准确率。这是因为本次使用的 AlexNet 并未作任何泛化处理，如正则化、归一化等。

表 3.2 AlexNet 在 CIFAR-10 上性能

评估指标	数据
训练集准确率	99.87%
测试集准确率	73.97%

AlexNet 对同一输入的可视化结果如图3.4所示。其中图中图片名字 l_*x*c_*y*w_*z*h_*k* 表示目标解释单元是卷积层中第 *x* 层的第 *y* 个通道中第 *z* 列第 *k* 个神经元，l_*x*No.*y* 则表示目标解释单元是全连接层中第 *x* 层中的第 *y* 个神经元。解释结果符合卷积神经网络的感受野计算，随着网络层数的加深，卷积网络的神经元

感受野也变大，并能感知更多的输入图像信息。图3.4中显示，第 9 层 28 通道 4 行 4 列的神经元有捕捉到飞机尾翼的图像边缘。

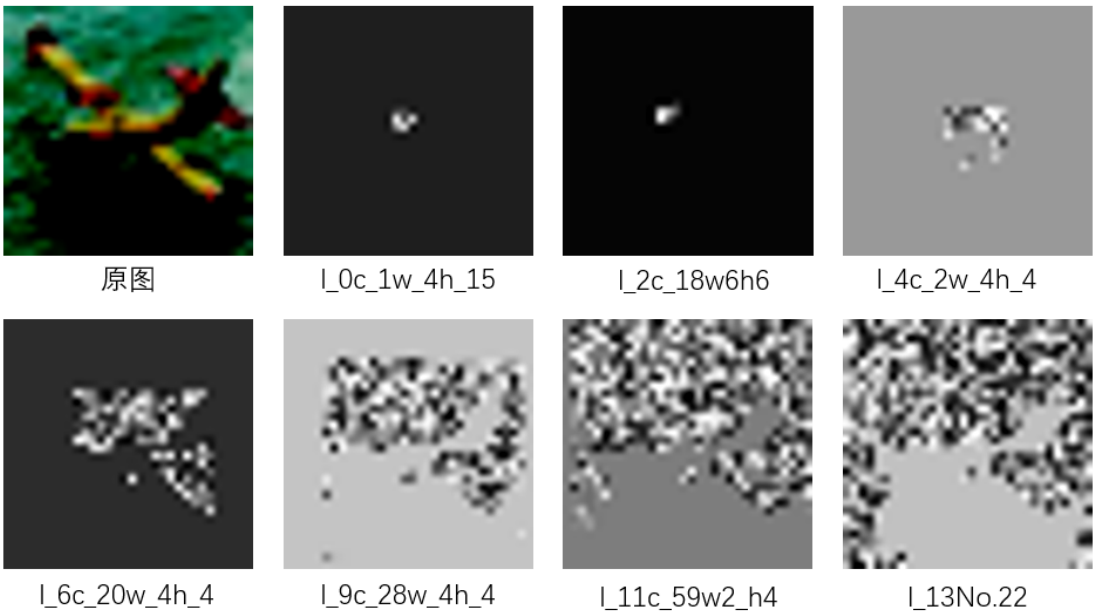


图 3.4 AlexNet 对同一输入不同目标解释单元的可视化结果

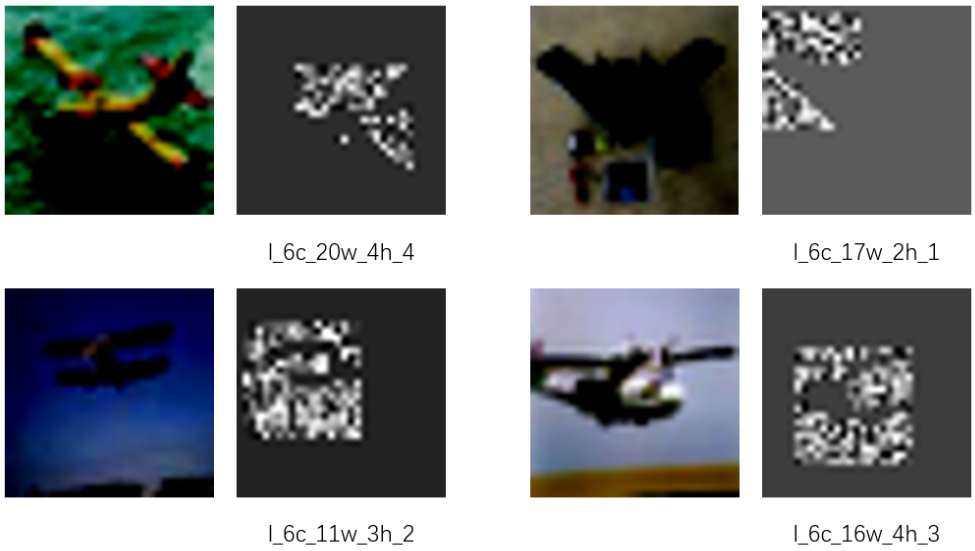


图 3.5 AlexNet 对同类别不同输入的可视化结果

除了感受野大小的显示，解释结果表明 AlexNet 对事物的轮廓捕捉能力也较强，图3.5展示在不同的输入上 AlexNet 的特征可视化结果。对于不同场景下的物体，AlexNet 网络都能够较为准确的识别出物体的轮廓。

卷积神经网络的语义信息常常被认为处于通道级别之中，即每个通道负责了不同的特征提取功能与识别。故将解释结果在通道级别上进行融合，以此提

取通道级别上的语义信息。对于通道上的解释结果，其计算方法为：

$$V_c^l = \sum_w \sum_h V_{c,w,h}^l, \tag{3.22}$$

其中 $V_{c,w,h}^l$ 是第 l 层第 c 个通道第 w 列第 h 个神经元的解释结果。

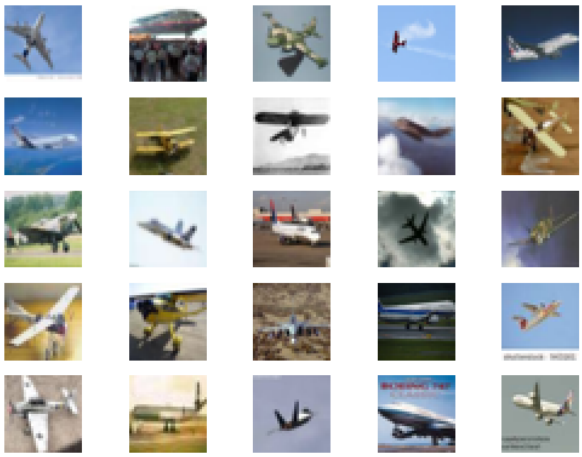


图 3.6 CIFAR-10 中标签为飞机的图像

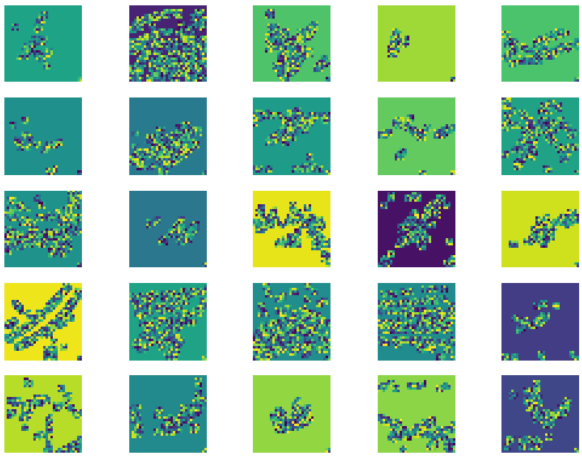


图 3.7 AlexNet 第 1 层第 39 通道解释结果

对 AlexNet 输入相同类别的图像，探究卷积神经网络如何在通道级别上去挖掘相似的语义信息。图3.6是输入的图像，本文得到一批次在相同通道上有相似表现的解释结果。图3.7是第 1 层 39 通道的解释结果，这里对解释结果进行了

viridis[66]色彩映射，将灰度图映射至默认的色彩以便于阅读。图3.7解释了第 1 层 39 通道对图片前景背景的区分与捕捉，与原始输入图像对比，解释结果的高亮区域基本是位于前景的物体，甚至是前景的字体，说明该通道神经元只对前后景敏感，并未具体区分前景中的具体事物。

图3.8所解释的目标是第 2 层的 62 通道，相比于第 1 层的解释结果，该通道不再对所有的物体都感兴趣，而是注意特定的具体的目标。位于图像前景的文字不再受到关注，模型转而对机体的全部或部分区域感兴趣。这体现了通道对语义特征掌握提升。

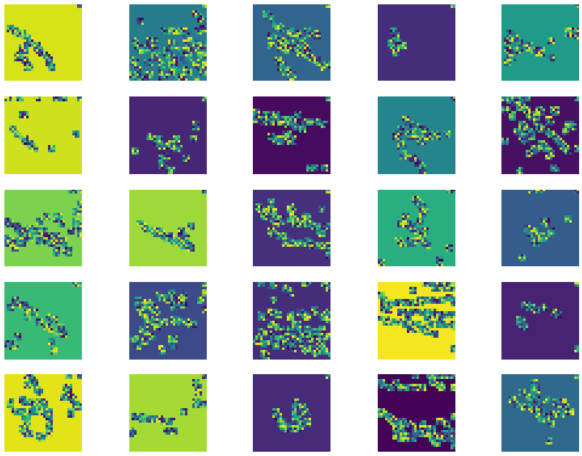


图 3.8　AlexNet 第 2 层第 62 通道解释结果

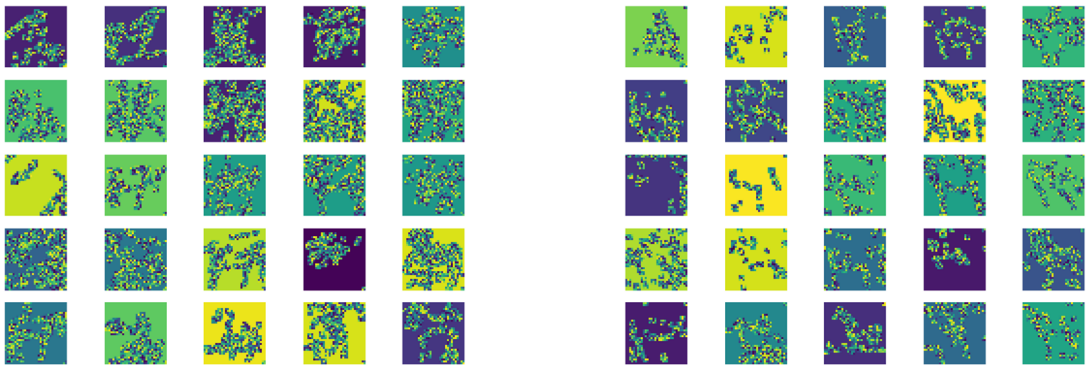


图 3.9　马类图像在通道级别上的解释结果

将其他类别的图像进行同样的可视化，在固定的通道上（如第 1 层 39 通道、第 2 层 62 通道）同样可以得到一样的可视化效果。如图3.9所示，左图中的通道

仍在致力挖掘前景内容，而右图中的通道则增加了语义信息，将前景中的内容进行了筛选。图3.10展示了更为细致的可视化结果，相比于第 1 层 39 通道的解释结果，可以看到第 2 层 62 通道试图将马的躯干与同样属于前景的阴影部分进行区分。

图 3.10 通道捕捉语义特征能力随层数的增加而变强

除了对物体轮廓的提取、前景背景的区分，从解释结果还能看到神经元对部分图案的反应，以及随着网络层数的增加，神经元学习的特征逐渐从具象的低级语义特征向抽象的高级语义特征过渡，图3.11展示了神经元对关键点提取以及抽象语义的提取，其中左图中第 2 层 29 通道尝试捕捉马的四肢与头部，而右图中第 5 层 118 通道则是提取难以直接理解的抽象特征。

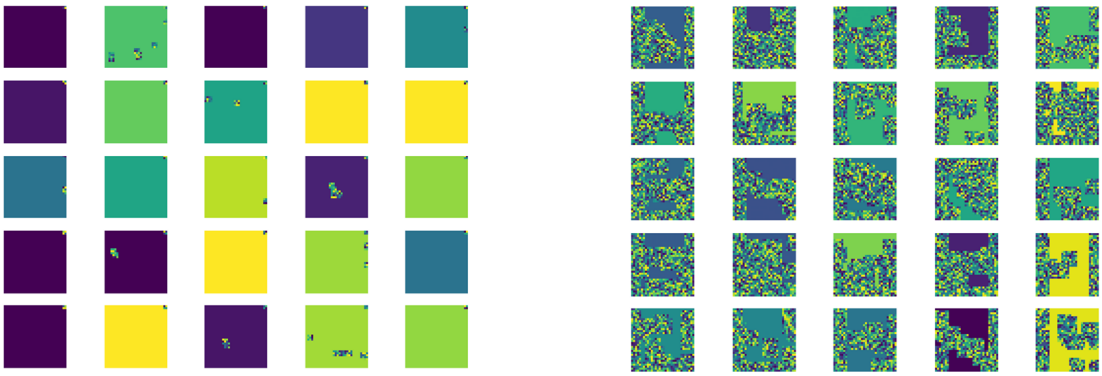


图 3.11 其他类型的语义特征可视化

3.3.3 特征可视化解释分析

上一节展示了前向相关性分数传播方法在多层感知机以及卷积神经网络上特征可视化的效果。为了验证方法的有效性，以通道级别的解释结果为例，本节从两方面设计了实验对其进行验证：

