基于LSTM+BPNN深度神经网络的新冠疫情预测模型

摘要：新冠肺炎疫情传播在全社会乃至全世界范围内造成了巨大影响。我们通过对互联网上面向开源及私有软件项目托管平台上所涌现的各类数据仓库进行调用、采集、汇总，利用长短期记忆人工神经网络（LSTM）模型对疫情期间人员流动历史数据进行解析和挖掘。预测近期内各个省市的人口迁入迁出规模。最后通过利用具有多层前馈结构的误差反向传播（BP）算法对处理过后的大数据模型进行细化、调参、优化，针对新冠肺炎可能爆发的未来时间地点难以预测等问题，达到创建出卓成有效的深度学习模型为目的，从而提出一种基于LSTM-BPNN深度神经网络的新冠疫情预测模型，以此为推动抗击疫情的发展贡献绵薄之力。

关键词：长短期记忆（LSTM）模型；误差反向传播（BP）算法；深度学习

1. 引言

社会价值新型冠状病毒肺炎(COVID-19)在全球范围内大范围传播流行,是全世界关注的公共卫生问题.本次疫情人员控制上,很多城市在人员的综合管理、疫情的实时管控、突发情况的处理等方面存在不足。但因我国强有力的疫情控制措施，目前疫情趋于稳定。然而，在疫情爆发早期，政策实施相对宽松，医疗资源相对缺乏，疫情传播扩散速度较快。本次疫情爆发与农历新年春运时间较吻合，且早期疫情传播受政策影响较小，因此，通过建立疫情早期流动人口数据与各地区疫情严重程度的模型对于疫情的检控预防具有积极的意义。

为了预测未来一些城市可能爆发的情况，我们首先采用时间序列预测的方法对不同城市的人口迁入迁出进行分析和预测。再在此基础之上用不同城市的已患病人数作为输出集，再用这些城市数天前的迁入迁出作为输入集。从而可预测未来所选城市的爆发情况。

目前传统的时间序列预测模型有移动平滑法、指数平滑法、趋势拟合法、自回归模型（Autoregressive model）、滑动平均模型（Moving Average Mode）、自回归条件异方差模型（Autoregressive conditional heteroskedasticity model）等等。但随着当下互联网核心科技不断的变化和迭代，在人工智能的应用技术高速发展下，人工神经网络在时间序列预测里的使用逐渐泛应曲当。而在这里面，长短期记忆（Long Short-Term Memory，LSTM）神经网络的实际运用不胜枚举。长短期记忆人工神经网络，它继承了循环神经网络（Recurrent Neural Network, RNN）所包涵的参数共享、记忆性和图灵完备等优点特性。与此同时，摒弃了长期以来传统循环神经网络的部分结构，针对存在的长期依赖问题而提出利用特殊隐式单元使网络储存增大设计，引入了细胞状态这一尤为重要的灵魂概念，正因如此，长短期记忆神经网络体内的细胞掌握着对不同状态去留的决定权。与此同时，我们结合BP神经网络非线性映射能力这一优点对已经被LSTM处理过的时间序列数据内部的繁琐机制进行求解，逼近非线性连续函数。该网络还具有强大的自学习和泛化能力，使得我们可以提取不同城市大量数据集中的信息特点和噪声污染，对其正确分类并记忆于网络权值中，提升我们网络的适应性和容错率，使得整张网络具有将训练成果运用到新环境的能力。

二.理论基础

本文对循环神经网络（Recurrent Neural Network，RNN）、长短期神经网络（Long Short-Term Memory，LSTM）自适应矩估计（Adam）优化器以及误差反向传播（Back Propagation,BP）神经网络进行概要说明。

2.1传统RNN与LSTM神经网络结构模型分析对比

在传统的RNN模型中，我们通过对一数据矩阵[x1,x2,x3,…,xn]进行预处理，并将其作为输入集传送入模型框架中,经过初始化的权值矩阵[u1,u2,u3,…un]时我们会将其输入矩阵x和矩阵w（w是上次数据处理后的隐藏层矩阵s）用计算方程式（1）~（2）得到我们新生的隐藏层数据矩阵s=[s1,s2,s3,…,sn]，矩阵s经过权值矩阵v=[v1,v2,v3,…,vn]和w=[w1,w2,w3,…,wn]通过计算方程式（3）后得到输出层矩阵o=[o1,o2,o3,…,on]。如图1

