

神经网络在数据中心能耗模型研究中的应用

智伟威, 周新星

(航天通信中心, 北京 100830)

摘要: 数据中心产业用户规模的不断增加和企业级应用的持续升级, 导致数据中心规模不断扩张, 随之而来的能耗和资源利用率问题逐渐成为研究热点。目前传统的数据中心节能策略中主要存在监控资源单一化和基于积分方式的能耗模型建立所导致的节能效率不高的问题。针对以上问题, 提出一种基于神经网络的数据中心能耗模型研究, 该架构包括实时自动化监控模块、能耗模块和基于神经网络算法的资源分配模块, 最终利用神经网络实现数据中心资源的高效分配。并在实验室进行仿真测试, 得到能耗节省率在 30% 以上。

关键词: 神经网络; 数据中心; 能耗

中图分类号: TP391.9, TP183 **文献标识码:** B

Application of Neural Network Algorithm in Data Center Energy Consumption Model Research

ZHI Wei - wei ZHOU Xin - xing

(Aerospace Communication Center, Beijing 100830, China)

ABSTRACT: The increasing scale of users in the data center industry and the continuous upgrading of enterprise-level applications have led to the continuous expansion of the data center scale, and the consequent energy consumption and resource utilization problems have gradually become research hotspots. At present, the traditional data center energy-saving strategy mainly has the problem of low energy-saving efficiency caused by the simplification of monitoring resources and the establishment of energy-based models based on the integral method. In view of the above problems, this paper proposes a data center energy consumption model based on neural network algorithm. The architecture includes real-time automatic monitoring module, energy consumption module and resource allocation module based on neural network algorithm. Finally, the neural network algorithm was used to realize the efficient allocation of data center resources. The simulation test was carried out in the laboratory, and the energy saving rate was over 30%.

KEYWORDS: Neural network; Data center; Energy consumption

1 引言

随着数据中心产业的不断成熟和发展, 企业级的数据中心也越来越多, 用户规模的不断增加和企业级应用的持续更新迭代需要越来越大的计算资源, 从而对数据中心规模提出更大的扩张需求。随之产生的大规模数据中心的能耗和资源利用率问题成为业界关注的热点。因此, 如何设计出更加高效节能的数据中心资源分配策略^[1], 提高计算资源利用率, 对最终实现数据中心的高效节能具有重大意义。

2 数据中心节能技术现状概述

2.1 传统方式

面向数据中心, 传统的资源分配策略主要是基于虚拟化资源^[2]的调度技术, 通过虚拟化技术将有限的物理资源重新分配。底层利用率较低的虚拟机, 会通过虚拟机动态迁移技术被整合到同一台服务器上, 空置的服务器将会被关闭或强制休眠^[3], 以达到降低能耗的目的。

2.2 节能技术存在的问题

影响数据中心高效节能资源分配方法的因素主要有以下三个: 一是数据中心资源监测方法是否准确高效^[4], 是否能够反馈准确的实时资源数据为资源分配策略提供支撑; 二

收稿日期: 2019-03-01 修回日期: 2019-04-13

是否能建立能够准确地计算出数据中心能源消耗情况的能耗计量模型^[5];三是能耗计量模型的高效率自适应算法^[6],在确定能耗模型后,能够进行高效稳定的极值寻优求解,并能快速地进行资源的分配。而目前的资源分配策略主要存在以下不足:首先,监控资源的方法基于单一的轮询方式或者事件驱动方式,在监测云数据中心虚拟资源的动态变化时灵活性较低;其次,目前通常采用对CPU利用率或者内存利用率等因子进行时间积分的方式建立能耗模型,在通过遗传算法对模型进行极值寻优,从而进行资源分配,这种做法不利于提高工程运用中的效率,而且也不能自适应的去调节模型参数。

3 神经网络算法在数据中心节能策略中的应用

3.1 整体设计与实现

为了解决以上问题,本文提出一种基于神经网络的数据中心能耗模型,其架构主要包括实时自动化监控模块、能耗模块和基于神经网络模型的资源分配模块。该架构既能灵活地监测云数据中心的虚拟资源,又能够提供自适应、自组织的资源分配策略,并将监控系统采集的数据不断加入到神经网络的训练集中,形成闭环的自适应数据中心资源分配系统,以达到数据中心节能的目的。