- 通道对网络性能的影响角度。根据解释结果设计结构化剪枝方法，分析挑选的通道对网络性能的影响；
- 通道对输入样本的抽象能力。在通道级别上对特征向量进行聚类分析，验证挑选的通道对输入样本的抽象能力。

剪枝分析

神经网络通过通道对输入样本进行特征转换，不同的通道对网络决策的贡献并不相同，一个朴素的设计是通过剪枝方法来验证各通道对于决策的重要程度。剪枝是一种模型压缩轻量化方法，使模型在性能尽可能不被影响的情况下减少冗余的参数。剪枝方法可以从减少的参数细粒度分为两类：结构化剪枝和非结构化剪枝。

1. 结构化剪枝。结构化剪枝是指剪枝的对象与网络结构直接相关，如网络层剪枝、卷积核剪枝和通道剪枝等。网络层剪枝是指对某一层进行删减；卷积核剪枝[67]指删减整个卷积核参数，如图3.12中左边的示意图，直接将具体的某个卷积核删减，这会导致输出的通道减少；通道剪枝[68]则是删减整个通道的参数，如图3.12中间的示意图，这将忽略接收到的特征向量的某个通道的所有值。
2. 非结构化剪枝[69-70]。非结构化剪枝指剪枝的对象与网络的结构不相关，直接对单个权重参数进行删减，如图3.12中最右边的示意图，直接对卷积核中对应的权重进行删减。

在此算法选择通道剪枝方法，将挑选出来的含有丰富语义特征的通道进行删减并测试性能，记为 good_channel。对照组则是挑选出语义特征较少的通道 bad_channel 以及随机挑选的通道 random，剪枝后得到网络在 CIFAR-10 数据集上进行测试。

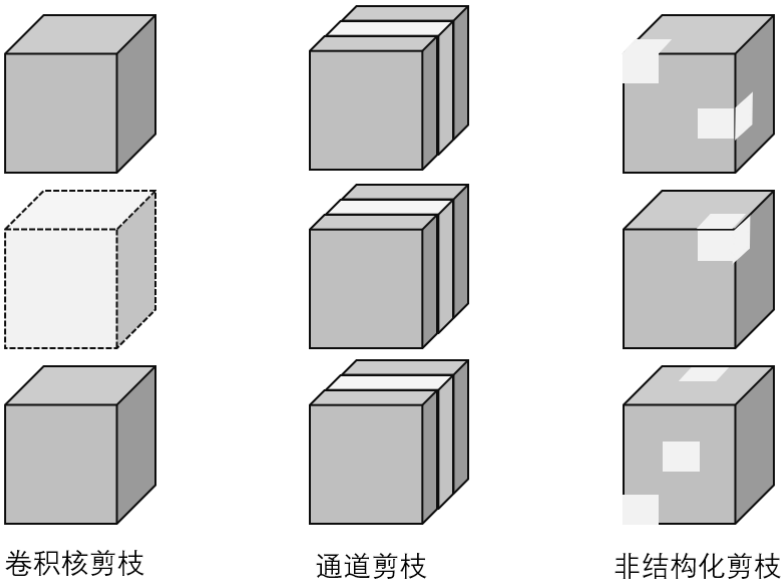


图 3.12 剪枝方法

表 3.3 剪枝后网络性能

方法	训练集准确率	测试集准确率
baseline	99.87%	73.97%
random	91.35%	69.23%
good_channel	**87.47%**	**67.24%**
bad_channel	95.40%	70.56%

测试结果如表3.3所示。对于剪枝工作，其核心是建立合适的指标用作剪枝与否的评估标准，对于神经网络剪枝，核心问题则是找出需要删减的网络权重。实验组 good_channel 中为 20 个人工挑选其特征可视化结果具有丰富语义如边缘、轮廓和前背景等特征的通道；实验组 bad_channel 为 20 个人工挑选的其特征可视化不具有具体特征，且激活较少的通道；random 组则为随机产生的 20 个通道。可见 good_channel 被删减后训练集准确率下降 12 个百分点，测试集准确率下降 6.7 个百分点，皆是下降最多的；bad_channel 被删减后受到影响最小，训练集准确率下降 4.4 个百分点，测试集只下降 3.4 个百分点；random 方法作为参照组，训练集准确率下降 8.5 个百分点，测试集下降 4.7 个百分点。从数据上看，good_channel 对网络性能的影响至关重要，而 bad_channel 对网络性能的影响要低于各个通道的平均表现。特别地，good_channel 在 36 925 322 个参数中只删减了 540 个参数，却导致了超过 10 个百分点的下降，其剪枝率只有约 0.001 4%，这从另一方面说明 good_channel 中捕捉丰富语义特征的通道对神经网络决策至

关重要。

聚类分析

考虑神经网络对输入样本进行变换，理想情况下同类输入样本会被变换至同样或相似的标签。从先验与经验来看，神经网络从不同颜色空间的样本中提取相同的语义特征，转换为特征向量，将稀疏的样本空间转至稠密的特征向量空间。这样，对不同通道的特征向量进行聚类分析，观察不同通道对输入样本的转换能力。这里先验地认为优秀的通道会提取共同的抽象特征，让输入样本更相似。

聚类是一种无监督学习的方法，用于将数据集中的样本分成不同的组（或簇），使得同一组内的样本之间相似度较高，而不同组之间的样本相似度较低。为了不引入新的难以解释的机制，避免出现用神经网络“解释”神经网络的情况，这里采用 K-means 聚类方法[71]，K-means 是基于欧式距离的聚类算法，这里简单认为两个目标的相似度由其欧式距离所决定。对于两个 n 维向量 $\mathbf{x}=\left(x_1, x_2, \cdots, x_n\right)$ 和 $\mathbf{y}=\left(y_1, y_2, \cdots, y_n\right)$ 其之间的欧式距离为：

$$d(\mathbf{x}, \mathbf{y})=\sqrt{\sum_{i=1}^{n}\left(x_i-y_i\right)^2}. \tag{3.23}$$

算法步骤为：

1. 选择初始化的 k 个样本作为聚类中心:$a_1, a_2, \cdots, a_k$。
2. 对于每个样本 x_i 计算其到 k 个聚类中心的距离并将其分到距离最小的聚类中心所对应的簇。
3. 更新每个类别 a_c，重新计算聚类中心 $a_c=\frac{1}{|c_i|}\sum_{x\in c_i} x$。
4. 重复 2、3 步骤，直到满足最小误差变换。

实验选择 CIFAR-10 中两类图片共 10000 张图像，将其通过 AlexNet 的前两层得到 10 000×96 × 32 × 32 的特征向量，按照通道分为 96 组得到 96 组 10 000×32 × 32 的特征向量，将每组向量进行 K-means 聚类，初始簇选择 2，观察聚类各个通道的聚类效果。

评估指标选择调整兰德系数[72]（Adjusted Rand Index，ARI）以及轮廓系数[73]

(Silhouette Coefficient)。调整兰德系数评估聚类方法是否正确将样本分至簇中，考虑了聚类结果与真实类别标签之间的一致性，同时纠正了随机聚类结果所带来的影响。调整兰德系数的计算方法：

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]}, \tag{3.24}$$

其中 RI 是兰德系数[74]，$E[RI]$ 是兰德系数的期望，兰德系数的计算公式为：

$$RI = \frac{a+d}{a+b+c+d}, \tag{3.25}$$

其中：

- a 描述的是在同一簇中，同一真实类别的样本对数。
- b 描述的是在同一簇中，不同真实类别的样本对数。
- c 描述的是在不同簇中，同一真实类别的样本对数。
- d 描述的是在不同簇中，不同真实类别的样本对数。

同样考察人工挑选的 good_channel 与 bad_channel 的情况。调整兰德系数的值域在 $[-1, 1]$，当兰德系数为正值时，表示聚类结果与真实类别标签的一致性高于随机分配；当兰德系数为负值时，表示聚类结果与真实类别标签的一致性低于随机分配；当兰德系数为 0 时，表示聚类结果与真实类别标签的一致性等同于随机分配。实验结果平均调整兰德系数如表3.4所示，ARI(avg) 表示对组内的调整兰德系数求均值。good_channel 得到的调整兰德系数均值最高，说明其聚类结果与真实类别标签的一致性更高；而 bad_channel 得到了最低的调整兰德系数均值，表明其聚类结果与真实类别标签一致性较低，更接近于随机分配。

表 3.4 各组数据聚类的调整兰德系数

channel	ARI(avg)
all	0.079 57
good_channel	**0.085 09**
bad_channel	0.072 20

图3.13展示了所有通道聚类后的调整兰德系数，其中 good_channel 中的特征向量聚类后与真实类别标签分布更接近，表明了 good_channel 对输入样本的共同特征提取能力更强。

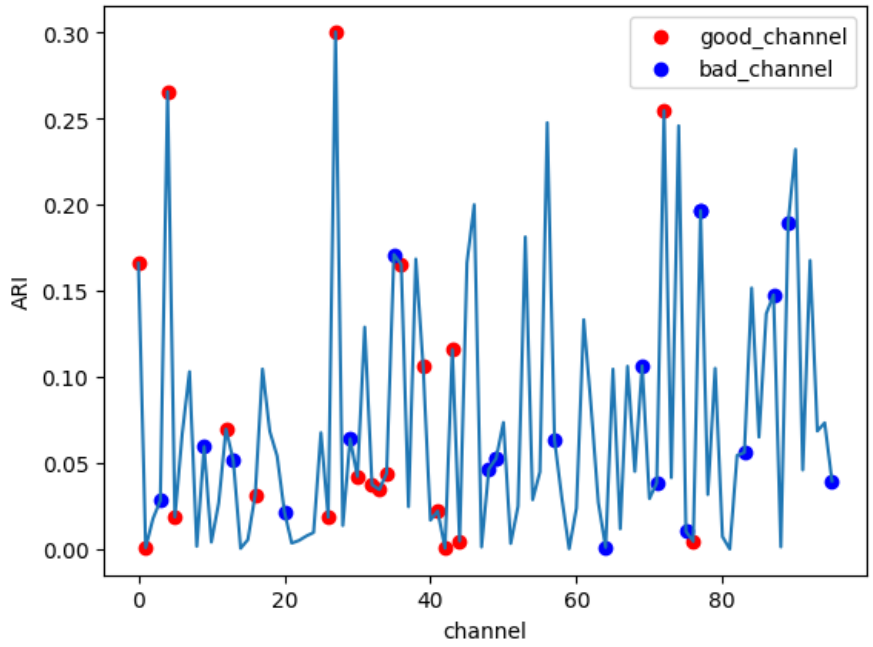


图 3.13 各组数据聚类的调整兰德系数分布

轮廓系数则同时考虑了聚类结果的紧密度（簇内距离）和分离度（簇间距离），用于衡量样本与其所属簇内样本的相似度，以及与其他簇样本的差异度，其公式为：

$$s = \frac{1}{N}\sum_{i=1}^{N} s(i), \tag{3.26}$$

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}, \tag{3.27}$$

其中 $a(i)$ 表示样本与同一簇中其他所有样本的平均距离，$b(i)$ 表示样本与其最近的其他簇中所以样本的平均距离。

轮廓系数的取值为 $[-1, 1]$，轮廓系数越大表示样本与其所属簇内其他样本相似度越高，与其他簇样本的差异度越大，聚类效果更好。组的平均轮廓系数如表3.5所示，good_channel 的轮廓系数均值更高，高于正常所有通道的均值；bad_channel 的轮廓系数均值较低，说明其聚类方法并没有将特征向量分为更为有效的两个簇。

轮廓系数的分布如图3.14所示。可见 good_channel 组中的结果普遍较高，而 bad_channel 则普遍较低，说明这些通道对输入特征的提取并不具有区分度，甚至做了无用转换。然而 K-means 采用的距离指标函数是欧式距离，对于图像数

表 3.5 各组数据聚类的轮廓系数

channel	silhouette(avg)
all	0.240 044 73
good_channel	**0.264 953 74**
bad_channel	0.129 083 13

据，欧式距离并不能特别好描述图像之间的相似性，比如欧式距离不具备平移等变性等适合图像相似度的描述。事实上，对于图像来说暂未有较好的衡量指标可以直接比较两个图像语义上的相似度，这也是图像分类任务试图用模型解决指标缺失问题的原因。图像的语义特征更多来自概念以及人为设定的模式，有数据集工作建立了具有语义概念的图像数据[41]，并用于解释神经网络是否真正具备语义上的识别能力。

