

doi: 10.7690/bgzd.2025.11.010

计及数据点活跃度的光互连网络数据资源冗余迭代消除方法

梁志勇^{1,2}, 杨 明^{1,2}, 江荣旺^{1,2}

(1. 三亚学院信息与智能工程学院, 海南 三亚 572022; 2. 三亚学院容淳铭院士工作站, 海南 三亚 572022)

摘要: 为有效减少数据中心光互连网络资源的冗余和迭代消除, 提高网络性能和效率, 以深度神经网络为技术依托, 针对数据中心光互连网络, 设计一种新的资源冗余迭代消除方法。依据数据中心光互连网络数据资源的特征相似性, 设置数据资源的分块边界, 并求取分类分量, 通过动态查找资源冗余特征, 分离冗余数据与正常数据, 从根本上保证冗余消除效果; 结合粗糙度函数, 分析冗余资源特征在空间内的离散型分布状态, 并压缩冗余参数; 采用深度神经网络算法计算数据冗余点的活跃度, 滤除超过活跃度阈值的资源, 实现冗余迭代消除。实验结果表明: 应用该方法后, 网络的空间缩减比最高可以达到 0.55, 证明该方法具有明显的有效性。

关键词: 深度神经网络; 数据中心; 网络资源冗余; 迭代消除;

中图分类号: TP183 **文献标志码:** A

Research on Iterative Elimination of Data Resource Redundancy in Optical Interconnection Networks Taking into Account Data Point Activity

Liang Zhiyong^{1,2}, Yang Ming^{1,2}, Jiang Rongwang^{1,2}

(1. School of Information & Intelligence Engineering, University of Sanya, Sanya 572022, China;

2. Academician Workstation of Chunming Rong, University of Sanya, Sanya 572022, China)

Abstract: The data features have a discrete distribution state in the network space, which leads to a relatively low reduction in the network space after eliminating data redundancy, affecting the effectiveness of redundancy elimination. Therefore, this study focuses on data centers and iteratively eliminates redundant data in optical interconnect network resources, taking into account the activity of data points. Firstly, based on the feature similarity of data resources in the data center's optical interconnection network, the partition boundaries of data resources are set, and the classification components are calculated. By dynamically searching for redundant features of resources, redundant data is separated from normal data to fundamentally ensure the effectiveness of redundancy elimination; Then, combined with the roughness function, analyze the discrete distribution state of redundant resource features in space and compress the redundant parameters; Finally, deep neural network algorithms are used to calculate the activity of redundant data points, filter out resources that exceed the activity threshold, and achieve iterative elimination of redundancy. The experimental results show that after applying this method, the maximum spatial reduction ratio of the network can reach 0.55, indicating that this method has significant effectiveness.

Keywords: deep neural networks; data center; network resource redundancy; iterative elimination

0 引言

数据中心光互连网络是一种高速、高效、可靠的网络架构, 主要应用于大型数据中心。这种网络采用光信号作为传输介质, 通过光路交换机进行高速数据传输和交换。相比于传统的电子分组交换机, 光互连网络具有高吞吐量、低延迟和低能耗等优势, 能够满足大型数据中心对于大规模数据传输和处理的要求。

在数据中心光互连网络中, 为了应对日益增长的数据传输需求, 设置了大规模的高性能光路交换机和光模块, 但造成了资源冗余问题。

资源冗余是指网络中存在未使用的资源, 不仅浪费网络资源, 而且会增加网络建设和维护成本。迭代消除是指网络中存在重复配置, 这不仅增加了网络的复杂性, 而且影响了网络的性能和效率。如何有效减少数据中心光互连网络资源的冗余和迭代消除, 提高网络性能和效率, 是一个亟待解决的问题。

文献[1]使用相关系数和独立信息等统计方法来计算不同指标之间的关联性, 基于指标之间的关联性和信息论原理, 计算每个指标的独立信息度量, 并对每个指标进行排序。选择独立信息测量值较高的指标作为核心指标, 剔除冗余指标。该方法可以