基于神经网络的数据中心能耗模型共包括基于代理节点的实时自动化监控模块、能耗模块和神经网络求解的节能策略模块,整体架构图如图1所示。首先,将数据中心集群中的物理主机进行编号(1,2...m),然后对物理主机内的虚拟机编号为(1.1,1.2...1.n),即 $V_{(m,n)}$,第m台主机上的第n台虚拟机,自动化监控模块将各个虚拟机的(R_{status} , C_{status})的值获取出来,以便选取不同的监控方式,并将监测到的各虚拟机的(U_{cpu} , U_{mem})获取出来并传给能耗模型模块,能耗模型

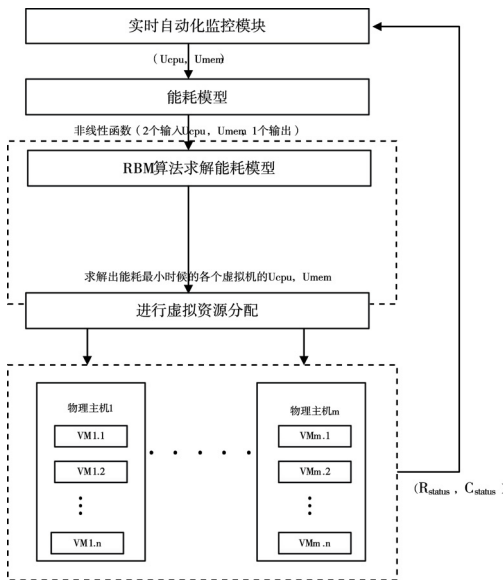


图1 整体架构图

模块获取2个输入参数 U_{cpu} 和 U_{mem} ,建立以 U_{cpu} 和 U_{mem} 为输入参数,以总体能耗 E 为参数的非线性函数模型。

基于神经网络模型的资源分配模块获取到 U_{cpu} 和 U_{mem} 并存储,利用RBM神经网络求解能耗模型,然后求解出能耗最小时,所对应的 U_{cpu} 和 U_{mem} 的参数矩阵,最后依据 U_{cpu} 和 U_{mem} 进行虚拟资源的分配。

此处需要说明的是,CPU、内存、网络接口以及磁盘存储量是服务器计算节点能耗的主要部分,本文中主要在建立能耗模型后,利用神经网络中的RBM算法对能耗模型进行非线性函数拟合求解,得出能耗最小时各虚拟机所对应的 U_{cpu} 和 U_{mem} ,从而进行资源分配,故本文中不考虑网络因素,并将其它因素作为基础常量进行建立模型。

3.2 实时自动化监控模块

实时自动化监控模块的架构图如图2所示,主要包括监控模块控制器、数据存储模块、集群监控器、代理节点五个部分。通过周期性的接收代理节点发送该节点处的CPU利用率 U_{cpu} 和内存的利用率 U_{mem} 数据,以此作为数据中心的资源监控信息。

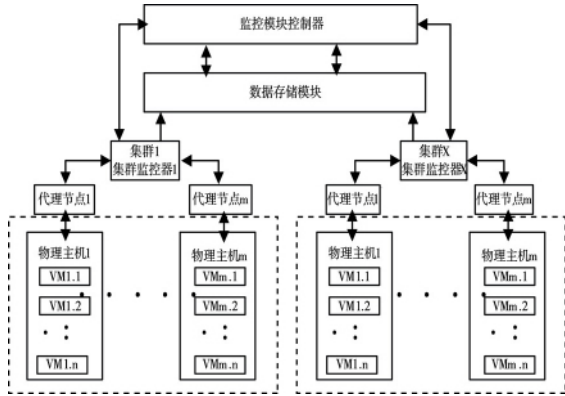


图2 实时自动化监控模块架构图

固定周期轮询的监控方式,能够监控到稳定不间断的有效数据,但是它灵活性不足,不能根据实际的状况调整监控周期,从一定程度上来说,消耗了资源。而事件驱动方式的监控能够在任务多、资源利用率比较大的情况下,及时触发,不会因为轮询次数过多而引起资源阻塞,为了保证监控系统的可靠性运行,本文采用两者结合的监控方式进行监控。