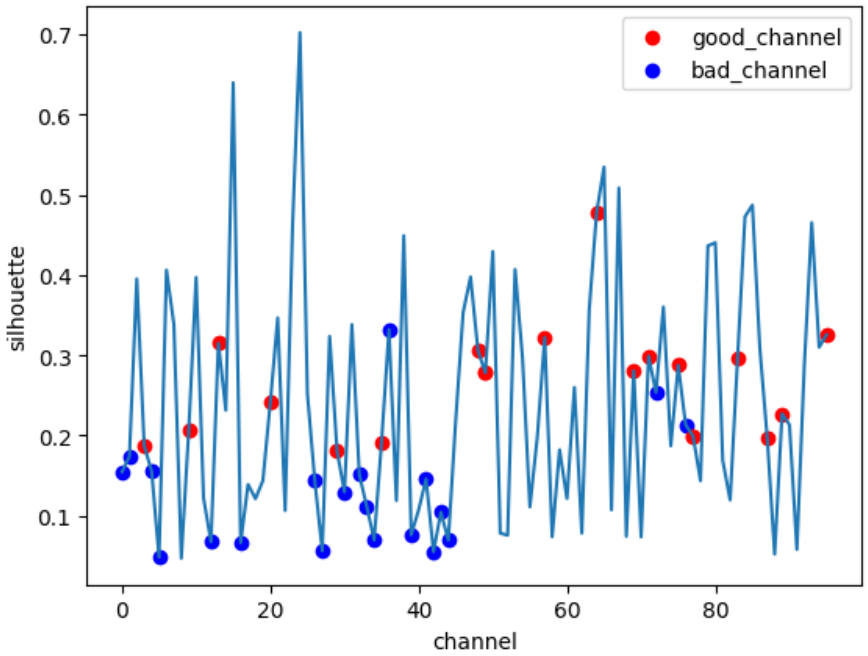


图 3.14 各组数据聚类的轮廓系数分布

剪枝和聚类两个小实验，较为充分地验证了人工挑选的 good_channel 中的通道比 bad_channel 中的通道对网络性能的影响更为重要，对输入特征的提取更为有效。而人工挑选的方法来自特征可视化方法的主观评价，这也从侧面验证了基于前向传播的特征可视化方法的有效性，能够较为有效地将神经网络中重要的通道所提取的输入特征进行可视化分析。

3.4 本章小结

本章首先分析了当前特征可视化工作存在的问题,并给出理想特征可视化方法应当具备的特点。然后基于前向传播的前向相关性分数传播方法，用于计算神经网络目标解释单元的可视化结果，并验证了前向相关性分数传播方法满足理想特征可视化方法应当具备的特点。通过实验，生成了对多层感知机和 AlexNet 的可视化结果，并在通道级别上生成了具有语义信息的可视化结果。最后通过设计剪枝和聚类实验，从侧面验证了前向相关性分数传播方法的有效性。

第四章　基于前向传播的特征归因可解释算法

本章在前文的基础上，以前向相关性分数传播方法为基础，提出一种基于前向传播的特征归因可解释算法（Features Attribution based on Forward Relevance Propagation，FAFRP），用以在多层感知机、卷积神经网络以及循环神经网络上进行特征归因计算，生成输入样本对应的显著性图，以解释输入样本对输出结果的贡献程度。

4.1　特征归因

将神经网络内部特征进行可视化并不能满足人类对神经网络决策依据的认知需求。在多数情况下，模型使用者更多会关注模型的结果是否正确，以及决策的依据。为了贴合人类自身思维习惯，研究者潜意识下会为模型输入与模型输出建立因果关系，即认为导致输出的原因是输入有何种特征。特征归因方法利用这一思维惯性，建立了输入-输出归因范式，对机器学习模型进行解释。其中最常见的方式即计算输入样本对于输出结果的贡献度，也可叫做显著性图。通过可视化对模型决策相关度较高的输入特征，提供对单个预测结果的解释信息，能够增加人类对模型决策的信任，同时解释信息与预测结果也可以互相验证，因此其是一种易于理解的可解释方法。

模型无关的特征归因方法中，因考虑不同特征之间的相互作用，需要对输入样本进行多次变更，并计算其输出结果变化，用以生成显著性图。如基于扰动的方法则需要设计不同的样本扰动方式与区域，统计各种样本变更后输出结果的变化，并计算对应特征的显著性图。这样的计算复杂度在实际应用中对计算资源要求较高，同时生成多个预测结果的解释所需时间也较多。

模型相关的特征归因方法中，大部分方法是基于梯度计算显著性图。而对于单一输入样本而言，其特征空间对输出结果的梯度极其不稳定，导致在显著性图的生成过程中难以保持稳定，并且在生成过程中容易引入噪声。

考虑以上因素，本章提出一个基于前向传播的特征归因算法，并将其推广至卷积神经网络和循环神经网络，用于为单个预测样本高效生成显著性图。

4.2 基于前向传播的特征归因可解释算法设计

4.2.1 多层感知机特征归因算法

利用前向相关性分数传播方法可以直接推广得到多层感知机对于输出结果的特征归因方法。公式3.5和公式3.8给出了对任意层的神经元节点 FRP 解释结果的计算方法。在多层感知机的最后一层，对类别为 c 的输出结点有：

$$V_c^l = \left(\sum_{j=1}^{(l-1)} w_{cj}^{(l)} V_j^{l-1} + W_{b_c^l} \right) \times |\mathrm{sgn}\left(Out_c^l\right)|, \tag{4.1}$$

$$\sum_m \sum_n \left(v_c^l\right)_{(x,y)} = S_c, \tag{4.2}$$

即当目标解释单元被选为最后输出层时，产生的解释结果所有元素值之和即可等于类别 c 的分数 S_c。故对多层感知机而言，当目标解释单元选择最后输出层神经元时，前向相关性分数传播方法可以直接求出特征归因的显著性图。

4.2.2 卷积神经网络特征归因算法

卷积神经网络中的卷积操作是一种具有平移等变性以及局部相关性的矩阵计算方法，其计算方式来自数学上的卷积概念：

$$(f * g)(\mathbf{X}) = \int_{-\infty}^{\infty} f(t)\, g(\mathbf{X} - t)\, dt, \tag{4.3}$$

其离散形式为：

$$(f * g)(\mathbf{X}) = \sum_{t=0}^{N} f(t)\, g(\mathbf{X} - t). \tag{4.4}$$

对应神经网络中的卷积计算，$f(t)$ 表示被积函数，即输入向量；$g(\mathbf{X} - t)$ 表示卷积核函数，即卷积核。则神经网络中卷积的计算公式为：

$$S(i,j) = (I * K)(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(i-m, j-n) \cdot K(m,n), \tag{4.5}$$

其中：

- $S(i,j)$ 是卷积操作的输出结果的第 i 行第 j 列的值；
- I 表示输入数据的二维矩阵，$I(i,j)$ 是输入矩阵第 i 行第 j 列的值；
- K 表示卷积核的二维矩阵，$K(m,n)$ 是卷积核矩阵第 m 行第 n 列的值；
- M 和 N 分别是输入数据的行数与列数。

对于输入尺寸为 $C_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}$ 而输出尺寸为 $C_{\text{out}} \times H_{\text{out}} \times W_{\text{out}}$ 的卷积操作，有输入输出尺寸的关系为：

$$H_{\text{out}} = \lfloor \frac{H_{in} + 2 \times \text{padding}[0] - \text{kernel\_size}[0]}{\text{stride}[0]} + 1 \rfloor, \tag{4.6}$$

$$W_{\text{out}} = \lfloor \frac{W_{in} + 2 \times \text{padding}[1] - \text{kernel\_size}[1]}{\text{stride}[1]} + 1 \rfloor, \tag{4.7}$$

其中 padding、kernel_size、stride 分别为填充矩阵（输入是否用 0 进行填充）、卷积核矩阵和步长矩阵（卷积核在输入移动的步长）。

前向相关性分数传播方法在卷积层中可以通过两种方法进行扩展。第一种为对卷积进行全连接退化操作。卷积计算本质上是一种参数共享的全连接操作，通过对其进行退化，可以将卷积网络退化为等价的全连接网络。以 $1 \times 3 \times 3$ 的输入向量以及 $1 \times 1 \times 2 \times 2$ 的卷积核为例，其权重矩阵为：

$$W = \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \end{bmatrix}. \tag{4.8}$$

通过公式4.6和公式4.7可以算出输出尺寸为 $1\times1\times2\times2$，则可以把输入转为 1×4 的尺寸，并用一个全连接层代替卷积层，整个卷积计算可以被全连接有效等换，其权重参数 W_f 为：

$$W_f=\begin{bmatrix} w_{00} & 0 & 0 & 0 \\ w_{01} & w_{00} & 0 & 0 \\ 0 & w_{01} & 0 & 0 \\ w_{10} & 0 & w_{00} & 0 \\ w_{11} & w_{10} & w_{01} & w_{00} \\ 0 & w_{11} & 0 & w_{01} \\ 0 & 0 & w_{10} & 0 \\ 0 & 0 & w_{11} & w_{10} \\ 0 & 0 & 0 & w_{11} \end{bmatrix}. \tag{4.9}$$

更有一般的尺寸为 $C_{\text{in}}\times H_{\text{in}}\times W_{\text{in}}$ 的输入，以及尺寸为 $C_{\text{in}}\times C_{\text{out}}\times$kernel_size[0]$\times$kernel_size[1] 的卷积核，求得转换后的输出尺寸为 $C_{\text{out}}\times H_{\text{out}}\times W_{\text{out}}$，故卷积计算可以被替代为全连接计算，其权重参数的尺寸为 $\left(C_{\text{in}}\cdot H_{\text{in}}\cdot W_{\text{in}}\right)\times\left(C_{\text{out}}\cdot H_{\text{out}}\cdot W_{\text{out}}\right)$。如此，卷积神经网络可以转换为全连接神经网络进行前向相关性分数计算。

同时，考虑公式3.8与公式4.5可得：

$$V_{\text{out}}(i,j)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}V_{\text{in}}(i-m,j-n)\cdot K(m,n), \tag{4.10}$$

$V_{\text{out}}(i,j)$ 中的每个元素 $v_{\text{out}}(i,j)_{x,y}$ 也满足：

$$v_{\text{out}}(i,j)_{x,y}=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}v_{\text{in}}(i-m,j-n)_{x,y}\cdot K(m,n). \tag{4.11}$$

整个输入向量的相关性分数尺寸为 $C_{\text{in}}\times H_{\text{in}}\times W_{\text{in}}\times N$，其中 N 为原始输入尺寸总维度 $W_0\times H_0$。将整个输入向量的相关性分数转化为 $N\times C_{\text{in}}\times H_{\text{in}}\times W_{\text{in}}$，把 N 当作批量大小，与卷积核进行一般的卷积计算得到输出结果 $N\times H_{\text{out}}\times W_{\text{out}}$，再将结果翻转维度得到 $H_{\text{out}}\times W_{\text{out}}\times N$，即获取到每个输出对应的解释结果。通

过这种转换，可以高效地利用深度学习框架中的矩阵运算方法，避免了对卷积层进行退化以及再进行全连接矩阵计算。具体的计算过程如算法1所示，将所有的

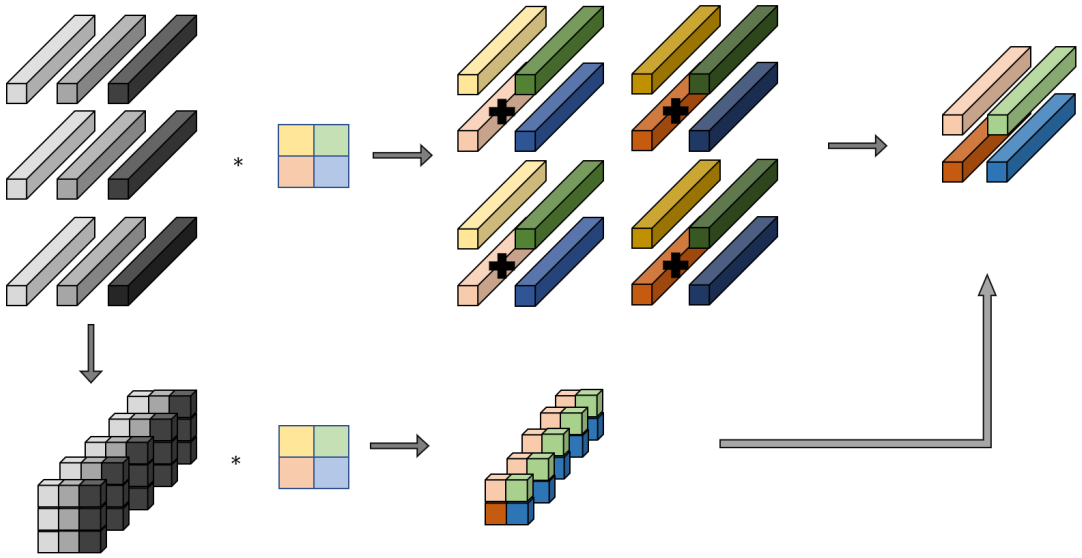


图 4.1 卷积层相关性分数计算通过卷积计算实现

相关性分数矩阵拼接在一起，并进行维度的翻转得到尺寸为 $N \times C_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}$ 的 V_{in}，算法再对 N 个 $H_{\text{in}} \times W_{\text{in}}$ 与卷积核 K_{conv} 进行正常的卷积操作，再将得到的 N 个 $H_{\text{out}} \times W_{\text{out}}$ 进行拼接并翻转维度，就可以得到 $H_{\text{out}} \times W_{\text{out}}$ 个维度为 N 的输出相关性分数。

算法 1 卷积层相关性分数计算算法

输入： 当前层所有的神经元节点的相关性分数矩阵 $V_{\text{in}}^{0}, V_{\text{in}}^{1}, \cdots, V_{\text{in}}^{W_{\text{in}} \times H_{\text{in}}}$，卷积核参数矩阵 K_{conv}

输出： 经过卷积层后输出节点的相关性分数矩阵 $V_{\text{out}}^{0}, V_{\text{out}}^{1}, \cdots, V_{\text{out}}^{W_{\text{out}} \times H_{\text{out}}}$

1: 所有输入节点的相关性分数矩阵拼接成 $V_{\text{in}} = cat\left(V_{\text{in}}^{0}, V_{\text{in}}^{1}, \cdots, V_{\text{in}}^{W_{\text{in}} \times H_{\text{in}}}\right)$