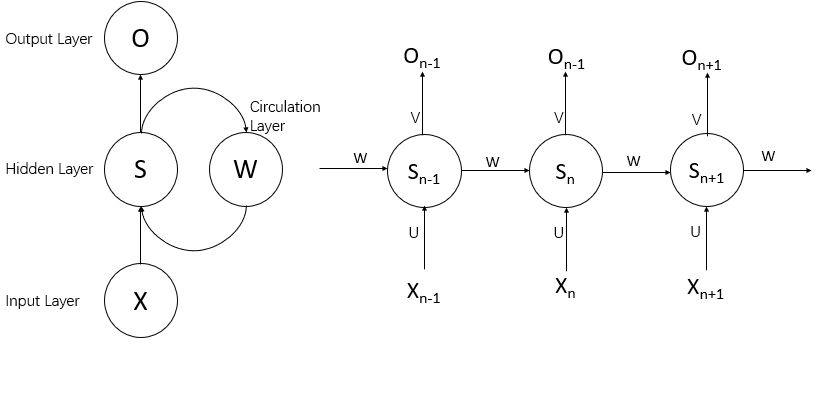


图1 RNN网络结构图及其展开

由此可见，传统RNN在理论上是利用过去的数据流信息与现如今的信息做一系列运算，如果是短期内的信息，我们确实可以得到较好的反馈，不过一旦时间久远，那么便会出现长依赖问题，因为我们本次提供的信息和需要处理该信息的地方相隔甚远，网络难以去正确地分析处理，或者说，极易学习到错误的知识。LSTM这一新兴理念则是为了解决这一问题而被提出的，它是一种特殊的循环神经网络结构，在原本的重复链接单元上添加到了四个网络层，如图2

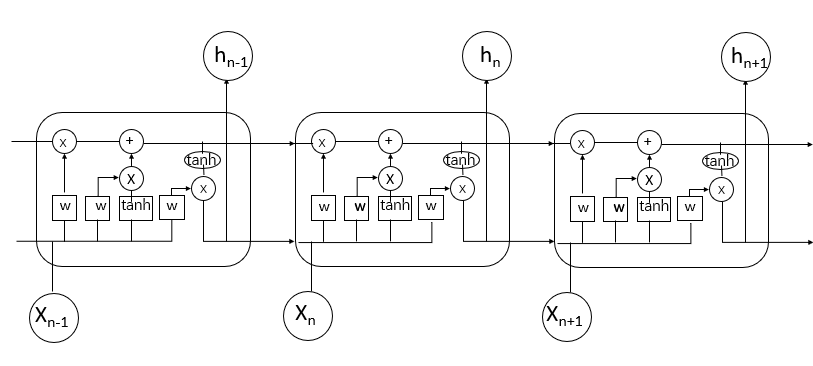


图2 LSTM网络结构图

LSTM网络有着一种处理信息的特殊结构，称之为门，它由一个点乘运算以及一

个限制数据输出大小范围为0到1的Sigmoid层组成。1为通过，而0则阻挡，也就是说该结构有着对输入数据是否可以流通的决定权。

2.2自适应矩估计Adam

Adam优化器是对随机梯度下降（Stochastic Gradient Descent,SGD)的扩展。它于2014年12月由Kingma与Lei Ba提出，通过考量梯度的一二阶距估计来迭代新的步长，从而达到更加高效地更新神经网络权重的目的。它主要有着以下几个特点1）目标固定。2)内存需求少。3）适合较大参数运算。4）适合稀疏梯度问题。5）超参数易调整。简而言之，通过自适应学习和动量法来加快网络函数的快速收敛是我们Adam优化器的重要特点。如下：

β1和β2都表示指数衰减率，β1可以控制当前梯度和动量的权重分配，β2负责操控过去梯度平方的作用情况。gt表示在t时刻函数的梯度。

mt = β1mt-1+(1-β1)gt

vt = β1vt-1+(1-β2)gt2

既有momentum储存过去梯度mt的指数衰减平均值的功能，同时也有RMSprop保留过去梯度数值的平方vt的指数衰减平均值的特点。因此，它也可以看作Momentum和RMSProp算法的结合。

2.3BP神经网络结构模型分析

反向传播神经网络是一种凭借其卓越的内部复杂模式下非线性映射能力和优异的提取数据特征自我学习能力而被广泛应用的多层前馈神经网络，它是McClelland与Rumelhart等人于1986年独立开发用来处理多层神经网络隐含层连接权学习障碍的结构模型。它是由输入层、输出层、隐含层所组成的结构。如图3