收稿日期: 2024-10-20; 修回日期: 2024-11-23

基金项目: 海南省重点研发项目(ZDYF2023GXJS007); 海南省院士创新平台科研项目(YSP TZ202145); 海南省高等学校科学研究项目(Hnky2025-33); 三亚学院重大专项课题(USY22XK-04)

第一作者: 梁志勇(1985—), 男, 湖南人, 硕士。

有效识别和消除冗余指标，提高评估的准确性。同时，避免了主观因素的影响，使评价结果更加客观。该方法计算独立信息测度的过程存在复杂度较高的问题，实用性偏低。数据质量不高或数据量不足，会影响独立信息计量的计算效果。文献[2]基于定位数据的特征和先验知识，构建贝叶斯网络模型，并利用贝叶斯网络模型进行概率推理，计算传感器数据的不确定性。通过比较不同传感器数据的不确定性来确定冗余信息的来源，采用适当的信息过滤机制，如删除、合并或更正冗余数据。该方法具有强大的建模能力，适用于定位数据的可变性和不确定性，具有较高的鲁棒性和适应性，但该方法构建模型和概率推理过程涉及到大量的概率计算和优化问题，使得运算流程比较复杂。

基于以上分析，笔者以深度神经网络为技术依托，针对数据中心光互连网络，设计一种新的资源冗余迭代消除方法。

1 数据中心光互连网络资源冗余迭代消

1.1 网络资源冗余特征动态查找

数据中心光互连网络资源具有相似性。在剔除冗余数据时，特征相似的资源数据可能会导致出现二次分类处理的现象，容易出现消除错误的情况，影响资源冗余迭代消除结果的准确性^[3]。在设计网络资源冗余迭代消除方法之前，应对资源冗余特征进行动态查找，将具有相似性的冗余数据与正常数据分离，保证冗余消除效果^[4]。

假设网络中的初始数据集合为 A ，集合中最大类别和最小类别分别为 $A_{\max}$ 和 $A_{\min}$ 。

$$A = \sum \frac{t_s}{A_{\max} \times A_{\min}} \quad (1)$$

式中 t_s 为时间数据集合。样本集合中的平均分类密度特征可以表示为：

$$p_m = \frac{C_l \times A}{k_0} \quad (2)$$

式中： C_l 为数据连接字节的长度； k_0 为数据定长分块的数量。

设置资源数据的分块边界，对不同数据块长度进行编码^[5]，则每个数据块的哈希值为：

$$h_s = \sqrt{\frac{|p_m|}{\sum g_l / l_l}} \quad (3)$$

式中： g_l 为哈希函数，哈希函数可以将任意大小的数据块作为输入，并产生固定长度的输出； l_l 为很

熟悉函数对应的比特值。

在网络空间的海量数据里，冗余数据的特征在时间和空间上通常为离散状态^[6]；因此，特征查找的过程也应基于离散状态下，结合数据块的哈希值求取数据资源的分类分量，表达式如下：

$$S_l = w_0 (1 + j_0) \exp \left(-\frac{h_s}{k_1} \right) \quad (4)$$

式中： w_0 为布隆转换矩阵； j_0 为相似度矩阵； k_1 为参考度，是用于衡量数据点被选作类代表点合适程度的指标。在数据处理和聚类分析中，选择合适的类代表点对于提高聚类效果和降低计算复杂度至关重要。