2: 将 V_{in} 进行转置，将维度 $\text{dim}_{\text{in}}^{0}, \text{dim}_{\text{in}}^{1}, \text{dim}_{\text{in}}^{2}$ 转置为 $\text{dim}_{\text{in}}^{2}, \text{dim}_{\text{in}}^{0}, \text{dim}_{\text{in}}^{1}$

3: **for** $i = 0$ **to** N **do**

4: $\quad V_{\text{out}}^{i} = V_{\text{in}}^{i} * K_{conv}$

5: **end for**

6: 将 V_{out} 进行转置，将维度 $\text{dim}_{\text{out}}^{0}, \text{dim}_{\text{out}}^{1}, \text{dim}_{\text{out}}^{2}$ 转置为 $\text{dim}_{\text{out}}^{1}, \text{dim}_{\text{out}}^{2}, \text{dim}_{\text{out}}^{0}$

4.2.3 循环神经网络特征归因算法

循环神经网络在处理序列数据的过程中，引入了隐状态，用于保存历史特征，并通过迭代与循环，在时空上建立对序列学习表示的模型。如图4.2所示，循

环神经网络有三个层次，输入层、隐藏层以及输出层，隐藏层中保存了隐状态用于记录历史特征，便于序列化学习上下文信息。

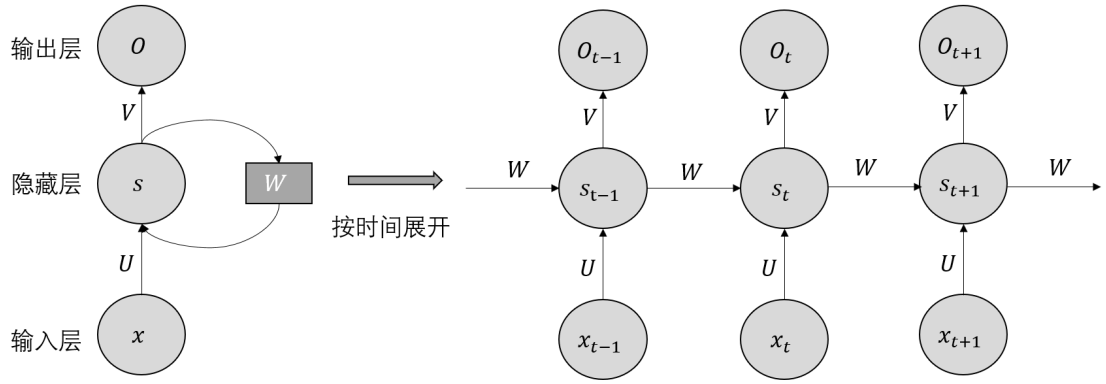


图 4.2 循环神经网络结构

公式4.12描述了循环神经网络输出的计算方式和隐藏状态的计算方式。其中 x_t 表示 t 时刻的输入，o_t 表示 t 时刻的输出，s_t 和 s_{t-1} 表示 t 时刻和 $t-1$ 时刻的隐藏状态，U、V 和 W 分别是输入层到隐藏层的权重矩阵、隐藏层到输出层的权重矩阵以及上一时刻隐藏状态的权重矩阵。

$$\begin{aligned} o_t &= g\left(V \times s_t\right), \\ s_t &= f\left(U \times x_t + W \times s_{t-1}\right). \end{aligned} \tag{4.12}$$

相比于卷积神经网络在单一时间步内的输入以及前向计算，循环神经网络会同时对多个时间步的输入进行处理，对于不同时间步的输入样本，其在循环神经网络中计算的次数并不相同，故循环神经网络对应的特征归因算法需要考虑输入样本之间不同时间步的区别。考虑到 V 在循环神经网络中代表权重矩阵，故下文中以 $Score_i^a$ 指代输入 i 对于输出 a 的显著性分数。

当 $t=0$ 时，得到 x_0 关于 o_0 的显著性分数为：

$$Score_{x_0}^{o_0} = g\left(V \times f\left(U \odot x_0\right)\right) \times \left|\operatorname{sgn}\left(o_0\right)\right|. \tag{4.13}$$

x_0 关于 s_0 的显著性分数为:

$$Score_{x_0}^{s_0} = f\left(U \odot x_0\right) \times \left|\operatorname{sgn}\left(s_0\right)\right|. \tag{4.14}$$

对于 x_t，以及任意的 k，$k \geqslant t$ 可以递推出 S_k 以及 O_k 的显著性分数为：

$$\begin{aligned}
&Score_{x_t}^{o_t} = g\left(V \times f\left(U \odot x_t\right)\right) \times |\mathrm{sgn}\left(o_t\right)|, \\
&Score_{x_t}^{s_t} = f\left(U \odot x_t\right) \times |\mathrm{sgn}\left(s_t\right)|, \\
&Score_{x_t}^{o_k} = g\left(V \times f\left(W \times Score_{x_t}^{s_k}\right)\right) \times |\mathrm{sgn}\left(o_t\right)|, \text{ where } t < k, \\
&Score_{x_t}^{s_k} = f\left(W \times Score_{x_t}^{s_{k-1}}\right) \times |\mathrm{sgn}\left(s_t\right)|, \text{ where } t < k.
\end{aligned} \tag{4.15}$$

从公式4.15中可知，对于同一输出结果，不同输入的样本将会考虑到不同时间步的影响，并且与循环神经网络一样，循环前向推理过程中即可获得对应的显著性分数。

4.3 实验与分析

4.3.1 实验设置

数据集

视觉任务上实验使用 CIFAR-10 作为视觉分类任务的数据集，并使用 Pascal VOC 2007 数据集[75]用于特征归因方法的评估。在循环神经网络上以文本分类任务为背景，使用 SST-2 数据集[76]作为文本分类的数据集。

PASCAL VOC 2007。PASCAL 全称 The Pattern Analysis, Statical Modeling and Computational Learning，是一个由欧盟资助的计算机技术委员会，其举办的 PASCAL VOC 挑战赛是一个世界级的计算机视觉竞赛。竞赛中使用的数据集常常被作为视觉对象的分类识别和检测的一个基准测试，提供了检测算法和学习性能的标准图像注释数据集和标准的评估系统。该比赛使用的数据集每年会根据挑战赛进行变动，其中比较出名的有 VOC2007 以及 VOC2012。实验选择 VOC2007 数据集，该数据集由 train/val/test 三部分组成，包含 9 963 张标注过的图片，共有 20 个类别，共标注出 24 640 个物体。目前只有 VOC2007 的测试集数据标签被公布。本次采用的数据集来自 VOC2007 测试集中随机抽选且只具有一个目标框的图像共 1000 张。

SST-2。SST-2（The Stanford Sentiment Treebank 2）是一个小型的单句分类任务，其中的内容来自电影评论中的句子以及对应的情感标注。该数据集中训练

集包含 67 350 个样本以及对应的句子粒度的标签，测试集则包含 1 821 个样本以及对应标签。该数据集中以单句为样本基准，句子长度长短不一。

模型

视觉任务上实验继续使用在 CIFAR-10 上进行预训练的 AlexNet 模型对算法进行评估，同时使用在 ImageNet 数据集上进行预训练的 VGG-16[77]用于显著性图的可视化展示。VGG-16 是 VGG 系列深度卷积网络结构的预训练网络，VGGNet 网络在 2014 年 ILSVRC 分类任务上赢得了冠军。VGG-16 整个网络都使用尺寸为 3×3 的卷积核以及尺寸为 2×2 的最大池化层，其中含可优化参数的层次共有 16 层。

文本分类任务实验选择以 RNN 神经元为基础搭建的循环神经网络，使用预训练的 GloVe6B-50d[78]作为词嵌入预训练模型，隐藏层维度为 64。该网络权重参数由 Kaiming 初始化方法[79]进行初始化，并在 SST-2 上训练 20 个 epoch，其中使用随机梯度下降方法进行优化，学习率为 0.01，动量值为 0.9，均为经验设置。该神经网络在 SST-2 验证集的准确率为 63%。

循环神经网络所使用的词嵌入模型为 GloVe，是一种将词汇映射到连续向量空间的网络模型，旨在将单词的语义信息以向量形式进行表示。输入的文本经过预处理并进行词嵌入得到对应的词向量，便于作为神经网络的输入。GloVe 模型通过建模词向量与共现矩阵的近似线性关系，利用全局的词共现统计信息，对单个单词进行词表征。本次实验使用的 GloVe6B-50d 是 GloVe 族模型中以 60 亿个单词为语料库训练的、词向量维度为 50 的词嵌入模型。

评估指标

本次实验对特征归因方法采用的评估方法为敏感度分数以及定位能力评估，并且为避免引入不必要的评估偏见以及主观性性，本次实验未采用基于视觉的人工评估。

敏感度分数[80]是一种居于显著性图的定量评估指标，旨在通过观察加入干扰对输出分数的影响以考察显著性图的显著程度。其具体指标由平均下降指标与平均上升指标组成，其中平均上升指标是通过显著性图对原样本进行正向增强，观察模型对目标类的输出分数上升比例；评价下降指标是通过显著性图对原样本进行遮掩或削弱，观察模型对目标类的输出分数下降百分比。其中平均

下降指标表达式为：

$$\sum_{i=1}^{N} \frac{\mathrm{Max}\left(0, Y_i^c - O_i^c\right)}{Y_i^c}, \tag{4.16}$$

平均上升指标表达式为：

$$\sum_{i=1}^{N} \frac{\mathrm{Sign}\left(Y_i^c < O_i^c\right)}{N}, \tag{4.17}$$

其中 Y_i^c 表示原始输入样本 i 对于类别 c 的预测分数；O_i^c 则是基于显著性图对原始输入样本 i 进行修改后的样本对于类别 c 的预测分数；Sign(x) 表示当 x 为真时输出 1，否则输出 0。敏感度分数中对原始样本进行的修改可能会破坏原有的上下文关系，导致修改后的样本对整体类别的预测评估可能会降低，故此处不采用分类任务的准确率进行评估。本次实验中采用归一化增强/削弱方法对原始样本处理。特别地，对于原始输入样本 i 以及显著性图 V_i^c，首先对显著性图进行负值过滤，确保没有负值，再进行 Max-Min 归一化，得到对应的修改权重，增强的方法为：

$$X'_{+} = X + X * \left(V_i^c\right)_{\mathrm{norm}}, \tag{4.18}$$

削弱的方法为：

$$X'_{-} = X * \left(V_i^c\right)_{\mathrm{norm}}, \tag{4.19}$$

其中：

$$\left(V_i^c\right)_{\mathrm{norm}} = \frac{V_i^c - \left(V_i^c\right)_{\mathrm{Min}}}{\left(V_i^c\right)_{\mathrm{Max}} - \left(V_i^c\right)_{\mathrm{Min}}}. \tag{4.20}$$

根据显著图中不同特征的显著性强度，特征会得到不同的修改效果。在增强方法中，显著性最强特征会变为之前的两倍强度，显著性最弱的特征则保持不变；在削弱方法中，显著性不强的特征会被削弱，显著性最强的特征则保持不变。

整个评估流程如算法2所示，对于数据集中的样本以及对应的显著性图，首先对显著性图进行负值过滤以及 Max-Min 归一化，并和原始输入样本结合得到增强和削弱的修改样本，再计算出所有样本的网络输出分数，最后根据得到的网络输出分数求解敏感度分数评估指标。

本次实验同样采用定位能力评估对显著性图进行评估。常见的定位能力评估有指向游戏[81]（Pointing Game）和基于能量的指向游戏[82]，两者都是基于对象定位的解释评估方法，用于视觉任务显著性图的评估。定位能力评估方法希

算法 2 显著性图敏感分数计算方法

输入: 网络模型 Net、数据集以及样本 Dataset $= \{X_1, X_2, \cdots, X_N\}$，以及每个样本对应的显著性图 $\{V_1, V_2, \cdots, V_N\}$

输出: 显著性图对应的敏感性分数平均下降 S_{Drop} 和平均上升 $S_{Increase}$

1: **for** $i = 1$ **to** N **do**
2: $V_i \leftarrow \mathrm{Max}\left(V_i, 0\right)$
3: $\left(V_i\right)_{\mathrm{norm}} = \frac{V_i - \left(V_i\right)_{\mathrm{Min}}}{\left(V_i\right)_{\mathrm{Max}} - \left(V_i\right)_{\mathrm{Min}}}$
4: $X_i{}'_{+} \leftarrow X_i + X_i * \left(V_i\right)_{\mathrm{norm}}$
5: $X_i{}'_{-} \leftarrow X_i * \left(V_i\right)_{\mathrm{norm}}$
6: $\left(O_i\right)_{+} \leftarrow Net\left(X_i{}'_{+}\right)$
7: $\left(O_i\right)_{-} \leftarrow Net\left(X_i{}'_{-}\right)$
8: $\left(Y_i\right) \leftarrow Net\left(X_i\right)$
9: **end for**
10: $S_{Drop} \leftarrow \sum_{i=1}^{N} \frac{\mathrm{Max}\left(0, Y_i - \left(O_i\right)_{-}\right)}{Y_i}$
11: $S_{Increase} \leftarrow \sum_{i=1}^{N} \frac{\mathrm{Sign}\left(Y_i < \left(O_i\right)_{+}\right)}{N}$