在仿真过程中,本文将 R_{status} 作为代表全局物理主机节点资源总使用状态的参数,将 C_{status} 作为代表集群监控器的资源总使用状态的参数。 Max_{status} 作为物理主机节点的资源总使用状态的最大值, Min_{status} 为最小值,则在 t 时刻,物理主机节点状态和集群监控器中的状态关系为

$$\Delta_{status} = \frac{|R_{status} - C_{status}|}{(Max_{status} - Min_{status})} \leq \rho \quad (1)$$

其中,资源状态信息误差的最大容系数为 ρ 。 ρ 越小,代表事件驱动下的更新监控数据越快, ρ 越大,代表事件驱动下的更

新监控数据就越平稳,此时发送资源监控信息就更适合采用轮询的方式。现给出本文中实现实时自动化监控模块的时序图为图3所示。

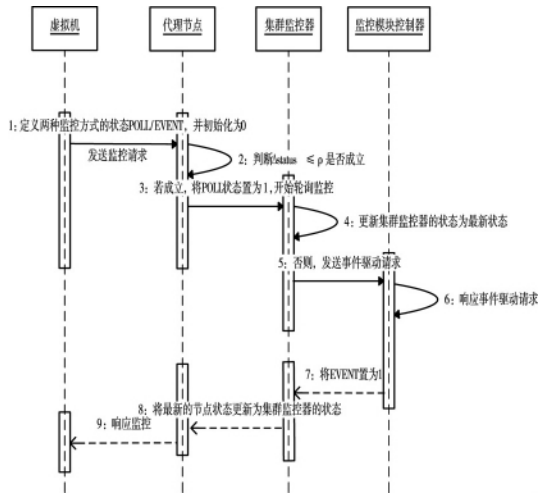


图3 自动化监控模块时序图

整个流程经过9步:

步骤1: 虚拟机发送监控请求,针对虚拟机的状态定义两种监控方式的状态关键字为POLL/EVENT,并将初始状态初始化为0;

步骤2: 请求传送到代理节点,并判断 $\Delta_{status} \leq \rho$ 是否成立;

步骤3: 若成立,将POLL状态置为1,开始轮询监控方式,并将监控信息存储到数据存储模块;

步骤4: 更新集群监控器的状态为最新状态,即将发过来的集群监控器的状态变成最新状态 C_{status} ;

步骤5: 否则,发送事件驱动的监控请求;

步骤6: 监控模块的控制器响应事件驱动请求;

步骤7: 集群监控器接收到来自监控模块控制器的信号后,将EVENT置为1;

步骤8: 开启事件驱动监控方式,将最新节点状态更新为集群监控器状态;

步骤9: 代理节点响应监控,并将信号传给虚拟机,将监控数据存储到数据存储模块。

关于 ρ 系数的确定,可以遵从以下规则:

根据 ρ 的大小,监控策略会随之转变,具体如下:

1) 当 ρ 较大时,可以采用周期性的轮询方式进行资源监控,而此时的集群资源监控系统只需按照这种方式进行不断的实时更新数据信息即可。事件驱动方式只有在集群监控器所获取到的数据资源信息状态波动大的时候触发,此时,集群监控器会以此方式读取物理节点的监控数据。

2) 当 ρ 较小时,则主要采用事件驱动监控方式作为监控策略, ρ 越小,代表资源状态信息对于资源的准确度要求越高,即误差的容忍度越低,也会对资源利用率的变化越敏感,

只要当资源利用率发生变动,哪怕变动很小时,集群监控器也会收到监控模块传来的状态信息。当资源利用率的 Δ_{status} 小于等于 ρ 时,监控器将周期性地发送资源监控数据,避免物理主机的状态信息在长时间内不能实时更新。

在本文中,根据作为实验对象的数据中心集群服务器的监控状况,取 ρ 的参考值为0.6,在不同规模的数据中心和集群规模中, ρ 的取值要根据监控虚拟机的状态的数据而定,此处不再赘述。

3.3 能耗模型和基于神经网络的节能策略

数据中心的能耗主要是各服务器计算节点的能耗,主要包括虚拟机CPU、磁盘存储、网络接口和内存四个部分。与系统资源相比,CPU和内存是计算节点能源消耗核心部分。而根据计算系统运行原理,CPU能耗的增加必然导致内存能耗的增加;磁盘存储的能耗变化则是一定的变动区间内,磁盘一旦使用,必然会有能源消耗;而网络的能耗的产生则主要是由网络交换设备的运转而产生的,在实际工程的资源分配策略中,主要考虑CPU和内存能耗,把其它的相关因素作为基础常量考虑。