利用矢量平均法定义数据集中分类项的中心点^[7]。计算公式如下：

$$c_x = \frac{\sum S_l}{n_0 \times x_p} + v_0 \quad (5)$$

式中： n_0 为类别中点个数； x_p 为相邻 2 个数据块间的距离； v_0 为系数矩阵。

将距离分类超平面较近的样本归为支持向量，由此计算出与其相关的分类似真度为：

$$\rho_f = c_x \times \sum_{i=1}^{A_{\min}} (g_h + l_2) H(t) \quad (6)$$

式中： g_h 为超平面的邻近数据值； l_2 为特征样本集合； $H(t)$ 为双曲正切函数。

根据数据资源迭代的次数，可推导出符合约束条件的数据特征的分辨系数^[8]，即：

$$a_z = \frac{\rho_f}{\|\gamma_0 / \alpha_1\|^2} / t_y \quad (7)$$

式中： γ_0 为迭代次数； α_1 为最后一个样本包含的特征数据量； t_y 为可调参数。

进而可利用式(8)对资源冗余特征进行动态查找，即：

$$f_d = a_z \sum p_q E(x) g_c [w_c (y_0 + k_p)] \quad (8)$$

式中： f_d 为资源冗余特征判别函数，在数据处理、系统优化和资源管理中起着关键作用，该函数旨在识别并量化数据集中存在的冗余特征，从而帮助决策者或系统自动优化资源配置，提高整体效率； p_q 为阈值参数； $E(x)$ 为权函数； g_c 为可传递相关矢量； w_c 为数据局部极小值； y_0 为资源子空间维度； k_p 为相似数据的数量。

利用上式对网络资源数据进行相似性判别，并识别出具有相似特征的冗余数据，实现对冗余特征

的动态查找, 为资源冗余参数压缩提供数据基础。

1.2 冗余参数压缩

在冗余资源规模较大特性的基础上, 对资源的冗余特征进行了动态识别与查找, 结合资源分割原理, 将资源数据划分为术语集、边界集和残差集 3 类, 对数据的概念信息、实体运算以及两者之间的关系进行描述^[9]。

由前文分析可知, 数据间具有较高的相似度, 且资源数据结构中的基本单位包含多种数据属性, 采用信息熵算法计算属性数据集合的冗余参数熵^[10-11]。计算公式如下:

$$x_s = \frac{f_d}{B(x) \sqrt{\lg a_g \times r_k}} \quad (9)$$

式中: f_d 为资源冗余特征判别函数; $B(x)$ 为信息熵函数; a_g 为 $[0, 1]$ 区间上的随机数; r_k 为变异系数。

基于相似特征的查找结果, 建立粗糙度函数, 并计算冗余资源特征在空间内的分布状态。表达式如下:

$$S(k) = w_u \times x_s \times z_0 \quad (10)$$

式中: w_u 为初始数据点到当前数据点的扩展步长; x_s 为多变量非线性函数; z_0 为冗余特征的基本概率赋值。

设定初始资源的术语集、边界集和残差集的窗口长度分别为 b_1 、 b_2 和 b_3 。则在回溯窗口下, 原始资源中各类数据集的增益匹配系数可表示为:

$$d_p = \frac{S(k) \sum \alpha_2}{(b_1 + b_2 + b_3) \sqrt{\phi_0 / L_c}} \quad (11)$$

式中: α_2 为时间常数; ϕ_0 为窗口计算量; L_c 为第 c 个数据块的滚动校验值。

为了将冗余特征与回溯窗口的滑动长度进行快速匹配, 对匹配适应度进行设定^[12], 表达式如下:

$$F_s = \sqrt{\mathbf{D}_u \|d_p / t_p\|^2} \quad (12)$$

式中: t_p 为非重复数据块; $\mathbf{D}_u$ 为状态矩阵。

当数据块匹配成功后, 窗口将直接移动到下一个数据块上^[13]。则此时的冗余参数的波动因子可表示为:

$$h_s = F_s \times \frac{\phi_1}{\gamma_k} + c_m \quad (13)$$

式中: ϕ_1 为冗余参数大小; γ_k 为冗余参数的置信度; c_m 为混合系数。

在不同数据集中, 将待压缩的冗余参数缓存在

等待序列中, 则利用冗余参数的拥挤程度与拥挤概率之间的关系判断缓存的阻塞程度^[14]。表达式如下:

$$V(s) = \begin{cases} 0, & I_q = 0 \\ \frac{I_q}{\pi}, & 0 < I_q < \pi \\ 1, & I_p \geq \pi \end{cases} \quad (14)$$