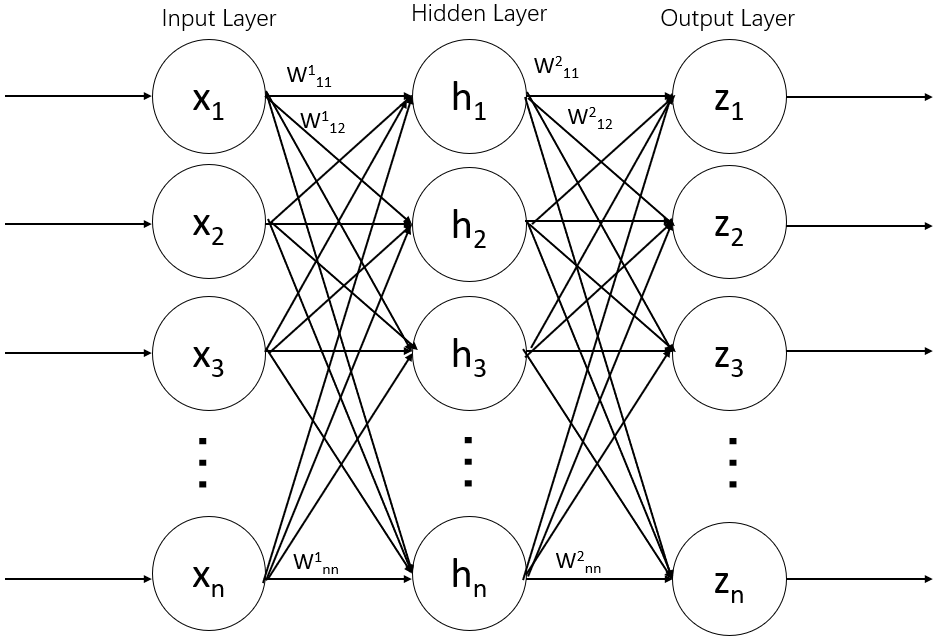


图3 BPNN网络结构图

如图所示，[x1,x2,x3,…,xn]是我们输入层的n维输入数据，在经过两层之间的权值点乘相加运算后，通过激活函数将数据压缩到隐含层并作为该层的输入向输出层传播，最终得到[z1,z2,z3,…,zn]。

3. LSTM-BPNN混合算法

本文采用LSTM-BPNN深度学习，具体流程框架如下图所示，左侧为LSTM神经网络部分，右侧为BPNN。首先初始化我们的各类超参数，接着从我们的数据仓库中提取出各地各时间的迁入迁出人数和患病人数。不断以某地5天的迁入人数[xn-4,xn-3,xn-2,xn-1,xn]为一组作为输入去预测临近的5天迁入人数[xn-3,xn-2,xn-1,xn,xn+1]，同时为了使我们的网络有更好的泛化使用能力，我们对所有地区的数据同时进行训练。同理，迁出人数和患病人数也如此训练。我们的主要目的是对未来某地疫情爆发情况进行分析，所以接下来我们将某地15天的迁入人数[xn-14,xn-13,… ,xn]和这15天的迁出人数[yn-14,yn-13,… ,yn]作为BPNN的输入集I1，如下

I1=

为了让我们的网络有更好的适应泛化能力，我们将a、b、c…z各地的I1输入集组合在一起形成一个大的输入矩阵I，如下：

I=

在这里我们采用MinMaxScaler函数对输入矩阵I进行数据归一化,最终得到的矩阵最终被压缩在[0,1]间,得到最终进入神经网络的输入矩阵Input。

Input =

输入层->隐藏层和隐藏层->隐藏层网络之间我们选取Relu为激活函数，

隐藏层->输出层之间我们选用Relu6作为激活函数。

输出层5得到输出矩阵Output，对其反归一化得到我们的真实数据矩阵，

Output=

可以得到未来第n+5可能爆发人数，当然，一次正向的并不能得出最优的输出，我们需要经过反向传播训练更新网络权值渐渐得出相对最优解。

我们用均方差平方根MSE函数作为损失函数对预测结果Zpredict和实际数据Ztrain求误差

MSE =

在SGD优化器，梯度下降的原则上，反向传递误差数据到两层隐含层和输入层，逐渐修正误差更新各层的网络连接权值。

同时我们也可以将BPNN所预测的爆发人数和LSTM所预测的人数作对比可见差异不大。

三、实验验证

3.1 实验准备

在实验开始之前，介绍我们所选用实验环境平台，以及相关训练数据集，然后用其他用来做预测模型的神经网络和我们现提出的LSTM-BPNN深度神经网络模型相对比，衡量评估不同模型之间的优缺点指标.

