# 1.Introduction

随着物联网和大数据技术的在环境领域的深入应用，累积大量空气质量监测数据，基于人工智能的机器学习模型不断被提出，其中以神经网络为主的网络模型在空气质量预测中得到了广泛的应用(Gao et al., 2018;Nieto et al., 2018;Park et al., 2018)，尤其是基于深度学习的预测模型(LeCun et al., 2015; Schmidhuber et al., 2015)可以免除传统机器学习的特征工程过程，更好的揭示空气质量复杂非线性机理，成为当前空气质量预测研究的热点。目前空气治理的主要任务是强化多污染物协同控制和区域协同治理，加强细颗粒物和臭氧（O3）协同控制。

多任务学习(MTL)的目标是同时学习多个相关预测任务,它比单个任务学习具有更强的泛化能力。多任务学习是指对于给定的模型，利用多个任务之间的相关性，对多个任务进行联合训练，来更好地概况我们的原始任务，通过共享表示从而提高模型在多个任务上的学习能力。基于深度学习的多任务学习最早可追溯到1997年，Caruana, 1993提出了神经网络中硬参数共享的结构，同时基于实证研究提出了多任务学习起作用的几个可能的机制。与多任务学习相结合的深度学习框架主要有下面两种方法：（1）基于硬约束的多任务深度学习方法。硬共享是目前应用最为广泛的共享机制，它把多个任务的数据表示嵌入到同一个语义空间中，再为每个任务使用任务特定层提取任务特定表示。大多数硬参数共享网络通过共享隐层来学习特征的共享表示，在输出端为各任务设计单独的分类器预测输出层,例如Collobert and Weston（2008）将词性标注、词块分割、命名实体识别及词语相似度任务统一到一个语言模型中，利用其他任务中自动学习的特征 来 提 升 语 义 角 色 标 注 任 务 的 性 能。Collobert and Weston 2008; Subramanian et al. 2018; Liu et al. 2019使用硬共享方法将特定于任务的层堆叠在共享层的顶部。（2）基于软约束的多任务深度学习方法。参数的软共享机制，可以不用考虑任务的相关性，不需要对任务相关性做任何假设，但由于为每个任务分配一个网络，常常需要增加很多参数，每个任务都有自己的模型和参数。然后对模型参数之间的距离进行正则化，以鼓励参数相似，例如Long et al., 2015使用L2距离进行正则化，而Yang et al., 2017使用轨迹范数,同时MTL正则化约束也给了软参数共享很大的启发。

目前，在深度多任务学习中，还存在一定的不足，在联合训练多个任务时,会产生知识负迁移现象，使得多个任务学习时，任务之间互相干扰，加大损失(王进 et al., 2022)，基于此，新的方法应该是如何极力减小负迁移现象的产生。我们从多任务的联合损失出发，提出一种自适应节点的多任务学习方法，其创新点是:将单任务与多任务相结合，在多个任务联合训练达到一定的条件时，将符合条件的节点复制，使得一部分多任务训练变成单任务训练，在享受了多任务联合训练带来的好处的同时，进一步优化每一个任务，确保了任务之间可以动态地改变它们所连接的节点数，从而动态地更新网络结构，极大地减少了负迁移所带来的影响，使得整体预测效果更好，代价仅仅是少量的时间。

本文结构如下：第 2 章将具体介绍本文的新模型框架，理论价值意义以及其具体的算法。第3章是模拟。实例在第 4 章。第 5 章是理论证明。最后第 6 章是总结。

# 2. Models and Algorithms

## 2.1 自适应节点的硬参数共享网络架构

本文在硬参数共享网络结构的基础上进行改进，研究了一种自适应节点的多任务深度学习方法，其网络架构主要包括4个部分: ①输入层②共享参数层1; ③自适应节点层；④输出层。其中硬参数网络结构如图1所示，新模型的整体架构图如图2所示。网络结构一共包括了4层，分别为输入层，共享层，自适应节点层以及输出层。