式中: I_q 为缓存队列的长度; π 为缓存时间。

为了解决冗余参数的拥挤现象, 对缓冲队列中的最大值与最小值进行优化^[15], 则得到的冗余参数的随机梯度为:

$$z_t = h_s \times V(s) \times L_\alpha \quad (15)$$

式中 L_α 为冗余参数判断值的层次系数。

基于式(15), 建立关联矩阵, 以判断数据点是否存在冗余参数的关联规则^[16]。表达式如下:

$$\mathbf{G}_m = z_t \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n'} \\ m_{21} & m_{22} & \cdots & m_{2n'} \\ \vdots & \vdots & & \vdots \\ m_{\sigma 1} & m_{\sigma 2} & \cdots & m_{\sigma n'} \end{bmatrix} \quad (16)$$

式中 $m_{\sigma n'}$ 为在 n' 时刻内冗余参数达到预设速率 σ 的标度值。

通过归一化冗余参数的权重系数, 对冗余参数进行优化压缩。计算公式为:

$$\vartheta = \mathbf{G}_m \times \sum m_s + \sqrt{\omega_t} \times h_e \quad (17)$$

式中: m_s 为冗余参数在数据集边缘的程度; ω_t 为资源 t 被访问的频率; h_e 为冗余参数的键值; ϑ 为冗余参数压缩结果。

基于资源冗余特征的动态识别与查找结果, 对资源类型进行分割, 结合冗余参数的随机梯度与关联规则, 对冗余参数进行优化压缩处理, 便于后续资源冗余的迭代消除。

1.3 基于深度神经网络的资源冗余迭代滤除

经过上述计算与分析过程后, 即可进行资源冗余点的迭代消除。笔者采用深度神经网络算法对冗余资源进行目标位置的确定, 并通过模型训练求取资源冗余点的活跃度, 以此实现冗余滤除。基于深度神经网络的资源冗余迭代滤除过程如图 1 所示。可以看出: 在基于深度神经网络的非线性变换中, 滤除过程仅需要 2 种数据, 即目标数据和查询数据。笔者将冗余特征作为目标数据, 将冗余参数压缩结果作为查询数据, 在输入层中, 将 2 种数据的矢量合并为一个矢量, 将其作为深度神经网络的输

入值^[17]。表达式为：

$$W_m = \vartheta \times H_g. \quad (18)$$

式中： ϑ 为冗余参数压缩结果； H_g 为状态矩阵。

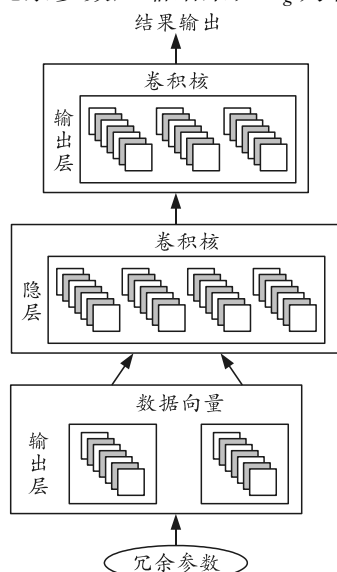


图1 基于深度神经网络的资源冗余迭代滤除过程

通过偏置项将输入数据传输至网络的隐藏层，获取冗余资源的位置坐标，即：

$$Q_c = \begin{cases} x_c = W_m \times l_\theta \\ y_c = W_m / T_l \end{cases}. \quad (19)$$

式中： l_θ 为残差项； T_l 为一阶滞后变量。

采用加权矩阵对网络结构进行更新，并统计获取新的连接神经元数量如下：

$$M_s = \sqrt{(x_c + y_c)^2} / j_h. \quad (20)$$

式中： j_h 为加权矩阵。根据连接神经元数量，计算资源冗余点的活跃度，并对其进行消除。表达式为：

$$\xi = \psi(t) \times v_k \times M_s. \quad (21)$$

式中： $\psi(t)$ 为特征函数； v_k 为冗余点的总活跃时间。

利用上式计算数据冗余点的活跃度，将超过活跃度阈值 Ω 的资源进行滤除，保留小于阈值 Ω 的网络资源数据，由此实现对资源冗余的迭代消除。至此，完成基于深度神经网络的数据中心光互连网络资源冗余迭代消除方法的设计。