3.1.1 实验环境平台

实验所使用计算机的配置如下: 处理器为Inter(R) Core(TM) i5-8250U，CPU 频率为1.60GHz 和1.80GHz; 内存为8.00GB; 操作系统为 Windows 10( 64 位) ; 程序设计语言为 Python 3.8 ( 64 位); 集成开发环境为 PyCharm Community Edition 2019.1.1。神经网络搭建中，LSTM、BPNN模型由Python的torch-1.6.0+cpu.dist-info和torchvision-0.7.0+cpu.dist-info程序库实现。

3.1.2 数据集

实验数据采用Github开源数据仓库，数据集提供2020年中国各地1月10号到1月18号的迁入迁出人数和患新冠肺炎人数记录。该记录如表1、表2、表3所示。为讨论三者之间的联系以及预测未来可能患病人数，本文选择200个地点的迁入和迁出作为训练集，余下40个作为测试集。

表1 各地各时间迁入人数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 地点  时间 | 武汉 | 孝感 | 黄冈 | …… | 吐鲁番 |
| 1.10 | 4.6 | 1.41 | 1.87 | **……** | 0.26 |
| 1.11 | 4.64 | 1.69 | 2.23 | **……** | 0.27 |
| 1.12 | 4.37 | 1.58 | 2.34 | **……** | 0.25 |
| 1.13 | 4.83 | 1.43 | 2.14 | **……** | 0.26 |
| …… | **……** | **……** | **……** | **……** | **……** |
| 1.18 | 4.23 | 2.03 | 3.65 | **……** | 0.24 |

表2 各地各时间迁出人数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 地点  时间 | 武汉 | 孝感 | 黄冈 | …… | 吐鲁番 |
| 1.10 | 6.62 | 0.97 | 1.04 | **……** | 0.2 |
| 1.11 | 7.56 | 1.11 | 1.04 | **……** | 0.22 |
| 1.12 | 6.22 | 1.09 | 0.99 | **……** | 0.19 |
| 1.13 | 5.76 | 1.16 | 1.09 | **……** | 0.16 |
| …… | **……** | **……** | **……** | **……** | **……** |
| 1.18 | 7.71 | 1.00 | 0.96 | **……** | 0.21 |

表3 各地各时间患病人数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 地点  时间 | 武汉 | 孝感 | 黄冈 | …… | 吐鲁番 |
| 1.24 | 572 | 26 | 64 | **……** | 0 |
| 1.30 | 2639 | 541 | 573 | **……** | 1 |
| 1.31 | 3215 | 628 | 726 | **……** | 1 |
| …… | **……** | **……** | **……** | **……** | **……** |
| 2.6 | 11618 | 2141 | 1897 | **……** | 1 |

3.1.3 模型对比

1）LSTM模型

我们只用过去时间段内的患病人数作为时间序列的输入用来分析未来的患病发展趋势。通过如图4和图5可以得知，无论从不断趋近0的损失值的角度来讲，还是从预测值和实际值有着极高的拟合度来说，这个单纯依靠LSTM所搭建的网络都看似是成功。不过当我们采用更多数量的数据或者使用更加久远的数据时，该网络呈现的是欠拟合的状态，如图6。这也正好说明了我们的LSTM确实可以解决一定程度上的梯度问题，但是面对更长的时间序列的时候，依然保留着原有RNN难以处理大量级数据的根本问题.

表4 LSTM预测模型测试训练集

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 区域 | 1 | 2 | 3 | 4 | 5 |
| 实际值 | 571 | 26 | 64 | 10 | 1 |
| 预测值 | 575 | 26.1 | 64．1 | 10．1 | 1.05 |
| 均方误差 | 7.80e+00 | 7.38e-03 | 4.77e-03 | 3.34e-03 | 1.04e-03 |
| 区域 | **6** | **7** | **8** | **9** | **10** |
| 实际值 | 5 | 0 | 21 | 1 | 0 |
| 预测值 | 5.08 | 0.072 | 2.11 | 0.0106 | 0.0631 |
| 均方误差 | 3.55e-03 | 2.59e-03 | 1.21e-03 | 1.84e-03 | 1.99e-03 |