望通过显著性图的定位能力来衡量生成显著性图的质量。借鉴图像检测任务以及检测数据集，指向游戏通过判断显著图中的最大点是否落入目标检测框中来衡量显著图所显示的定位能力。基于能量的指向游戏不关心显著图中的最大点而是关注显著图中有多少能量落入目标对象边界框中。具体来说，基于能量的指向游戏依据目标类别的边界框将输入图像进行二值化，边界框中区域分配为 1，边界框外分配为 0，再将二值化的图与显著性图进行逐点乘法并求和以获得目标边界框中的能量。具体的，其计算方式为:

$$S_{energy} = \frac{\sum V^c_{(i,j)\in bbox}}{\sum V^c_{(i,j)\in bbox} + \sum V^c_{(i,j)\notin bbox}}. \tag{4.21}$$

基于能量的指向游戏克服了由噪声带来的最大点偏移问题，故本次实验采用基于能量的指向游戏对显著性图进行定位能力评估。

基准

为了进行评估对比实验，选择了三种特征归因方法作为基准，进行对比实验。

- Mask。Mask 是一种基于扰动的黑盒模型特征归因方法，通过对输入施加扰动得到输出的分数变化，并以此作为扰动区域的显著性图。
- GradCAM。GradCAM 是一种基于梯度的特征归因方法，用基于梯度的方

法替代类激活映射中的全局平均化权重，并与特征图进行融合得到显著性图。

- GradCAM++[83]。GradCAM++ 是 GradCAM 的优化版本，通过像素级别的加权求和得到了更细粒度的显著性图。

4.3.2 实验数据结果

本节通过对比本章提出的特征归因算法与其他可解释方法在敏感性分数以及定位能力评估上的指标，验证算法的有效性。

表4.1给出不同方法对 AlexNet 的显著性图在敏感度分数上的对比实验结果。其中平均下降指标越低表明显著性图对物体的定位能力越强，平均上升指标越高表明显著性图对物体的识别能力越强。平均下降指标揭示了通过削弱图像中非显著区域的特征，样本在网络中的得分降低的程度。本文的方法在比较的四种方法中降低幅度最小，这表明即使只依赖显著区内的样本，也足以使网络输出与原始样本的得分相近。平均上升指标则反映了增强显著区内样本对网络类别得分输出的提升效果。本文的方法在这四种方法中效果最显著，说明显著区内的特征与输出得分之间存在较强的联系，强化这些区域内的特征可以显著提高与类别相关的输出得分。结合两个指标说明本文的方法生成的显著性图能有效捕捉与类别输出分数相关的特征区域，体现了算法的有效性。

表 4.1 敏感度分数评估结果

方法	平均下降 (%)	平均上升 (%)
Mask	86.9	54.9
GradCAM	76.7	56.7
GradCAM++	86.3	47.7
FAFRP(Ours)	**67.9**	**62.3**

敏感度分数通过指定目标类别来计算下降或上升的比例，只考虑对目标类别的敏感程度。为考虑显著性图对所有类别输出的影响，可以将敏感度分数计算中的输出分数 Y_i 和 O_i 换为经过 Softmax 后的置信度，即 $\text{Softmax}(Y)_i$ 和 $\text{Softmax}(O)_i$。此时敏感度分数（Softmax 后）不仅仅只关注显著性图对指定类别的影响，而是考虑对所有输出结果的影响。此时平均下降指标观察模型对目标

类的置信度下降百分比，计算公式为:

$$\sum_{i=1}^{N} \frac{\text{Max}\left(0, \text{Softmax}(Y)_i^c - \text{Softmax}(O)_i^c\right)}{\text{Softmax}(Y)_i^c}, \tag{4.22}$$

平均上升指标观察模型对目标类的置信度上升比例，计算方式为:

$$\sum_{i=1}^{N} \frac{\text{Sign}\left(\text{Softmax}(Y)_i^c < \text{Softmax}(O)_i^c\right)}{N}. \tag{4.23}$$

表4.2给出了不同方法在 CIFAR-10 上的敏感度分数（Softmax 后）。相比于其他三种特征归因方法，FAFRP 在平均下降指标和平均上升指标都取得了最优的性能。这表明，经过 FAFRP 生成的显著性图与目标类别联系更为紧密，其显著区域对其他类别的影响更小。

表 4.2 敏感度分数（Softmax 后）评估结果

方法	平均下降 (%)	平均上升 (%)
Mask	80.3	38.4
GradCAM	68.3	35.1
GradCAM++	75.2	32.7
FAFRP(Ours)	**67.5**	**48.1**

表4.3给出不同方法对 AlexNet 的显著性图在基于能量的指向游戏上的对比实验结果。较高的指标分数意味着生成的显著性图在更大程度上突出了原始图像中与目标物体相关的区域，这一结果验证了本文提出方法在目标物体定位功能上的有效性。相较于其他三种特征归因方法，本文的方法展现出最为显著的效果，其生成的显著性图大多数情况下准确覆盖了与目标类别紧密相关的原始输入图像区域。这表明，本文的方法在定位与目标类别相关的输入特征方面具备出色的性能。

表 4.3 定位能力评估结果

	Mask	GradCAM	GradCAM++	FAFRP(Ours)
$S_{energy}(\%)$	56.1	48.1	49.3	**57.2**

图4.3展示了 VGG16 在 VOC2007 测试集上的显著性图可视化效果，其中红色方框为物体的真实位置。可以看到，LRPFT 生成的显著图大多数能量是落入

物体的真实定位框中。

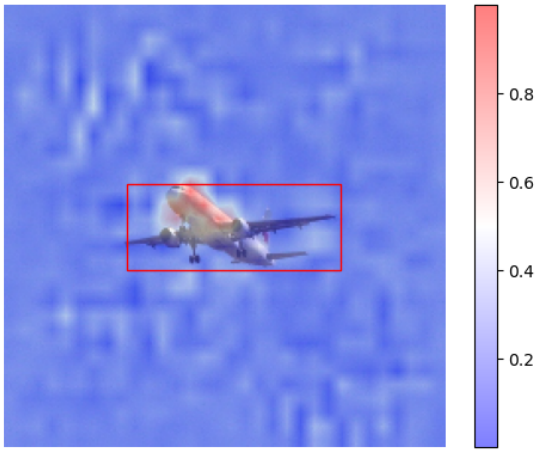
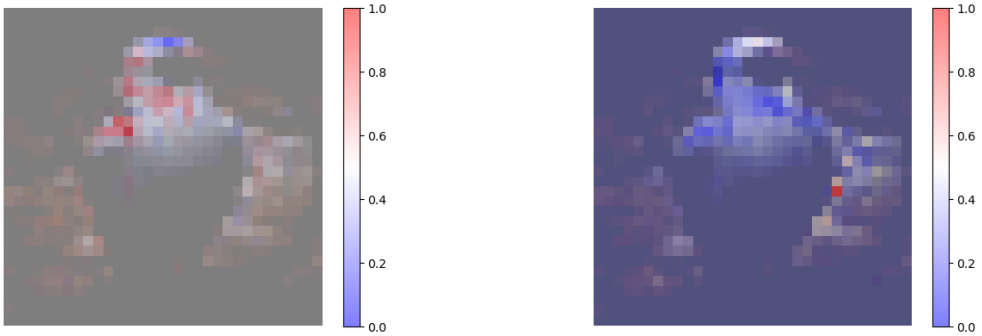


图 4.3 VGG16 在 VOC2007 测试集上的物体定位效果

4.3.3 显著性图可视化效果

图4.4展示了输入图像在不同输出分数上的显著性图，所有的显著性图都是LRPFT 生成的结果与原始图像叠加而来，旁边的图例表示了显著性程度。其中图4.4(a)表示输入图像关于鸟类别的显著性图（鸟类输出概率为 39%），可以看到显著性图中突出了鸟类的颈部和周围环境，这表示模型依据环境与物体头部判断其类别为鸟类。其中图4.4(b)表示输入图像关于猫类别的显著性图（猫类输出概率不足 1%），整个图像对于猫类别判断基本为负面贡献。

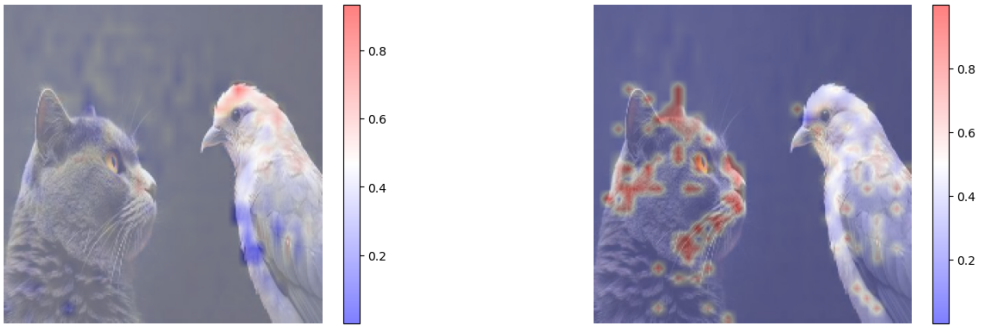


(a) 输入图像关于鸟类别的显著性图像 **(b) 输入图像关于猫类别的显著性图像**

图 4.4 AlexNet 在 CIFAR-10 上的显著性图可视化效果

图4.5展示了 VGG16 在含有不同物体的输入图像上的显著性图。其中图4.5(a)表示输入图像关于鸟类别的显著性图，即使鸟类输出概率不足 5%，但显著性图仍

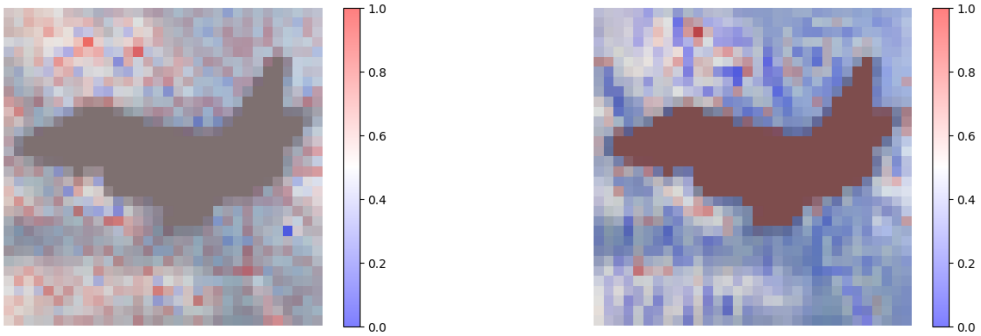
然突出了鸟类的头部，表明这部分区域对鸟类分数有较高贡献。图4.5(b)表示输入图像关于猫类别的显著性图（猫类输出概率约为 35%），显著性图表明 VGG16 关注到了猫的头部轮廓以及面部器官，正确地揭示了模型的决策依据。



(a) 输入关于鸟类别的显著性图像　　(b) 输入关于猫类别的显著性图像

图 4.5　VGG16 在含有不同类别输入图像上的显著性图可视化效果

图4.6展示了同一网络在初始化阶段和训练收敛阶段对同一输入图像的显著性图对比。图4.6(a)是网络进行随机初始化时对输入图像生成的显著性图，此时网络对输入图像输出飞机类别的概率为 9.8%；图4.6(b)是网络经过训练后对输入图像生成的显著性图，此时网络对输入图像输出飞机类别的概率为 99.9%。随着分类精度的提升，显著性图会变得更加集中，并主要分布在物体真是位置附近。



(a) 随机初始化的网络生成的显著性图　　(b) 预训练网络生成的显著性图

图 4.6　随机权重的初始网络与预训练网络在同一输入图像上的显著性图对比

图4.7展示不同特征归因方法在同一输入图像上的显著性图，与 GradCAM 和 GradCAM++ 相比，本文的方法生成的显著性图定位更准确；与 Mask 相比，本文的方法生成的显著性图噪声更少。

图 4.7 不同特征归因方法在同一输入图像上的显著性图对比

同样地，将本文的方法推广至文本分类中情感分类任务，对于输入语句“This movie is not bad and quite enjoyable.”，网络得到正面情绪概率为 98% 的分类结果。本文的方法会为每个单词生成长度为 50 的显著性特征向量（与词向量化后的输入向量长度一致），对其进行求和得到每个单词对应的显著性分数。图4.8展示了对文本分类任务的显著性图可视化效果，可以看到单词“bad”和“quite”对网络输出正面情绪结果作出负面贡献，而单词“enjoyable”对网络输出作出最大正面贡献。

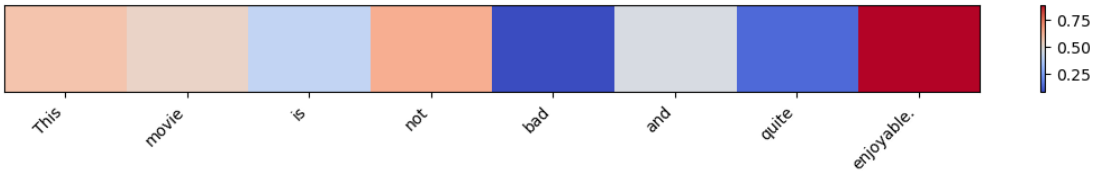


图 4.8 FAFRP 在情感分类任务上的显著性图可视化

4.4 本章小结

在本章节里，首先对当前特征归因方法进行了描述与总结。结合当下主流特征归因方法，本文将前向相关性分数传播方法扩展至特征归因领域，并推导了卷积神经网络以及循环神经网络的特征归因算法。通过实验对比，本文的方法在生成显著性图的过程中既使显著区域对目标类别输出分数保持高度敏感，又能使显著性图具备对目标物体的定位能力，验证了本文方法的有效性以及解释性。

第五章　神经网络可解释性系统

前文介绍了基于前向传播的可解释算法，本章将算法投入实际应用中，设计开发神经网络可解释性系统，使得用户可以上传自己的网络与数据，并对其进行解释与分析，以便用户更好地了解神经网络内部原理。本章首先对研发背景进行阐述，然后描述系统设计与架构，最后展示系统的最终效果。