2 实例论证分析

2.1 实验准备

实验采用包括多个节点和光链路的数据中心光互连网络作为研究对象，并模拟数据传输和通信，使用 TensorFlow 和 PyTorch 深度学习框架来构建和训练深度神经网络模型，并使用 NS-3 等网络模拟器模拟数据中心光互连网络的运行环境。

实验中使用的数据样本数量为 10 000 个，每个样本包含网络性能指标，如网络连接信息、传输速率和延迟。通过训练深度神经网络模型，预测和优化数据中心光互连网络的资源利用率和性能。数据集的详细描述如表 1 所示。

表 1 实验数据集描述

样本编号	样本类型	数据量个数	属性个数	冗余率/%
1	链路层数据样本	100	2	20
2	物理层数据样本 (环境数据)	210	4	15
3	物理层数据样本 (光信号强度数据)	320	3	16
4	网络层数据样本	154	4	23
5	应用层数据样本	203	1	22
6	管理数据样本	145	5	20
7	安全相关数据样本	189	9	19
8	时间戳数据	364	2	19

依据资源冗余的判断规则，按照时间顺序逐一消除，迭代消除流程如图 2 所示。

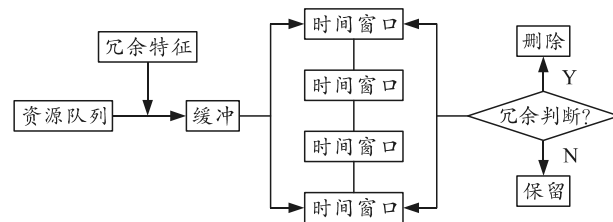


图2 资源冗余消除流程

实验设置了 1 个工作攻击节点和 6 个任务攻击节点。所有节点都具有 AMD CPU 和 32 GB 内存，并通过 2 Gbps 以太网连接到攻击节点。在 Opnet 仿真环境中，模拟网络路由节点两端的宽带链路拥塞现象。使用 YB-CA266 型号存储服务器相关数据，使用 System x3650M3 型号备份服务器相关数据，并通过 Windows 10 处理器处理数据。

2.2 实验说明

为保证深度神经网络的泛化能力和训练效果，实验中采用 3 层结构的网络，其具体参数设置如表 2 所示。

表 2 深度神经网络结构设计

层数	神经元数量	激活函数	权重矩阵维度	偏置向量维度
输入层	100	ReLU	1 000*1 000	1 000*1
隐藏层 1	500	ReLU	1 000*500	500*1
隐藏层 2	200	ReLU	500*200	200*1
输出层	100	Softmax	200*100	100*1

表 2 中，输入层有 100 个神经元，用于接收来自数据集的特征，并作为输入；隐藏层 1 有 500 个神经元，隐藏层 2 有 200 个神经元，用于数据转换和抽象；输出层有 100 个神经元，用于预测网络性能。

能指标。

2.3 资源冗余迭代消除结果与分析

实验中,引入文献[1]基于独立信息测度的冗余消除法(方法 1)、文献[2]基于贝叶斯网络的冗余信息过滤法(方法 2)作为对比。分别采用这 2 种方法和本文中方法对数据中心光互连网络资源冗余进行迭代消除。采用空间缩减比来评估不同方法的消除性能。该指标的计算公式为:

$$Z=\eta_1/\eta_2 \quad (22)$$

式中 η_1 、 η_2 分别为网络资源冗余的删除量和资源样本总量。

利用式(22)分别计算不同方法的空间缩减比,该指标数值越高,表明方法对于资源冗余的消除效果越好。对比结果如图 3 所示。

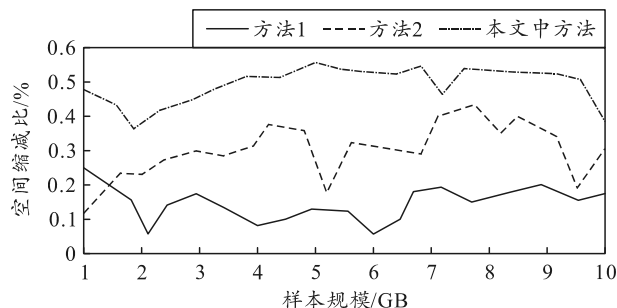


图 3 基于不同方法下的资源冗余消除结果对比

根据图 3 可知:在相同网络资源规模条件下,应用本文中方法进行资源冗余消除后,网络的空间缩减比明显高于对照组方法,在数值上最高可以达到 0.55。

为了进一步验证本文中方法在消除资源冗余方面的有效性,接着进行了对比测试。比较了本文中方法与 2 种对比方法在消除相同数量冗余资源时所消耗的时间,对比结果如图 4 所示。