Lk

L1

共享层1

共享层2

输出层

…

…



W (1)

W (2)

…

*X*

输入层

W (0)

图1 硬参数网络结构

Lk

L1

共享层1

自适应节点层

输出层

…

…



W (1)

W (2)

…

*X*

输入层

W (0)

图2 新模型的整体架构

## 2.2 自适应节点算法

假设空间足够大，可以同时包含多个任务的解；假设有k个任务，输出层共有k个节点，分别记为，输入层共有z个节点，分别记为，共享层1一共有m个节点，分别记为，在最初的自适应节点层一共有j个节点，分别记，记输入层到共享层1的权重矩阵为，其中,偏置矩阵为;记共享层1到自适应节点层的权重矩阵为，其中。记到的参数记，到的参数记为，以此类推，偏置矩阵为;自适应节点层到输出层的参数矩阵为，偏置矩阵为。

为了简单起见，我们以两个任务为例来阐述我们算法的思想，对于两个任务的整体损失函数为：



在反向传播过程中，对模型参数求梯度如下:,当,,时，本文用Adam算法更新参数(Kingma and Ba，2015),更新参数







以此类推，其中代表学习率，其中代表经过矫正后过往梯度与当前梯度的均值，代表经过矫正后过往梯度与当前梯度平方的均值，是为了维持数值稳定性而添加的常数，和代表的是指数衰减率。

此时对于任务 ，由于需要优化的梯度方向与整体多任务需要优化的梯度方向相同，损失函数减小，对于任务，由于需要优化的梯度方向与整体多任务需要优化的梯度方向相反，损失函数增加。对于整体多任务，根据Adam算法原理，整体多任务损失函数减少。

反之，当时，同理，对于任务，损失函数增加；对于任务，损失函数减少；整体损失函数减少。其它相似情况同理。

对于，它们两项单任务梯度的符号与相同，这将在第五章中给出证明。

剔除其它参数对于损失函数变动的影响，我们将节点 复制，新节点为， 与所相连接的参数记为,我们令的初值等于。

重新更新参数。



以此类推，此时为任务的单任务节点，为任务的单任务节点。

本文中的单任务节点是指该自适应节点层的节点仅仅与输出层的某个任务的节点相关的节点，例如，是指它们分别只与输出层中的和相连，但仍然都与共享层1中的所有节点为全连接的关系。

此时，假设新模型的整体损失函数，对于任务，的损失函数对的梯度为大于原模型中整体多任务需要优化的梯度，则损失函数更大幅度减小;对于任务，需要更新的梯度与原先的整体多任务无关，即对于任务2本身，根据Adam算法原理，损失函数依然减小。则整体损失函数更大幅度减小。

本文采用Mini-batch方法训练网络,算法步骤如下:

Remark:这里以两个任务为例，多个任务也与之相似。

步骤1：整体数据集按70%和30%的比例分别随机划分为训练集和测试集，将训练集中30%的数据作为验证集，用于选择出最优的模型。设置迭代次数I、每批次数据中预热训练次数阈值（在迭代过程中的最低迭代次数值），连续梯度异号阈值（在迭代过程中单任务的梯度方向连续互为反方向的次数）等超参数; 随机初始化共享层权重矩阵和偏置矩阵

其中设置预热训练次数阈值是为了防止模型过早的进行节点复制，避免误判了全局最优点的位置。设置连续梯度异号阈值是为了防止模型所找的复制节点为局部最优点，它可以更准确地帮助新网络模型找到全局最优点的位置。

步骤2:训练集上对进行2个任务进行联合学习，通过网络前向传播，得到输出值和损失函数值:





其中, 为输入的样本数据，输入层到共享层1，共享层1到自适应节点层，以及自适应节点层到输出层的激活函数全部设置为Sigmoid函数。接着通过误差反向传播，采用Adam算法更新输入层，共享层1和自适应节点层的参数、、、、、。