5.1　系统研发背景

神经网络作为高效的机器学习模型方法，在许多领域取得了令人瞩目的成就，包括图像识别、语音识别和自然语言处理等。然而，随着神经网络广泛的应用，人们对神经网络的决策过程和工作原理提出了更多的疑问和需求。用户可能会对自己的数据被用于训练神经网络感到担忧，并希望能够了解模型对其数据的处理方式和决策逻辑。提供解释性系统可以增强用户对数据隐私和模型使用的接受度，并促进用户对模型训练的参与和反馈。同时，研究人员对神经网络的工作原理和决策过程的理解是推动学术研究和技术进步的关键，解释性系统可以帮助研究人员更好地理解神经网络的内部机制，发现模型的局限性和改进空间，从而推动神经网络领域的发展。

基于这些需求，本章设计一套给用户提供上传网络与数据，并且使用可解释性算法对网络与数据进行可视化分析与解释的系统。

5.2　需求分析

神经网络可解释性的目的是方便用户使用并对自己的数据与网络进行分析和解释。从以下几个角度进行需求分析。

5.2.1 用户需求分析

对于系统使用的主体，其需求有：

- 不同用户可能对解释性系统的需求有所不同，希望系统能够提供灵活的定制化功能，以满足自身的需求，并且系统需要提供自定义的解释方法。
- 用户希望能够理解神经网络模型的决策过程，包括网络对输入数据的处理方式、特征的提取方式以及最终的预测结果。用户希望系统能够提供直观的解释，帮助他们理解模型是如何做出决策的。
- 用户希望系统能够保护其数据的隐私，不泄露敏感信息。在解释模型决策的过程中，系统需要确保用户的个人数据不被泄露或滥用，同时也需要对数据进行合法和安全的处理。
- 系统需要设计成用户友好的界面，使用户能够方便地访问解释结果，并与系统进行交互。这包括设计直观的用户界面、提供清晰的解释文本和图形，并允许用户根据自己的需求进行定制。

5.2.2 功能需求分析

该系统为实现神经网络可解释算法功能，需要至少以下功能：

- 对神经网络的解释与分析。对模型结构的分析，包括网络层次、神经元数量和连接方式等；对网络参数的的解释，包括权重、偏置的含义和作用等；对模型决策过程的解释，包括输入特征的处理、隐藏层的激活和输出结果的生成。
- 对特征重要性的分析。分析输入特征对输出结果的影响程度，评估特征的重要性。
- 数据隐私的保护。确保用户的数据隐私得到保护，不泄露敏感信息。
- 用户交互和定制。提供用户友好的界面，支持用户对解释结果进行交互和定制，并允许用户根据需求自定义解释功能和显示方式。

除了以上功能需求外，系统还应考虑到其拓展性和可维护性。为了实现这一点，系统应该采用模块化的设计，将不同的功能划分为独立的模块，使其易于扩展、替换或升级。这种模块化设计可以提高系统的灵活性和可维护性，使其能够

适应不断变化的需求和技术发展。同时，模块化设计还能够降低系统的耦合度，减少对整体系统的影响，从而更容易进行单独模块的测试、调试和优化。通过模块化设计，系统可以更好地应对未来的需求变化和技术挑战，保持其长期可持续发展的能力。

5.3 系统设计

整体架构如图5.1所示，采用前后端分离架构，前端与后端分离并互相独立，之间通过 API 进行通信。其中前端负责用户界面与交互，后端负责处理业务逻辑、进行计算并处理保存数据。

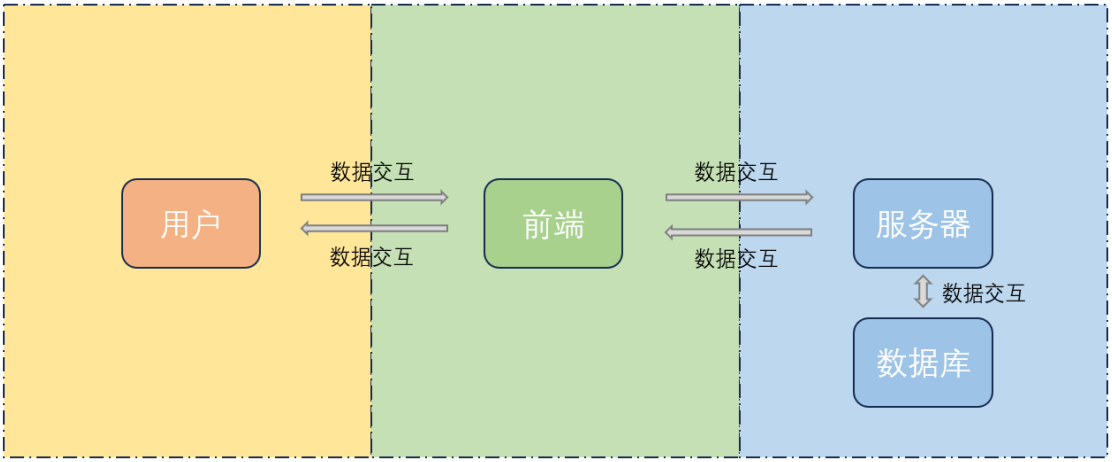


图 5.1 整体系统架构

前端的架构如图5.2，为了保证系统的性能和稳定性，前端采用了反向代理服务器来实现负载均衡。通过反向代理服务器，系统可以有效地将浏览器的请求分发到多个前端服务器上，从而分担服务器的负载并提高系统的响应能力。这种架构不仅可以提高系统的整体性能，还可以增强系统的可靠性和容错能力，确保用户能够获得稳定和高效的服务体验。

后端的架构如图5.3所示。其中请求管理器负责接收来自前端的请求，并将请求转发给相应的服务器进行处理。模型管理器则负责管理后端系统中的各种模型，包括系统提供给用户的实例模型以及用户自行上传的模型。模型管理器负责模型的加载、更新、存储和释放等操作，以确保系统能够有效地使用模型来进行数据分析和预测。任务管理器是后端系统的核心，负责管理后端系统中的所有任务的调度、执行、监控和报告等操作。数据管理器是与数据库的接口，负

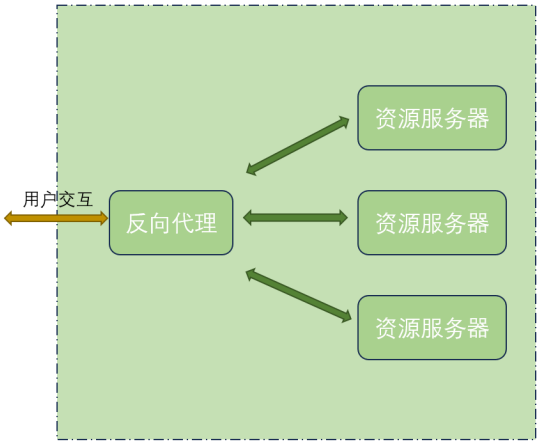


图 5.2 前端架构

责管理后端系统中的各种数据，包括模型权重、数据集图像等。数据管理器负责数据的存储、检索、更新、删除和备份等操作，以确保系统能够有效地使用数据来支持业务需求。

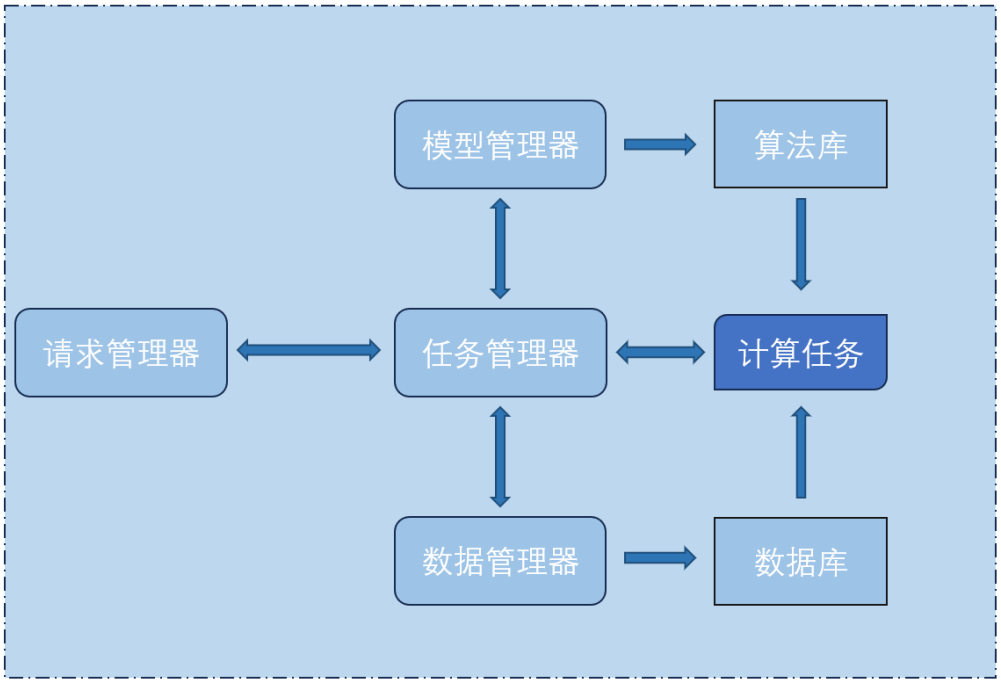


图 5.3 后端架构

5.4 系统实现

基于前文中的系统架构设计，本节对神经网络可解释性系统进行了具体实现，本节将从后端功能、前端交互两个方面进行介绍。

5.4.1 系统开发环境

首先对系统开发环境进行简要说明。本系统前端使用 Nginx 作为反向代理服务器，Nginx 是一款高性能的开源的 Web 服务器，也可以用作反向代理服务器、负载均衡器，将来自浏览器的请求转发到多个前端服务器上，并且可以根据请求的负载情况进行动态负载均衡。使用 Nginx 可以快速地向浏览器提供静态文件服务，包括 HTML、CSS、JavaScript、图像和视频等静态资源，并且可以配置 HTTP 缓存来加速内容传输和提高用户体验。前端页面使用 JavaScript+CSS 进行开发，使用 JavaScript 实现网页的动态交互和行为逻辑，使用 CSS 实现网页的外观设计和样式美化。

后端使用 Python 语言进行开发。后端使用开源的 Flask 框架作为请求管理器，负责网络请求的接收以及路由 API 的交接，并使用 HTTP 协议与前端进行数据交互，Flask 是一个功能强大、灵活易用的 Python Web 应用框架，适用于各种类型的 Web 应用和项目开发。后端服务器深度学习算法部分使用 PyTorch 深度学习框架，PyTorch 架构提供了适用于深度学习的动态计算图、自动求导等功能，封装了一系列常见模型与函数功能，非常便于进行深度学习应用与开发。后端服务器数据部分选择使用 MySQL 数据库，MySQL 是常见的关系数据库，在此使用 MySQL 保存系统中关键数据如图像、解释结果等。

5.4.2 后端功能

后端主要实现三大功能：

1. 接收并实例化来自用户的网络以及权重；
2. 接收用户选择的输入样本以及解释方法，进行算法计算；
3. 根据用户需求，将算法结果返回至前端。

对于用户上传的网络模型与权重，后端会交至模型管理器，模型管理器负责对网络进行实例化，并在其生命周期内进行维护。模型的具体可解释算法需要用户自行选择输入样本，后端接收到样本之后调用算法库中的神经网络可解释算法，并由任务管理器建立可解释性算法计算任务，任务完成后结果与相关信息存入数据库中。当后端接收到用户的解释结果请求时，通过数据管理器访问

数据库，取出对应的解释结果，返回给前端。一次完整的交互解释所对应的时序图如图5.4所示。

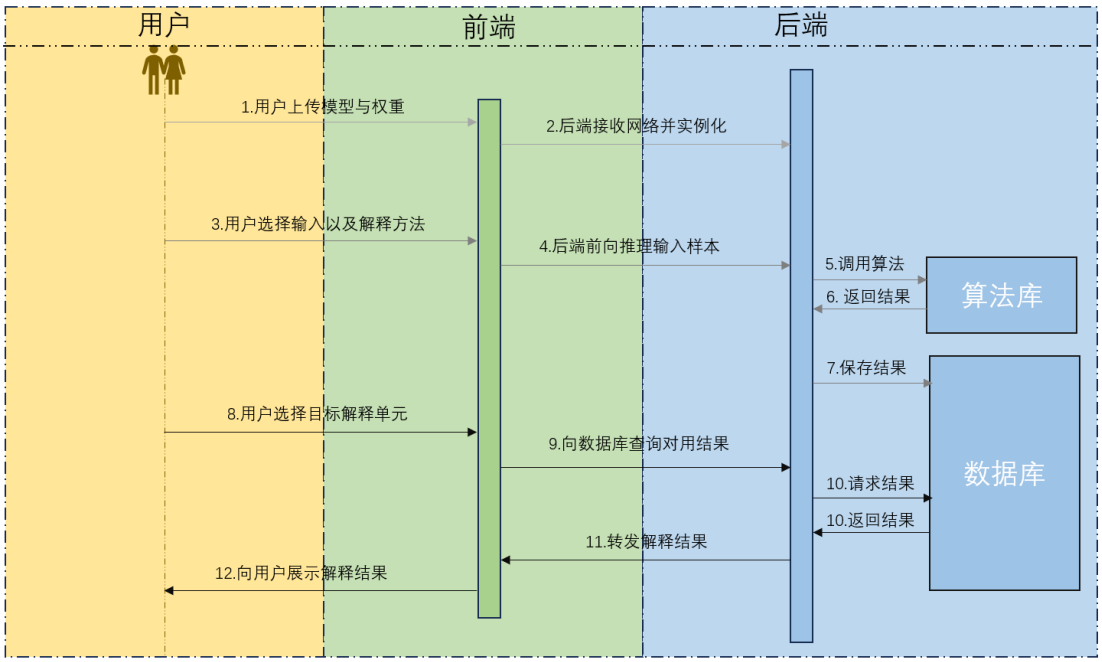


图 5.4 用户完整使用 UML 时序图

5.4.3 前端交互

前端是与用户进行交互的核心模块，前端页面负责接收用户的命令与操作，并将合适的请求发送给后端服务器，进行任务请求与资源请求，同时接收来自后端的算法结果，向用户进行展示。