CPU利用率 U_{cpu} 与整个能耗系统的负载成正比的,所以本文将CPU利用率 U_{cpu} 、内存利用率 U_{mem} 和能耗用正相关函数来计算总的能耗值。式(2)是利用CPU利用率 U_{cpu} 来得到服务器功耗 P 的线性函数。

$$P \propto P_0 + P_\alpha * U_{cpu} + P_\beta * U_{mem} \quad (2)$$

其中 P_0 是常数,代表除 U_{cpu} 和 U_{mem} 之外的相关因素的基础常数, P_α 代表 U_{cpu} 对能耗的影响系数, P_β 代表 U_{mem} 对能耗的影响系数。如图2中,数据中心中有 $m * n$ 台虚拟机,则总的功率为

$$P_{\text{总}} \propto m * n * P_0 + P_\alpha * U_{cpu} + m * n * P_\beta * U_{mem} \quad (3)$$

则

$$E = \int_{t_0}^t P_{\text{总}} \quad (4)$$

针对以上能耗模型,对数据中心的能耗模型的参数进行求解,本文采用神经网络RBM算法。

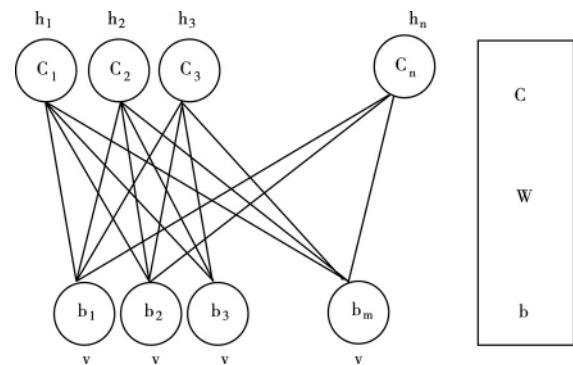


图4 RBM算法模型图

图4描述了2层的RBM算法模型, C 和 b 分别代表的

是可见层(visible) 和隐层(hidden) 数据是从可见层输入的 , 通过计算可以得出隐层的值。简单 RBM 模型的参数描述如下 ,层与隐层之间的权值是 W_{nm} ,图中 h 层有 n 个节点 v 有 m 个节点 ,单个节点用 v_j 和 h_i 描述 ,所以 $\forall i, j, p_j \in \{0, 1\}$, 这里的 $\{0, 1\}$ 是一个集合 ,这个集合中仅仅只有 2 个元素 ,即只能取 0 或者 1 b 是隐层(hidden) 的偏置 c 是可见层(visible) 的偏置。

如果一个训练样本是 $X = (x_1, x_2 \dots x_m)$,每个分量对应着可见层(visible) 层的一个节点 ,也就是可见层(visible) 层有 m 个节点 ,样本通过可见层(visible) 层输入进去 ,经过一系列运算得到一个向量 $Y = (y_1, y_2 \dots y_n)$,那么这组值同理也是与隐层(hidden) 层的每个节点一一对应。即可以理解为 RBM 模型将一个维度为 m 的向量(即一个样本有 m 个特征) 映射到一个维度为 n 的向量(一个样本具有 n 个特征) 。

RBM 更新权值和偏执的算法过程如下:

由 V_0 计算 H_0 ,

先计算出每个 h_{0i} 对应的 $p(h_{0i} = 1 | v_0)$

$$p(h_{0i} = 1 | v_0) = \text{sigmoid}(c_i + \sum_{j=1}^m w_{ij} v_{0j})$$

再对该值求 *simple* 运算即为 H_0 ,即产生一个 0 ~ 1 的随机数 ,若 $p(h_{0i} = 1 | v_0)$ 大于这个随机数 ,则最终值 h_{0i} 取 1 ,否则取 0。