步骤3:设置T轮迭代,分别计算损失函数对共享层1到自适应节点层的权重矩阵的梯度，若损失函数对其权重的梯度方向与单任务损失函数对其权重的梯度方向相反，经过t轮迭代仍然如此，若，且总迭代次数 ，我们将自适应节点层中与其梯度相反所对应的节点复制为，将其与共享层1中所有节点相连，不妨设任务或者其中一个的损失函数中对原先的梯度＞0，整体的损失函数L对原先的梯度＞0，则将另外一个损失函数中对原先的梯度＜0的任务所连接的自适应节点层的节点的网络线拆连到所对应复制的节点，令 与所相对应的参数记为，我们令的初值等于，利用Adam算法（Kingma和Ba，2015）重新更新。使得两个任务在学习到大量的共享信息之后，又进一步降低了它们的联合损失，减少了负迁移现象对模型带来的影响；

步骤4:当第一次复制完节点，更新完权重参数矩阵后，用网格搜索法寻找最优的预热训练次数阈值，连续梯度异号阈值等超参数，重复步骤3，看模型是否满足复制第二次节点的条件，若满足，则重复步骤3的操作，复制第三次节点；不满足则停止迭代，以此类推。

步骤5:在测试集上评估模型性能，RMSE等相关评价指标。

## 2.3 模型损失函数

在本文的多任务深度神经网络模型回归问题中，采用均方根误差(root mean square error,RMSE) 作为评价指标，其计算公式为:



# 3.Simulation

模型： 

模拟1:输入的为服从最大值为8，最小值为0的均匀分布的n\*5维随机数，为服从最大值为4，最小值为0的均匀分布的5\*4维随机数， 为服从最大值为4，最小值为0的均匀分布的4\*3维随机数，为服从最大值为4，最小值为0的均匀分布的3\*2维随机数，噪音为服从均值为0，方差为1的正态分布的n\*2维随机数，为n\*2维矩阵。

模拟2:输入的为服从标准正态分布的n\*5维随机数，为服从标准正态分布的5\*4维随机数，为服从标准正态分布的4\*3维随机数，为服从服从标准正态分布的3\*2维随机数，噪音为服从均值为0，方差为0.2的正态分布的n\*2维随机数，输出为n\*2维矩阵。

模拟3:输入的为服从参数为10的指数分布的n\*5维随机数，为服从标准正态分布的5\*4维随机数，为服从标准正态分布的4\*3维随机数，为服从标准正态分布的3\*2维随机数，噪音为服从标准正态分布的n\*2维随机数，输出为n\*2维矩阵。

本文构建的新模型算法程序，是在Python语言环境中，基于TensorFlow框架实现的(Gonçalves等,2016)。经实验，设置输入层的神经元数为5，共享层的神经元数为4，最初的自适应节点层神经元数为3，输出层的神经元数为2。Adam 优化算法学习率=0.001，指数衰减率=0.9 ，=0.9999 。

对于模型1，模型2，模型3，经过调参可以得到最优的超参数如下:

表1 模拟数据的超参数表

|  |  |  |  |
| --- | --- | --- | --- |
| 超参数 | 均匀分布 | 标准正态分布 | 指数分布 |
|  | 6 | 6 | 6 |
|  | 6 | 7 | 5 |
|  | 4 | 8 | 8 |
|  | 50 | 50 | 50 |
|  | 20 | 20 | 25 |
|  | 10 | 20 | 15 |
|  | 570 | 510 | 530 |

其中每个超参数的含义如下:

-未进行节点复制时原网络进行第一次复制节点的连续梯度异号阈值。

-第一次复制节点后的模型到第二次进行复制节点的连续梯度异号阈值。

-第二次复制节点后的模型到第三次进行复制节点的连续梯度异号阈值。

-未进行节点复制时原网络进行第一次复制节点的预热训练次数阈值

-第一次复制节点后的模型到第二次进行复制节点的预热训练次数阈值。

-第二次复制节点后的模型到第三次进行复制节点的预热训练次数阈值。

I – 新模型的累积总迭代训练次数

对于上述模型分别设置与硬参数共享网络结构的对比实验，为保证对比的有效性，硬参数共享网络结构的模型总迭代次数与新模型复制节点前和每次复制节点后的累加总迭代次数相等。