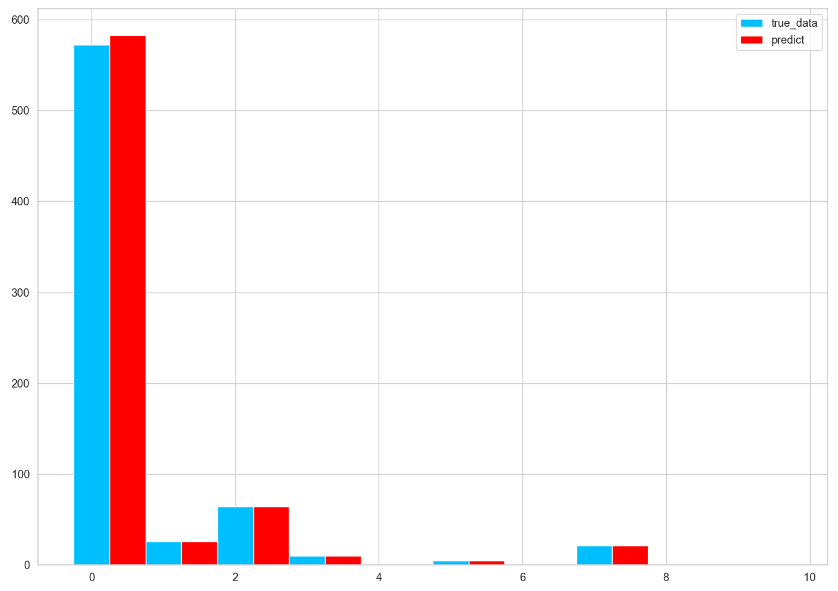


图4 使用训练集得出的预测值和实际值对比图

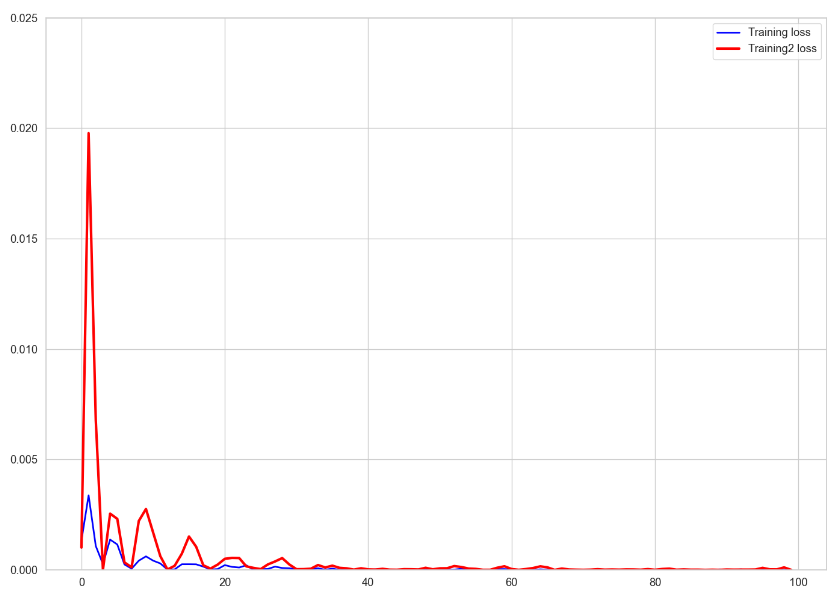


图5 使用训练集得出的损失函数对比图

表5 LSTM预测模型测试测试集

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 区域 | 1 | 2 | 3 | 4 | 5 |
| 实际值 | 11618 | 2141 | 1897 | 885 | 471 |
| 预测值 | 909.51 | 88.89 | 118.47 | 35.96 | 14.9 |
| 均方误差 | 5.73e+07 | 2.11e+06 | 1.58e+06 | 3.60e+05 | 1.04e+05 |
| 区域 | **6** | **7** | **8** | **9** | **10** |
| 实际值 | 915 | 838 | 553 | 610 | 635 |
| 预测值 | 32.00 | 24.86 | 36.78 | 19.06 | 18.84 |
| 均方误差 | 3.90e+05 | 3.31e+05 | 1.33e+05 | 1.75e+05 | 1.90e+05 |