用户引导以及使用说明。为方便用户更快地熟悉和了解系统的功能和操作流程，降低学习成本，交互页面提供了较为简洁的用户引导与使用说明，如图5.5中的左图所示。

上传网络以及权重操作。为用户提供上传网络以及权重的接口，便于用户对自己的网络以及数据进行解释，图5.5中的右图展示了上传接口。

网络结构以及特征变量展示。对于用户上传的网络以及输入数据，页面会渲染出网络的结构以及参数，并对神经元激活情况进行标注，同时标注输出类别。并且用户可以点击查看每个单独通道的特征向量，图5.6展示了前端页面所渲染的网络结构以及参数，并对第 1 层的第 2 个通道的特征向量进行了查看，其中每个点代表了特征向量中的一个神经元，并对激活的神经元进行了染色处理。右

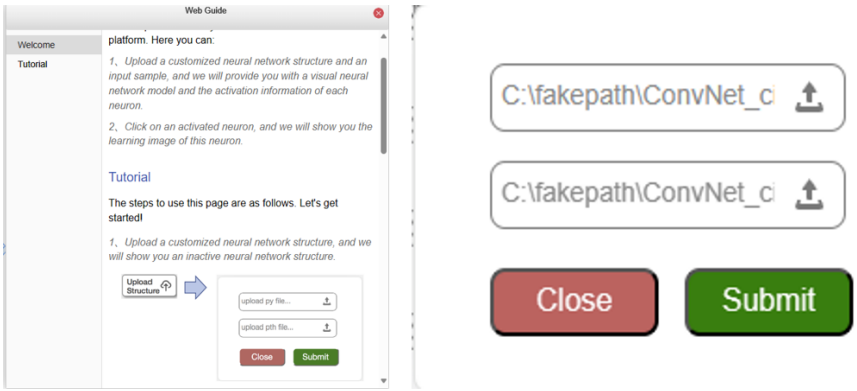


图 5.5 用户引导与上传接口

方标注了输出神经元中的最大值，即分类类别。

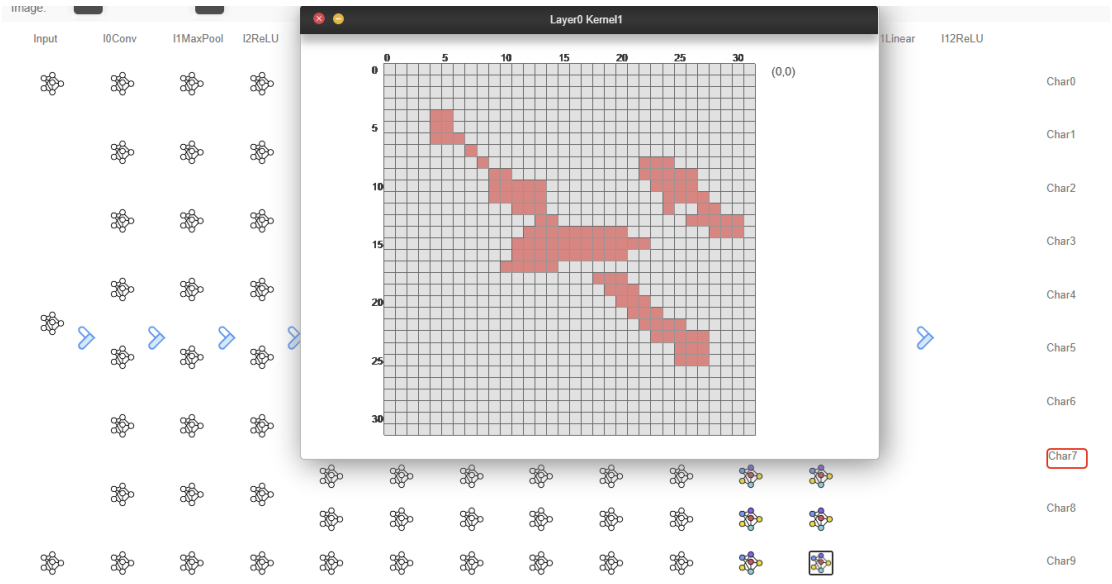


图 5.6 查看网络结构以及特征向量

输入样本选择与上传。系统提供了示例图片，用户也可以自行选择上传自己的图片，如图5.7所示。

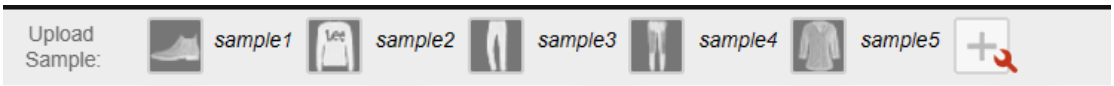


图 5.7 选择输入样本以及上传接口

解释结果展示。用户点击目标解释单元，即可向后端发起解释结果请求，通过点击小图，可将解释结果进行放大展示，如图5.8所示。用户可以通过放大缩小按钮对解释结果进行对应操作。

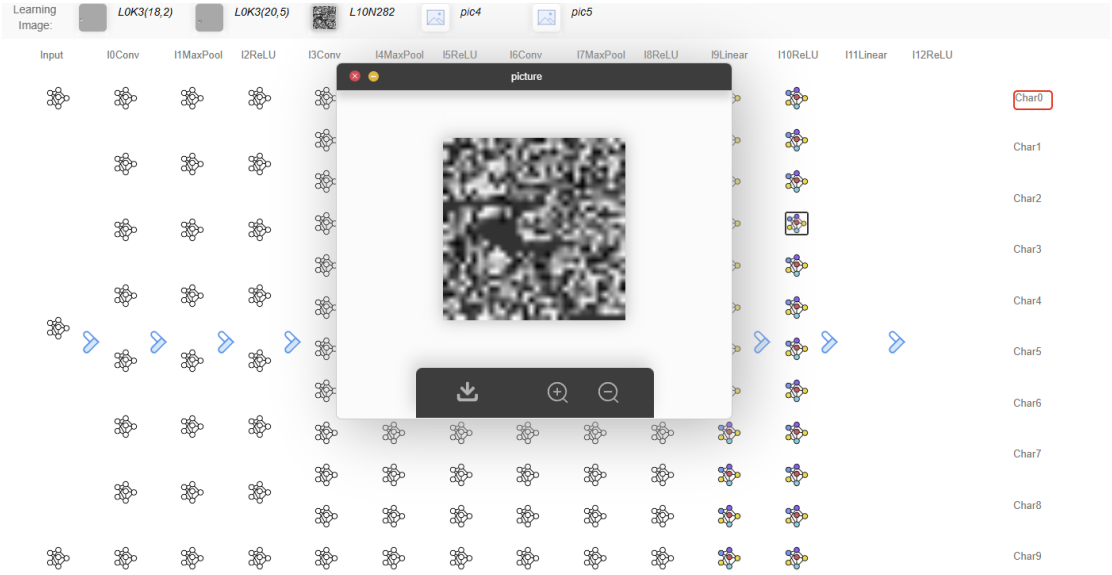


图 5.8 解释结果查看

整个前端的页面如图5.9所示。包含了中间的展示部分、右上方的上传接口、上方的输入样本选取与解释结果选取栏。

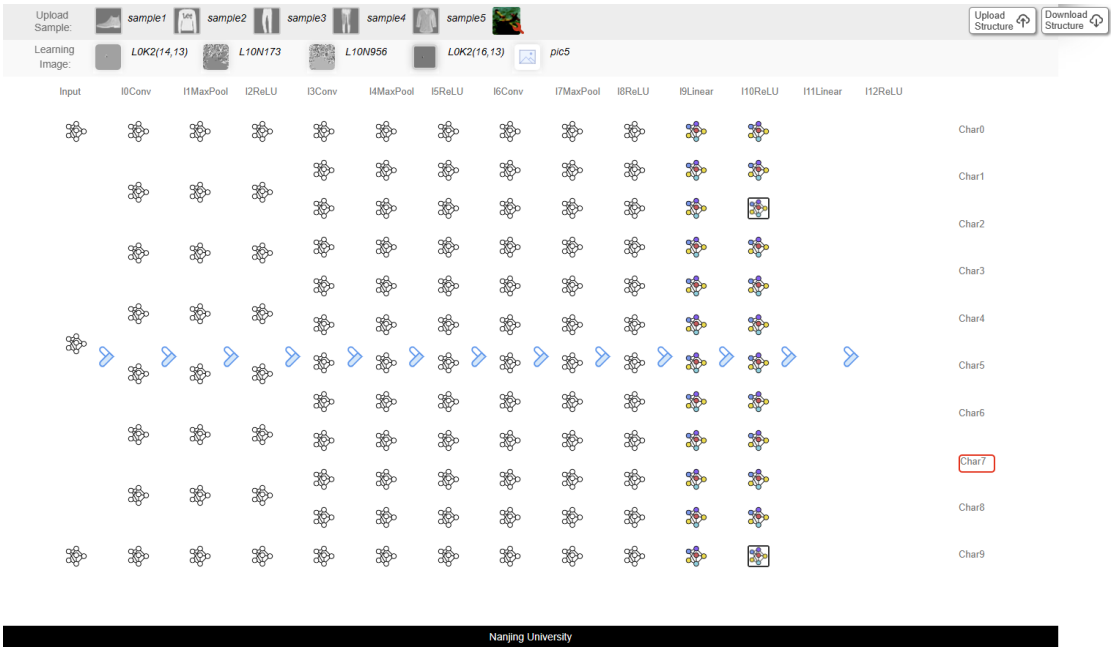


图 5.9 前端整体页面

5.5 本章小结

本章介绍了搭建的神经网络可解释系统。该系统为用户提供了自行上传网络与权重，选择输入样本并进行可解释分析的功能。本章首先对系统的背景以及需求进行了介绍，并分别介绍了前后端模块架构以及功能，最后展示了神经网络可解释系统的效果展示，给出了对应的截图。本文提出的算法被集成至神经网络可解释系统中，从侧面说明了算法的实用性以及有效性。

第六章　总结与展望

神经网络作为高效的机器学习模型，在日常生活中扮演着越来越重要的角色，从简单的分类任务到复杂的生活场景，特别是关乎安全与健康的领域，神经网络无不发挥着重要的作用。同时，对神经网络的理解也须跟随上应用的脚步。近年来不少工作开始对神经网络内部工作原理以及决策机制发起探索，借助机器学习可解释性并根据神经网络结构设计特有的可解释方法，对神经网络内部特征和决策过程进行解释与分析。本文根据当前主流的可解释性范式，提出一种基于前向传播的可解释方法，用于对神经网络进行特征可视化，在考虑神经网络结构的情况下，生成了稳定的可视化图像。此外，将该方法延展至特征归因方法，生成输入样本对于输出结果的显著性图。本文的主要研究内容与贡献如下：

1. 本文根据神经网络内部原理，结合神经网络的结构特点，提出了一种基于前向传播的特征可视化可解释方法，名为前向相关性分数传播方法。本方法通过固定相关性分数矩阵尺寸，并借鉴神经网络前向传播原理，设计相关性分数的前向传播规则，进而求得所有神经元对于输入样本的特征可视化解释结果。以视觉分类任务为基础，将可视化结果在通道级别上进行融合，得到了一系列具有语义信息的解释结果。最后，通过设计剪枝与聚类等实验，从侧面验证了该方法的有效性。
2. 本文进一步将基于前向传播的解释方法延展至特征归因方向，并分别得到卷积神经网络和循环神经网络上的特征归因方法，通过前向传播得到任意输入关于输出结果的显著性图，并在不同的任务上进行了可视化实验。本方法具有灵活性和通用性，可以用在任意的神经网络中进行特征归因分析。
3. 本文将所提出的可解释性算法集成，搭建了一个通用便利的神经网络可解释系统。该系统为用户提供对神经网络的自主分析功能，便于用户对自己的网络模型以及输入样本进行可解释性分析，达到进一步满足用户了解网络原理以及决策依据的需求，验证了本文提出的算法的实际应用价值。

在本文提出的方法基础上，可以沿着不同的方向继续进一步研究。对于特征可视化方法，可以进一步对具有语义的解释结果进行总结与分析，挖掘神经网络对语义信息的学习过程以及提取过程，揭示神经网络的决策过程。其次，对于语义信息需要进一步设计相关评估指标而非人工评估，如此可以增加解释结果分析工作的效率与实用性。对于特征归因方法，可以进一步将其扩展至其他类型的神经网络，如图神经网络、编码器-解码器架构等，在其他任务上进行特征归因分析。

参考文献

[1] ROSENBLATT F. The perceptron: a probabilistic model for information storage and organization in the brain.[J]. Psychological Review, 1958, 65(6): 386.