对于上面三种模拟模型，分别取样本量n为1000，2000和3000。整体数据集按70%和30%的比例分别随机划分为训练集和测试集，将训练集中30%的数据作为验证集，用于选择出最优的模型。测试集用于评估模型性能和稳健性，并采用100轮结果的均值作为实验结果。

本文采用均方根误差(root mean square error,RMSE) 作为评价指标，

表2 100次模拟RMSE的均值()

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | n=1000 | | n=2000 | | n=3000 | |
|  | 硬参数 | 新模型 | 硬参数 | 新模型 | 硬参数 | 新模型 |
| 均匀分布 | 0.314 | 0.224 | 0.284 | 0.184 | 0.258 | 0.142 |
| 标准正态分布 | 1.604 | 1.382 | 1.230 | 1.038 | 0.894 | 0.764 |
| 指数分布 | 0.627 | 0.483 | 0.554 | 0.436 | 0.526 | 0.418 |

表2分别为模型1，模型2，模型3在新网络模型和硬参数共享模型上的对比实验结果。表中的数值为RMSE值，结果越小越好。随着样本量的增大，新网络模型和硬参数共享模型的RMSE 值越来越小，效果越来越好。对于样本量为1000的情况，相对于硬参数共享模型，新网络模型在3个模拟数据上的性能分别平均提升了28.7%(均匀分布)、13.8%（正态分布）、22.9%（指数分布）；对于样本量为2000的情况，相对于硬参数共享模型，新网络模型在3个模拟数据模型上的性能分别平均提升了35.2%(均匀分布)、15.6%（正态分布）、21.3%（指数分布）；对于样本量为3000的情况，相对于硬参数共享模型，新网络模型在3个模拟数据模型上的性能分别平均提升了44.9%(均匀分布)、14.5%（正态分布）、20.5%（指数分布）。

表3 1000个样本100次模拟的平均耗时表

|  |  |  |
| --- | --- | --- |
|  | 硬参数 | 新模型 |
| 均匀分布 | 85s | 92s |
| 标准正态分布 | 97s | 102s |
| 指数分布 | 89s | 93s |

表3表示模拟数据的耗时表，在耗时方面，就1000个样本而言，新网络模型在3个模拟数据上的平均每轮模拟耗时为92s(均匀分布)，102s(标准正态分布)，93s(指数分布)，使用硬参数网络模型的平均每轮模拟耗时为85s（均匀分布），97s(正态分布)，89s(指数分布)。不难看出，新网络模型的整体性能优于硬参数共享模型，说明了新网络模型相较于原模型具有良好的预测和估计能力，代价仅仅是少量的时间。

下面是样本量为1000，模拟数据分别为均匀分布、标准正态分布和指数分布的新模型与原模型的RMSE变化对比图，由于在复制节点之前新模型和原模型的RMSE的十分接近，在模拟中不展示复制节点前新模型和原模型的RMSE变化过程，直接从新模型第一次进行节点复制开始展示。其中新模型中的三个标记点依次代表新模型第一次进行节点复制，第二次进行节点复制和第三次进行节点复制。原模型和新模型的最后两个标记点代表损失函数的最低值点。由图中可以看出在迭代过程中新模型的RMSE几乎一直低于比原模型的RMSE值，说明了新模型的的良好性能。