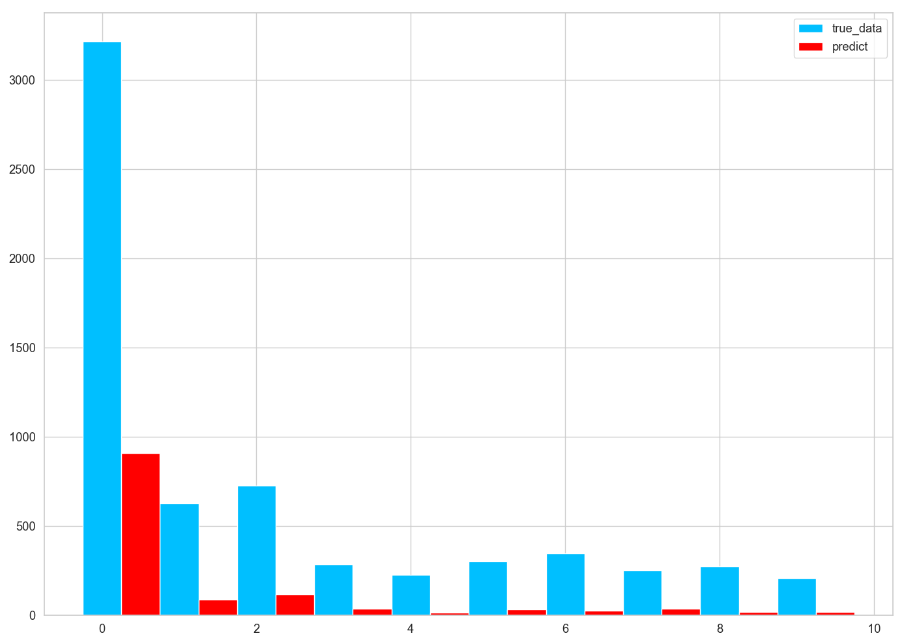


图6 使用测试集的LSTM模型结果图

2）BPNN模型

同理，我们使用过去时间段内患病人数作为网络的输入层来推测未来的患病人数。最后用10个地方的数据进行测试结果如表1以及图7所示，红色代表预测的数值，蓝色代表实际的数值，由表4可见，不仅和实际值相差巨大，且在我们预测第一个区域的数据的时候，因为该数据过大输出为Inf，在对过大数据进行压缩后，其数值依然居高不下，如图7的情况，

表6 BPNN预测模型测试测试集

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 区域 | 1 | 2 | 3 | 4 | 5 |
| 实际值 | 11618 | 2141 | 1897 | 885 | 471 |
| 预测值 | 30000 | 5784 | 6828 | 2280 | 1759 |
| 均方误差 | 1.690e+08 | 6.66e+06 | 1.22e+07 | 9.73e+05 | 8.29e+05 |
| 区域 | **6** | **7** | **8** | **9** | **10** |
| 实际值 | 915 | 838 | 553 | 610 | 635 |
| 预测值 | 2334 | 2808 | 2304 | 1821 | 1558 |
| 均方误差 | 1.01e+06 | 1.94e+06 | 1.53e+06 | 7.33e+05 | 4.26e+05 |

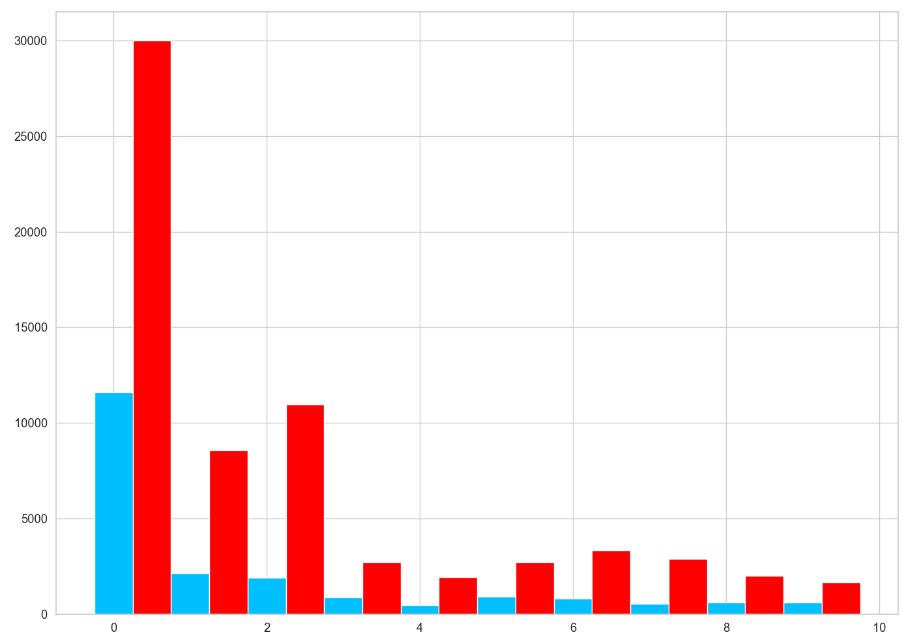


图7使用测试集的BPNN预测模型

3）LSTM-BPNN混合模型

从上述两个模型可知，LSTM对非特大量级数据有着极好拟合度，而BPNN则可以有效提取复杂内部机制的数据，并利用其特征数值进行自适应学习和泛化。为了和如上两个模型做对比我们依旧采用同样的10个测试集作为测试，由表7和图9可知，相对于前者有着更好的学习能力与预测能力。

表7 LSTM-BPNN预测模型测试测试集

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 区域 | 1 | 2 | 3 | 4 | 5 |
| 实际值 | 11618 | 2141 | 1897 | 885 | 471 |
| 预测值 | 16500 | 425 | 2050 | 964 | 114.06 |
| 均方误差 | 1.19e+07 | 1.47e+06 | 1.17e+04 | 3.12e+03 | 6.37e+04 |
| 区域 | **6** | **7** | **8** | **9** | **10** |
| 实际值 | 915 | 838 | 553 | 610 | 635 |
| 预测值 | 78.94 | 567 | 231.5 | 254.8 | 231.8 |
| 均方误差 | 3.49e+05 | 3.67e+04 | 5.17e+04 | 6.31e+04 | 8.13e+04 |