式中 c_i 是隐层(hidden) 层的偏执。由 H_0 计算 V_1 ,计算出每个 v_{1j} 对应的 $p(v_{1j} = 1 | h_0)$

$$p(v_{1j} = 1 | h_0) = \text{sigmoid}(b_j + \sum_{i=1}^n w_{ij} h_{0i})$$

再对该值求 *simple* 运算即为 V_1 ,即产生一个 0 ~ 1 的随机数 ,若 $p(v_{1j} = 1 | h_0)$ 大于这个随机数 ,则最终值 v_{1j} 取 1 ,否则取 0。

式中 b_j 是可见层(visible) 层的偏执。

由 V_1 计算 H_1 ,计算出每个 h_{1i} 对应的 Y 中的 $p(h_{1i} = 1 | v_1)$

$$p(h_{1i} = 1 | v_1) = \text{sigmoid}(c_i + \sum_{j=1}^m w_{ij} v_{1j})$$

此值不做 *simple* 运算 ,直接即为 H_1 。式中 c_i 是隐层(hidden) 层的偏执。

最后 ,更新权值 w 和 2 个偏执 b, c

$$w \leftarrow w + \text{lr}(h_0 v_0 - p(h_1 = 1 | v_1) v_1)$$

$$b \leftarrow b + \text{lr}(h_0 - p(h_1 = 1 | v_1) v_1)$$

$$c \leftarrow c + \text{lr}(v_0 - v_1)$$

此过程中 V_0, V_1 为可见层(visible) 层的值 V_0 为最初的输入值 V_1 相当于方向训练得出的值 H_0, H_1 为隐层(hidden) 层的值 H_0 为正向得出的值 H_1 为反向得出的值。

3.4 算法实现

拟合函数的算法核心实现为:

核心部分代码的实现具体如下:

// 定义前向由 V_0 计算 H_0 的函数

```
void RBMLayer::Forward_v0_h0( int iter) {
    string item = ROUND_PATH + int2string( iter + 1) ;
    // 如果算法层的运算带有偏执 bias
    if ( rbmp_. flag_bias)
    {
        bbias_ - > writeData( file0_bias , bbias_ - > count_);
        cbias_ - > writeData( file0_cbias , cbias_ - > count_);
    }
    for ( int i=0; i < hidden_units_0 - > count_; i + +)
    {
        for ( int j=0; j < visible_units_0 - > count_; j + +)
        {
            hidden_units_0 - > v_data. at( i) = visible_units_0 - > v_
            data. at( j) *
            weight_ - > v_data. at( weight_ - > offset( 0 , i , j) ) ;
        }
        if ( rbmp_. flag_bias)
        {
            hidden_units_0 - > v_data. at( i) + = bbias_ - > v_
            data. at( i) ;
        }
    }
    hidden_units_0 - > writeBatchFile( file0_outp) ;
    // 进行 sigmoid 运算
    sigmoidLayer* sig = new sigmoidLayer( ) ;
    for ( int i=0; i < hidden_units_0 - > count_; i + +)
    {
        hidden_units_0 - > v_data. at( i) = sig - > sigmoid( hidden
        _units_0 - > v_data. at( i) ) ;
    }
    hidden_units_0 - > writeBatchFile( file0_outp_sig) ;
    // 进行 sample 运算
    sample( hidden_units_0 , hidden_units_0) ;
    hidden_units_0 - > writeBatchFile( file0_outp_sig_sam-
    ple) ;
    hidden_units_0 - > writeBatchFile( h0) ;}
```

4 实验及结果

4.1 实验平台

本文实验环境采用 Intel Core 4590 处理器 ,内存 8G ,采用虚拟集群建立的方式 ,Linux 内核版本为 2. 6. 32 - 220. e16. 86_64 ,监控性能计算器的工具是 perf - 2. 6. 32 ,编译环境为 gcc - 4. 4. 6。

4.2 实验过程及结果

为了有效采样 ,均对整数型和浮点型 Benchmark 进行运算 ,均进行运行 1000 遍 ,采集对与虚拟机能耗的降低率。实

验统计结果如表 1 所示。从表 1 中可以看出,该模型在计算整数型和浮点型 benchmark 时均有能耗节省的效果,当计算相对复杂的浮点型数据的时候,效果更好。