[2] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533-536.

[3] HE K, GIRSHICK R, DOLLÁR P. Rethinking imagenet pre-training[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 4918-4927.

[4] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25.

[5] RADFORD A, NARASIMHAN K, SALIMANS T, et al. Improving language understanding by generative pre-training[J]., 2018.

[6] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J]. OpenAI blog, 2019, 1(8): 9.

[7] BROWN T, MANN B, RYDER N, et al. Language models are few-shot learners[J]. Advances in Neural Information Processing Systems, 2020, 33: 1877-1901.

[8] ACHIAM J, ADLER S, AGARWAL S, et al. Gpt-4 technical report[J]. ArXiv preprint arXiv:2303.08774, 2023.

[9] KIRAN B R, SOBH I, TALPAERT V, et al. Deep reinforcement learning for autonomous driving: A survey[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(6): 4909-4926.

[10] HU Y, YANG J, CHEN L, et al. Planning-oriented autonomous driving[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 17853-17862.

[11] KOMATSU M, SAKAI A, KOMATSU R, et al. Detection of cardiac structural abnormalities in fetal ultrasound videos using deep learning[J]. Applied Sciences, 2021, 11(1): 371.

[12] CIVIT-MASOT J, LUNA-PEREJÓN F, DOMÍNGUEZ MORALES M, et al. Deep learning system for COVID-19 diagnosis aid using X-ray pulmonary images[J]. Applied Sciences, 2020, 10(13): 4640.

[13] LI J, MONROE W, RITTER A, et al. Deep reinforcement learning for dialogue generation[J]. ArXiv preprint arXiv:1606.01541, 2016.

[14] NI J, YOUNG T, PANDELEA V, et al. Recent advances in deep learning based dialogue systems: A systematic survey[J]. Artificial Intelligence Review, 2023, 56(4): 3055-3155.

[15] HEATON J B, POLSON N G, WITTE J H. Deep learning for finance: Deep portfolios[J]. Applied Stochastic Models in Business and Industry, 2017, 33(1): 3-12.

[16] HUANG J, CHAI J, CHO S. Deep learning in finance and banking: A literature review and classification[J]. Frontiers of Business Research in China, 2020, 14(1): 13.

[17] SMOLA A J, SCHÖLKOPF B. A tutorial on support vector regression[J]. Statistics and Computing, 2004, 14: 199-222.

[18] QUINLAN J R. Induction of decision trees[J]. Machine Learning, 1986, 1: 81-106.

[19] HUNT E B, MARIN J, STONE P J. Experiments in induction[J]., 1966.

[20] PHILLIPS P J, PHILLIPS P J, HAHN C A, et al. Four principles of explainable artificial intelligence[J]., 2021.

[21] MILLER T. Explanation in artificial intelligence: Insights from the social sciences[J]. Artificial Intelligence, 2019, 267: 1-38.

[22] FRIEDMAN J H. Multivariate adaptive regression splines[J]. The Annals of Statistics, 1991, 19(1): 1-67.

[23] FRIEDMAN J H. Greedy function approximation: a gradient boosting machine[J]. Annals of Statistics, 2001: 1189-1232.

[24] FRIEDMAN J H, POPESCU B E. Predictive learning via rule ensembles[J]. ArXiv preprint arXiv:0811.1679, 2008.

[25] INGLIS A, PARNELL A, HURLEY C B. Visualizing variable importance and variable interaction effects in machine learning models[J]. Journal of Computational and Graphical Statistics, 2022, 31(3): 766-778.

[26] APLEY D W, ZHU J. Visualizing the effects of predictor variables in black box supervised learning models[J]. Journal of the Royal Statistical Society Series B: Statistical Methodology, 2020, 82(4): 1059-1086.

[27] HOOKER G. Generalized functional anova diagnostics for high-dimensional functions of dependent variables[J]. Journal of Computational and Graphical Statistics, 2007, 16(3): 709-732.

[28] CROMBECQ K, DE TOMMASI L, GORISSEN D, et al. A novel sequential design strategy for global surrogate modeling[C]//Proceedings of the 2009 Winter Simulation Conference (WSC). 2009: 731-742.

[29] GOLDSTEIN A, KAPELNER A, BLEICH J, et al. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation[J]. Journal of Computational and Graphical Statistics, 2015, 24(1): 44-65.

[30] GOLDSTEIN A, KAPELNER A, BLEICH J, et al. Package ‘ICEbox’[J].,

[31] WACHTER S, MITTELSTADT B, RUSSELL C. Counterfactual explanations without opening the black box: Automated decisions and the GDPR[J]. Harv. JL & Tech., 2017, 31: 841.

[32] DANDL S, MOLNAR C, BINDER M, et al. Multi-objective counterfactual explanations[C]//International Conference on Parallel Problem Solving from Nature. 2020: 448-469.

[33] RIBEIRO M T, SINGH S, GUESTRIN C. " Why should i trust you?" Explaining the predictions of any classifier[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016: 1135-1144.

[34] SLACK D, HILGARD S, JIA E, et al. Fooling lime and shap: Adversarial attacks on post hoc explanation methods[C]//Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society. 2020: 180-186.

[35] SHAPLEY L S, et al. A value for n-person games[J]. Classics in Game Theory, 1953: 307-317.

[36] ŠTRUMBELJ E, KONONENKO I. Explaining prediction models and individual predictions with feature contributions[J]. Knowledge and Information Systems, 2014, 41: 647-665.

[37] OLAH C, MORDVINTSEV A, SCHUBERT L. Feature visualization[J]. Distill, 2017, 2(11): e7.

[38] OLAH C, SATYANARAYAN A, JOHNSON I, et al. The building blocks of interpretability[J]. Distill, 2018, 3(3): e10.

[39] SIMONYAN K, VEDALDI A, ZISSERMAN A. Deep inside convolutional networks: Visualising image classification models and saliency maps[J]. ArXiv preprint arXiv:1312.6034, 2013.

[40] SHRIKUMAR A, GREENSIDE P, KUNDAJE A. Learning important features through propagating activation differences[C]//International Conference on Machine Learning. 2017: 3145-3153.

[41] KIM B, WATTENBERG M, GILMER J, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)[C]//International Conference on Machine Learning. 2018: 2668-2677.

[42] KOH P W, NGUYEN T, TANG Y S, et al. Concept bottleneck models[C]//International Conference on Machine Learning. 2020: 5338-5348.

[43] GHORBANI A, WEXLER J, ZOU J Y, et al. Towards automatic concept-based explanations[J]. Advances in Neural Information Processing Systems, 2019, 32.

[44] ERHAN D, BENGIO Y, COURVILLE A, et al. Visualizing higher-layer features of a deep network[J]. University of Montreal, 2009, 1341(3): 1.

[45] MCCULLOCH W S, PITTS W. A logical calculus of the ideas immanent in nervous activity[J]. The Bulletin of Mathematical Biophysics, 1943, 5: 115-133.

[46] HORNIK K, STINCHCOMBE M, WHITE H. Multilayer feedforward networks are universal approximators[J]. Neural Networks, 1989, 2(5): 359-366.

[47] HOCHREITER S. Untersuchungen zu dynamischen neuronalen Netzen[J]. Diploma, Technische Universität München, 1991, 91(1): 31.

[48] HOCHREITER S, BENGIO Y, FRASCONI P, et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies[Z]. 2001.

[49] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks[C]//Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. 2010: 249-256.

[50] LOWE D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60: 91-110.

[51] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition: vol. 1. 2005: 886-893.

[52] DENG J, DONG W, SOCHER R, et al. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE Conference on Computer Vision and Pattern Recognition. 2009: 248-255.

[53] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30.

[54] SPARCK JONES K. A statistical interpretation of term specificity and its application in retrieval[J]. Journal of Documentation, 1972, 28(1): 11-21.

[55] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[J]. Advances in Neural Information Processing Systems, 2013, 26.

[56] ZHANG H. The optimality of naive Bayes[J]. Aa, 2004, 1(2): 3.

[57] COVER T, HART P. Nearest neighbor pattern classification[J]. IEEE Transactions on Information Theory, 1967, 13(1): 21-27.

[58] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[59] KARPATHY A, JOHNSON J, FEI-FEI L. Visualizing and understanding recurrent networks[J]. ArXiv preprint arXiv:1506.02078, 2015.

[60] ZEILER M D, FERGUS R. Visualizing and understanding convolutional networks[C]//Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13. 2014: 818-833.

[61] SELVARAJU R R, COGSWELL M, DAS A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 618-626.

[62] ZHOU B, KHOSLA A, LAPEDRIZA A, et al. Learning deep features for discriminative localization[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2921-2929.

[63] BACH S, BINDER A, MONTAVON G, et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation[J]. PloS One, 2015, 10(7): e0130140.

[64] LECUN Y, CORTES C, BURGES C. MNIST handwritten digit database[J]. ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist, 2010, 2.

[65] KRIZHEVSKY A, HINTON G, et al. Learning multiple layers of features from tiny images[J]. Technical report, 2009.

[66] GARNIER S, Ross, Noam, et al. Viridis(Lite) - Colorblind-Friendly Color Maps for R[A/OL]. 2023. https://sjmgarnier.github.io/viridis/.

[67] HAN S, POOL J, TRAN J, et al. Learning both weights and connections for efficient neural network[J]. Advances in Neural Information Processing Systems, 2015, 28.

[68] HE Y, ZHANG X, SUN J. Channel pruning for accelerating very deep neural networks[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1389-1397.

[69] LECUN Y, DENKER J, SOLLA S. Optimal brain damage[J]. Advances in Neural Information Processing Systems, 1989, 2.

[70] HASSIBI B, STORK D. Second order derivatives for network pruning: Optimal brain surgeon[J]. Advances in Neural Information Processing Systems, 1992, 5.

[71] MACQUEEN J, et al. Some methods for classification and analysis of multivariate observations[C]//Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability: vol. 1: 14. 1967: 281-297.

[72] HUBERT L, ARABIE P. Comparing partitions[J]. Journal of Classification, 1985, 2: 193-218.

[73] ROUSSEEUW P J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis[J]. Journal of Computational and Applied Mathematics, 1987, 20: 53-65.

[74] RAND W M. Objective criteria for the evaluation of clustering methods[J]. Journal of the American Statistical Association, 1971, 66(336): 846-850.

[75] EVERINGHAM M, VAN GOOL L, WILLIAMS C K I, et al. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results[Z]. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[76] SOCHER R, PERELYGIN A, WU J, et al. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank[C]//Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2013: 1631-1642.

[77] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[J]. ArXiv preprint arXiv:1409.1556, 2014.

[78] PENNINGTON J, SOCHER R, MANNING C D. GloVe: Global Vectors for Word Representation[C]//Empirical Methods in Natural Language Processing (EMNLP). 2014: 1532-1543.

[79] HE K, ZHANG X, REN S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]//Proceedings of the IEEE International Conference on Computer Vision. 2015: 1026-1034.

[80] FONG R C, VEDALDI A. Interpretable explanations of black boxes by meaningful perturbation[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 3429-3437.

[81] ZHANG J, BARGAL S A, LIN Z, et al. Top-down neural attention by excitation backprop[J]. International Journal of Computer Vision, 2018, 126(10): 1084-1102.

[82] WANG H, WANG Z, DU M, et al. Score-CAM: Score-weighted visual explanations for convolutional neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2020: 24-25.

[83] CHATTOPADHAY A, SARKAR A, HOWLADER P, et al. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks[C]//2018 IEEE Winter Conference on Applications of Computer Vision. 2018: 839-847.

致　　谢

三年时间转眼匆匆即逝，校园时光再一次如白驹从指缝溜走。回首研究生三年，有欢笑有眼泪，有喜悦有苦恼。在这里，遇见传授我知识的人、带给我快乐的人、引领我成长的人，在此，向所有认识的人、帮助过我的人由衷地表示感谢。

感谢我的导师申富饶教授。学生朽木，但申老师仍孜孜不倦向我传道授业解惑。无论是与老师的个人讨论，还是每周一的组会，无一不是我难得可贵的学习机会。申老师对我的教诲与指点，将成为我宝贵的精神财富，伴随一生。

感谢我的同门 RINC 组的所有成员。鄙人嘴钝，同门所给予的陪伴、快乐难以言表。生活学习的方方面面，有你们的存在让这条道路显得额外的难忘与灿烂。

感谢我的朋友以及家人。亲友的支持与肯定是我力量的来源与前进的动力，短短一行感谢不足回报。

感谢电影、文学、游戏还有体育。感谢遇见的一切助我成长之事，活过的刹那，绚烂而美好。

简历与科研成果

基本信息

张凌茗，男，汉族，1997 年 5 月出生，重庆市人。

教育背景

2021 年 9 月 - 2024 年 6 月	南京大学计算机科学与技术系	硕士
2014 年 9 月 - 2018 年 6 月	南京大学物理学院	本科

攻读硕士学位期间发表的学术成果

1. 窦慧, **张凌茗**, 韩峰, 申富饶, 赵健. 卷积神经网络的可解释性研究综述 [J]. 软件学报,2022

攻读硕士学位期间参与的科研课题

1. 科技部重大项目“基于神经可塑性的脉冲神经网络高效学习机制与类脑智能系统”（参与课题年限 2021 年 9 月——2024 年 6 月），负责神经网络模型相关研究。
2. 国家电网项目“基于多维巡检影像匹配和对比技术的变电设备缺陷分析技术研究”（参与课题年限 2021 年 9 月——2022 年 12 月），负责图像对比与目标检测相关研究。