图3 模拟数据为均匀分布的新模型与原模型的RMSE变化对比图

图4 模拟数据为标准正态分布的新模型与原模型的RMSE变化对比图

图5 模拟数据为指数分布的新模型与原模型的RMSE变化对比图

# 4. Example

本节考虑南京市空气污染数据，该数据来源于中国气象数据网和真气环境大数据中心。数据包含了南京市2013年12月2日至2022年8月28日的空气质量数据(AQI)、PM2.5浓度()、PM10浓度()、CO浓度()、SOs浓度()、NO2浓度()、臭氧8小时O3\_8h()这个7变量，其中PM2.5浓度和臭氧8小时分别为输出变量和。PM10浓度、CO浓度、SOs浓度和NO2浓度分别为输入变量,,和，样本量n为3191个。具体考虑输入变量,,和来估计未来两个任务和的输出值。

空气污染预测模型： 

其中输入是n\*4维矩阵，为4\*4维矩阵， 为4\*3维矩阵，为3\*2维矩阵。

对于此预测模型，经过调参可以得到最优的超参数如下:

表4 实例的超参数表

|  |  |
| --- | --- |
| 超参数 | 数值 |
|  | 15 |
|  | 8 |
|  | 7 |
|  | 70 |
|  | 20 |
|  | 20 |
|  | 300 |

将南京市的空气污染数据分别放入新网络模型和原网络模型，且估计效率的衡量仍采用RMSE，结果如表5所示，对比新模型和旧模型的RMSE可知，新网络模型的性能要比旧模型的效果提升9.1%。

表6是南京市的空气污染数据放入不同模型的耗时表，新模型的时间为321s，原模型的时间为310s，新模型仅仅比原模型多了11s。

表5 两种模型的RMSE值(10-2)

|  |  |
| --- | --- |
|  | RMSE |
| 原模型 | 1.3829 |
| 新模型 | 1.2584 |

表6 两种模型的耗时

|  |  |
| --- | --- |
|  | RMSE |
| 原模型 | 321s |
| 新模型 | 310s |

图6 南京市空气污染数据新模型与原模型的RMSE变化对比图

上图是南京市空气污染数据新模型与原模型的RMSE变化对比图，图中的标记点模拟数据时的标记点含义一样，不同的是，展示了在第一次复制节点前两个模型的RMSE变化趋势，实质上是大致相同。从图中可以看出对于南京市空气污染数据，在迭代过程中新模型的RMSE几乎一直低于比原模型的RMSE值，证实了新模型的的良好性能。

# 5. Proof

## 5.1梯度同号证明

在硬参数共享神经网络中，各层的激活函数选择sigmoid函数。试证明任意隐藏层中任意节点所连接的上一层各个权重，在反向传播更新参数时对于各个任务损失函数的梯度方向相同。

设总层数为n的硬参数共享网络中，共k个任务， 表示第i层中j(i)个节点的输出向量, 表示第i-1层中连接第i层第P个节点的权重向量。

在模型正向传播过程中



损失函数：



由于sigmoid函数值域为（0，1），故对于任意i和j有，i = 1，2…n, j =1，2…ji。

在反向传播过程中，

根据分母布局链式求导法则，第d层第p个节点 所连接的上层权重为（j(d-1)为d-1层网络的节点数）对于任务i损失函数Li的梯度有：



不妨令



因此,，得证。

Remark：该证明仅仅只针对sigmoid激活函数，对于值域包含于 或 的激活函数同样适用，如Relu、Softplus等。

## 5.2新模型拥有更低的损失函数

本文以两个任务为例，其中第n层为输出层，第n-1层为自适应节点层，设假设损失函数与参数的关系在一定范围内是单调不变的。

对于原模型整体损失函数。

表示第n-1层连接到第n层的权重矩阵，其中表示第n-1层中连接第n层中第1个节点的权重向量，第n-1层的节点数为，自适应节点层的节点向量记为 。

记整体两个任务的损失函数对n-2层中连接第n-1层中第i（其中）个节点的参数向量的梯度为

 (1)

记任务的损失函数对参数向量的梯度为

 (2)

记任务的损失函数对参数的梯度为

 (3)

当(1)＞0,(2)＞0,(3)＜0时.