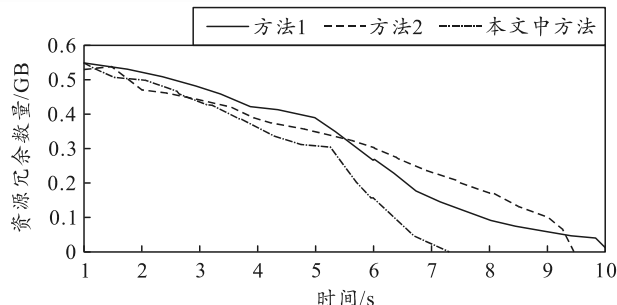


图 4 基于不同方法下的资源冗余消除时间对比

根据图 4 可知:在相同网络资源规模下,笔者提出的方法在消除资源冗余方面所需时间最短,大约仅需 7 s 即可完全消除闲聊冗余资源,相比之下,其他 2 种对比方法分别需要 9.5 和 10 s。这表明本

文中方法在处理冗余资源时具有较高的效率,显著缩短了处理时间,从而有效滤除网络资源的冗余数据。产生上述结果的原因如下:

1) 基于特征相似性的资源分类:该方法通过识别数据资源的分类分量,可以将资源按照其特征相似性进行分组,使得相似的资源被归为一类。这样做可以大大减少资源之间的差异性,同时也减少了网络中不必要的重复资源。

2) 动态查找冗余特征:该方法能够动态地查找网络中存在的冗余特征,通过分析资源的分布状态和粗糙度函数来确定冗余参数。通过对冗余参数的压缩,可以进一步减少网络中的冗余资源。

3) 深度神经网络算法计算活跃度:该方法利用深度神经网络算法计算数据冗余点的活跃度,活跃度高的资源会被保留,而低活跃度的资源则会被滤除。通过设置适当的活跃度阈值,可以移除那些没有被广泛使用或者没有必要存在的资源,从而进一步优化网络空间利用率。

2.4 对比实验分析

为进一步验证本文中方法在资源冗余消除方面的优越性,采用加速比这一指标来定性评价不同方法的资源冗余消除性能。加速比的计算方式如下:

$$Q_s=Q_1/Q_2 \quad (23)$$

式中: Q_1 为串行执行时间,表示任务在单核或单个处理器上的执行时间; Q_2 为并行执行时间,表示任务在多个核或多个处理器上的执行时间。加速比越高,表明方法的消除效果越好。对比结果如图 5 所示。

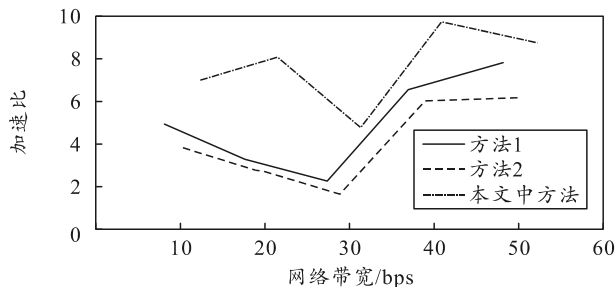


图 5 基于不同方法的网络加速比结果对比

根据图 5 可知:在数值方面,应用本文中方法对网络资源进行冗余迭代消除后,网络的加速比最高可以达到 9.7,而应用方法 1 和 2 后,网络的加速比最高分别达到 7.8、6.0。结果说明本文中方法网络空间的利用率得到了提升,进而表明本文中方法对于网络资源冗余消除的可靠性。