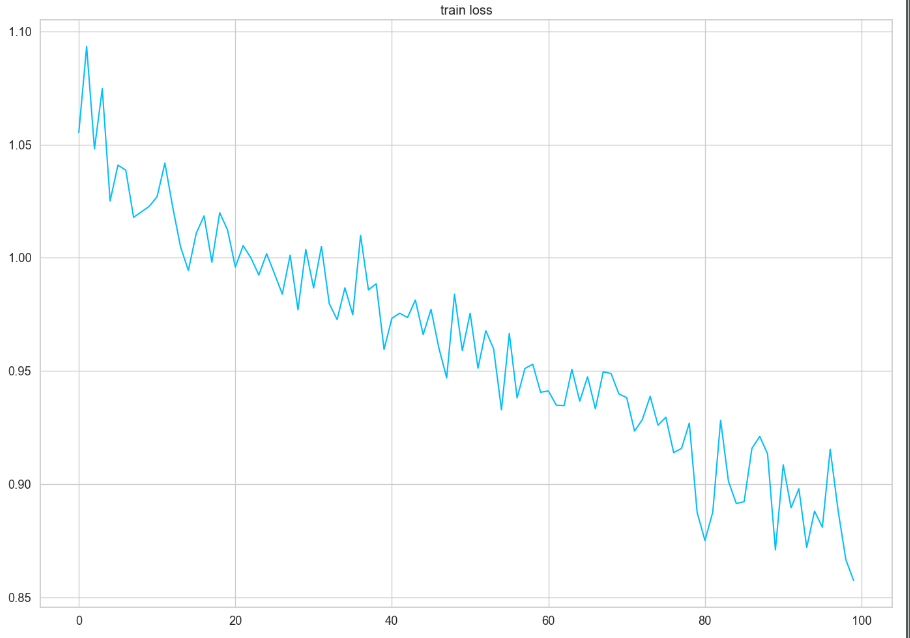


图8使用测试集的LSTM-BPNN损失函数图

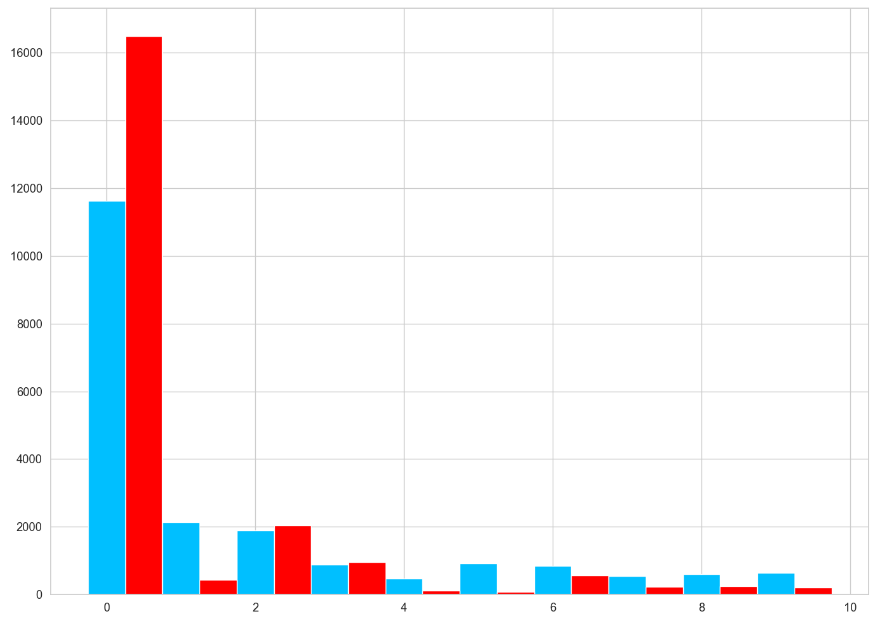


图9使用测试集的LSTM-BPNN预测模型

（4）实验结果分析对比

为了表现出不同的神经网络算法的预测效果，我们抽样了10个区域的数据，对其均方误差统计取平均值。实验结果表明单纯BP神经网络的均方误差最大，基于长短期记忆神经网络的模型和LSTM-BPNN模型预测的误差最小，其中我们的LSTM-BPNN混合模型误差最小，且不存在小量级数据规模的依赖，故而在可以极佳地避免过拟合问题的同时，也保留着有极强的泛化学习能力，如表8所示。为了更加清楚明了地表现LSTM-BPNN的优越性，我们采用将四条曲线放在同一张折现图的数据可视化处理方式，如图10所示。横坐标表示10个区域，纵坐标表示患病人数，绿色实线表示真实数值，橙色虚线表示BP神经网络预测的的各地数值，红色虚线表示LSTM预测的各地数值，浅蓝色表示BPNN-LSTM混合模型预测的各地数值。