更新关于整体两个任务的参数

 (4)

因(1)＞0，向负梯度方向迭代，值减少，根据梯度下降算法原理整体损失函数L下降。

对于任务，因(2)＞0，该任务损失函数L1对参数的更新方向与整体损失函数L对参数的更新方向相同，故L1下降。

对于任务y2，因(3) ＜0，该任务损失函数L2对参数的更新方向与整体损失函数L对参数的更新方向相反，故L2上升。

因，不妨设在此轮迭代中,下降了个单位，上升了个单位，则下降了个单位。

经过复制节点，新节点为仅与任务y2相关的单任务节点，原节点为只与任务y1相关的单任务节点。且使得==，对于新模型的整体损失函数。

记为仅与任务相关的参数向量，的损失函数对的梯度为

 (5)

记为仅与任务相关的参数向量，的损失函数对的梯度为

 (6)

更新关于任务的参数

 (7)

原模型中损失函数对参数向量的梯度为



将(1)与(5)对比,其中



对于（2）中



故(1)中的第一项与(5)的值相等,又因（3）<0，故,下降（）个单位。

更新关于任务的参数

 (8)

因，故的值增加，根据梯度下降算法原理，下降。

不妨设下降了个单位。

因，故下降了 个单位。

故下降的单位多于下降的单位。

当(1)＞0,(2)＜0,(3)＞0;(1)＜0,(2)＜0,(3)＞0;(1)＜0,(2)＞0,(3)＜0。的时候同理可证得下降的单位多于下降的单位。

Remark:此结论使用的优化算法也可以是其它优化算法（例如:adam算法等），不仅仅局限于梯度下降法。

# 7.References

M.Gao,L.Yin,J.Ning,Artifificial neural network model for ozone concentration estimation and Monte Carlo analysis,Atmospheric Environment 184(2018) 129-139.

P.J.G.Nieto,G.-G.E,A.B.S，anchez,A.A.R.Miranda,Air quality modeling using the PSO SVM-based approach,MLP neural network,and M5 model tree in the metropolitan area of Oviedo(Northern Spain),Environmental Modeling and Assessment 23(3)(2018)229-247.

S.Park,M.Kim,M.KIM,H.-G.Namgung,K-T.Kim,K.H.Cho,S.-B.Kwon,Predicting PM10 concentration in Seoul metropolitan subway stations using artifificial neural network(ANN),Journal of Hazardous Materials 341(2018)75-82.

Y.LeCun,Y.Bengio,G.Hinton,Deeplearning,Nature 521(7553)(2015)436-444.

J.Schmidhuber,Deep learning in neural networks:an overview,Neural Networks 61(2015)85-117.

Caruana R A. Multitask Learning: A Knowledge-Based Source of Inductive Bias[J]. Machine Learning Proceedings, 1993, 10(1):41-48.

Long D, Cohn T, Bird S, et al. Low Resource Dependency Parsing: Cross-lingual Parameter Sharing in a Neural Network Parser[C]// Meeting of the Association for Computational Linguistics and the, International Joint Conference on Natural Language Processing. 2015:845-850.

Yang, Y. et. al. 2017. Trace Norm Regularized Deep Multi-Task Learning. ICLR2017 workshop.

Collobert R,Weston J.A unified architecture for natural language processing:Deepneural networks with multitask learning[C]Proceedings of the 25th international conference on Machine learning-lCML,08,New York:ACM Press,2008:160-167.

Liu, X.; He, P.; Chen, W.; and Gao, J. 2019. Multi-taskdeep neural networks for natural language understanding. In ACL, 4487–4496.

Subramanian, S.; Trischler, A.; Bengio, Y.; and Pal, C. J.2018. Learning general purpose distributed sentence representations via large scale multi-task learning. In ICLR.

王进,任超,舒雅宁,彭浩,闫振兴. 基于稠密连接注意力单任务提升的深度多任务学习[J]. 重庆邮电大学学报(自然科学版),2022,34(03):393-401.

Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[A]. International Conference on Learning Representations (ICLR), 2015: 1‒15．