表 1 实验结果

计算类型	运行次数	正确率	能耗节省率
整数型 Benchmark	1000	100%	35%
浮点型 Benchmark	1000	100%	48%

5 结束语

本文主要针对目前数据中心资源分配策略中的监控资源方法单一、灵活性有限的问题,提出一种基于神经网络模型的数据中心节能策略方法,本方法不仅可以提供自动化的监控模块和能耗模型,还能提供基于神经网络模型的节能策略,经过试验具有良好的能耗节省率。但是在能耗模型的建立中仅考虑 CPU 能耗和内存能耗,并未考虑网络等能耗问题。若建立更接近真实情况的数据中心能耗模型,应尽可能多的考虑影响因素,这亦可作为下一步的研究方向。

参考文献:

[1] J Misra ,I. Saha. Artificialneural networks in hardware: A survey of two decades of progress[J]. Neurocomputing ,2010 ,74

(1 - 3) : 239 - 255.
[2] H Esmailzadeh , A Sampson , L Ceze , et al. Neural Acceleration for General - PurposeApproximate Programs [C]. In 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture , 2012: 449 - 460.
[3] 陈云霄. 体系结构研究者眼中的神经网络硬件[J]. 中国计算机学会通讯 ,2015 (7) .
[4] 刘江. 数据中心计算资源节能算法研究[D]. 电子科技大学 ,2013.
[5] 雷成军 ,罗亮 ,吴文俊. 基于云计算的能耗监控与节能方法研究[J]. 计算机应用与软件 ,2011 (11) .
[6] Paul A Merolla , John V Arthur , Rodrigo Alvarez - icaza , et al. A million spiling - neuron interatedcircuit with a scalable communication network and interface[J]. Science ,2014 ,345(6197) .



【作者简介】

智伟威(1991 -) ,女(汉族) ,山东人 ,硕士研究生 ,工程师 ,主要研究领域为计算机算法、新一代数据中心节能等。

周新星(1980 -) ,男(汉族) ,湖南人 ,硕士研究生 ,高级工程师 ,主要研究领域为云数据中心规划实施

与运营管理。

(上接第 229 页)

对相应的待隐藏数据信息进行获取 ,为今后的研究提供了良好的理论基础与数据信息。

参考文献:

[1] 王兴 ,邵艳明 ,杨波 ,等. 基于激光雷达与红外数据融合的跟踪算法[J]. 红外技术 ,2019 ,41(10) : 947 - 955.
[2] 陈星 ,李战武 ,胡晓东. 杂波环境下雷达/红外数据融合目标跟踪算法的研究[J]. 小型微型计算机系统 ,2019 ,40(08) : 1794 - 1798.
[3] 查宇飞 ,吴敏 ,库涛 ,等. 基于位置敏感模型的深度跟踪算法研究[J]. 电子学报 ,2019 ,47(10) : 2076 - 2082.
[4] 宋珂慧 ,张莹 ,张江伟 ,等. 基于生成式对抗网络的结构化数据表生成模型[J]. 计算机研究与发展 ,2019 ,56(9) : 1832 - 1842.
[5] 田慧明 ,吴成茂 ,田小平. 基于混沌理论和整数变换的可逆信息隐藏[J]. 计算机工程与应用 ,2019 ,55(12) : 194 - 201.
[6] 赵雪青 ,师昕. 基于视觉颜色感知——光学相似的图像去雾方法[J]. 计算机应用研究 ,2019 ,36(9) : 2845 - 2848.

[7] 于静 ,韩鲁青. 一种改进的求解支持向量机模型的坐标梯度下降算法[J]. 系统科学与数学 ,2018 ,38(5) : 583 - 590.
[8] 裴亮 ,刘阳 ,高琳. 结合全卷积神经网络与条件随机场的资源 3 号遥感影像云检测[J]. 激光与光电子学进展 ,2019 ,56(10) : 261 - 267.
[9] 王丽雯 ,黄旭. 大数据分析下不完备数据多重准确填补仿真[J]. 计算机仿真 ,2019 ,36(7) : 367 - 370.
[10] 温宗周 ,程少康 ,李丽敏 ,等. 改进粒子滤波与均值漂移特征融合的目标跟踪[J]. 计算机工程与设计 ,2019 ,40(8) : 2306 - 2311.



【作者简介】

金 秋(1980 -) ,女(汉族) ,山东曹县人 ,硕士 ,讲师 ,研究方向:网络安全。

林 馥(1978 -) ,男(汉族) ,湖南邵阳人 ,博士研究生 ,副教授 ,研究方向:网络安全、网络通信等。