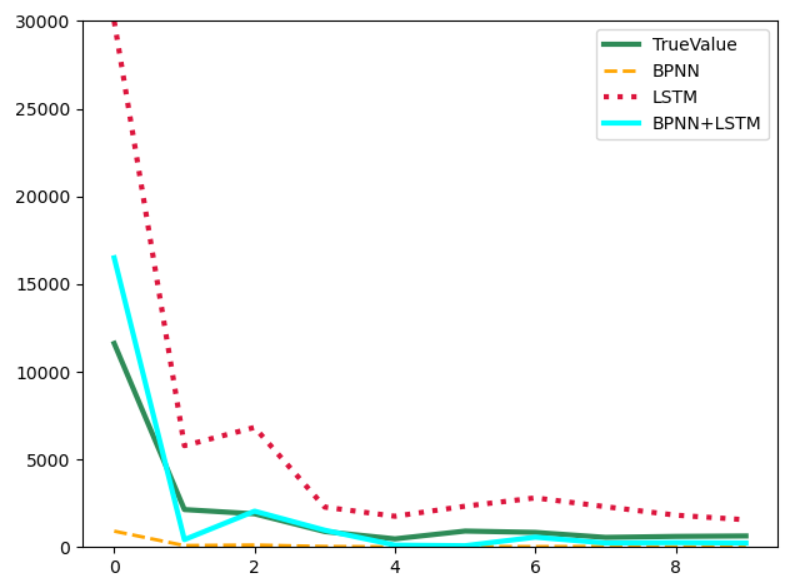


图10 基于不同神经网络模型的预测折线图

表8 基于不同神经网络模型的均方差比较（部分）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | LSTM | | BPNN | | LSTM-BPNN | |
| **区域** | **实际值** | **预测值** | | **均方差** | **预测值** | **均方差** | **预测值** | **均方差** |
| **1** | 11618 | 909.51 | | 5.73e+07 | 30000 | 1.69e+08 | 16500 | 1.19e+07 |
| **2** | 2141 | 88.89 | | 2.11e+06 | 5784 | 6.66e+06 | 425 | 1.47e+06 |
| **3** | 1897 | 118.47 | | 1.58e+06 | 6828 | 1.22e+07 | 2050 | 1.17e+04 |
| **4** | 885 | 35.96 | | 3.60e+05 | 2280 | 9.73e+05 | 964 | 3.12e+03 |
| **5** | 471 | 14.9 | | 1.04e+05 | 1759 | 8.29e+05 | 114.06 | 6.37e+04 |
| **6** | 915 | 32.00 | | 3.90e+05 | 2334 | 1.01e+06 | 78.94 | 3.49e+05 |
| **7** | 838 | 24.86 | | 3.31e+05 | 2808 | 1.94e+06 | 567 | 3.67e+04 |
| **8** | 553 | 36.78 | | 1.33e+05 | 2304 | 1.53e+06 | 231.5 | 5.17e+04 |
| **9** | 610 | 19.06 | | 1.75e+05 | 1821 | 7.33e+05 | 254.8 | 6.31e+04 |
| **10** | 635 | 18.84 | | 1.90e+05 | 1558 | 4.26e+05 | 231.8 | 8.13e+04 |
| **平均均方误差** | |  | | 6.27e+06 |  | 1.95e+07 |  | 1.40e+06 |

4．结论

本文提出了基于LSTM+BPNN深度神经网络的新冠疫情预测模型，包括对网络模型的参数选择、网络结构、最终预测和模型对比等相关内容。通过实验证明：

1）与传统的RNN模型和LSTM模型相比，LSTM+BPNN深度神经网络结构模型汲取了LSTM优秀的拟合时间序列能力，并且因为在BPNN中合理选择权值、阈值和更加适当的网络层次结构解决了庞大数据难以精准处理的障碍。

2）与其他类型的神经网络相比，由于BP神经网络需要对大量特征数值分析纠正再学习，LSTM需要对较长时间序列进行拟合，故而该混合模型在训练过程中也需要更多的时间去调整精度。

3）LSTM+BPNN深度神经网络模型的均方误差更小更集中，表明与BPNN、LSTM模型相比，LSTM+BPNN深度神经网络预测得出的结果更加准确稳定，更能合理地预测新冠疫情。

LSTM+BPNN深度神经网络完善了单独LSTM和BPNN的部分缺陷，经过LSTM时间序列的预测得出新的迁入和迁出数据，再用BPNN对不同时间、不同地点的两者数据提取精炼得出最终结果。目前尚处于初步研究阶段，该网络结构的优化和其他预测模型相比能否还拥有卓越的性能等问题将成为后续开展工作的主要目标，由此使得该网络能更进一步。