基于CNN-LSTM网络的在线多任务销售预测方法

作者1姓名1, 2a，作者2姓名1, 2b†，作者3姓名1, 2a, 2b

(1. 单位1名称 单位1部门名称, 广西 桂林 514000; 2. 单位2名称 a. 单位2部门a名称; b. 单位2部门b名称, 上海 200093)

摘　要：为解决传统时间序列预测模型在多任务销售预测任务中模型泛化能力一般且只能进行离线预测的问题，提出了结合长期和短期存储网络(LSTM)和卷积神经网络(CNN)的混合深度学习模型，并设计了基于滑动窗口的数据输入机制以实现在线预测。实验利用宁波M公司的销售数据进行了实证分析,实验结果表明：CNN-LSTM模型以最小的信息损失来进行建模，对比单一模型、传统的经济预测模型、回归模型以及机器学习模型，CNN-LSTM 模型不仅在测试集上取得了最小平均绝对百分比误差，同时模型预测性能在长时间预测中优于LSTM。

关键词：关键词1；关键词2；关键词3﹝一般可选3~8个关键词，通识的英文关键词可不翻译﹞

中图分类号：TPxxx﹝查询请参考http://ztflh.xhma.com﹞

Online multi-task sales forecasting method based on CNN-LSTM network

Familyname Givenname1, 2a, Zhang Jianguo1, 2b† , Deng Zhenrong1, 2

(1. 单位1部门英文名称, 单位1英文名称, Guilin Guangxi 541004, China; 2. a. 单位2部门a英文名称, b. 单位2部门b英文名称, 单位2英文名称, Shanghai 200093, China)

Abstract: In order to solve the problem that the traditional time series prediction model has general model generalization ability and can only make offline prediction in multi-task sales prediction tasks, a hybrid deep learning model combining long-term and short-term storage networks (LSTM) and convolutional neural networks (CNN) is proposed. A data input mechanism based on sliding window is designed to realize online prediction. The sales data of Ningbo M Company was used experimentally and empirical analysis. The experimental results show that the CNN-LSTM model is modeled with minimal information loss. Compared with the single model, traditional economic prediction model, regression model and machine learning model, the CNN-LSTM model not only achieves the minimum average absolute percentage error on the test set, but also the model prediction performance is better than LSTM in long-term prediction.

0 引言

销售是企业对供应链和运营进行优化时必要的一步(Rao, 1985)。KATI S（2018年2月）[1]研究表明，需求因素对未来产生影响的前提下进行[2].准确的销售预测可以推动产业链的发展，是企业运营的重要组成部分。相反，当企业不能科学的做出销售预测和发展趋势的判断时[3]，往往会导致公司的供应链和运营部门进入混乱或者停滞状态(mindykim, 2019)。

事实上，时间序列预测不仅在工业尤为重要，而且在学术界亦是如此。用Adhikari和Agrawal (2013)的话来说，”研究人员做了很多努力来开发有效的模型,并想以此来提高时间序列的预测精度“。然而，不同

的销售渠道和销售场景有者不同的销售时间序列预测方法，这也让很多研究人员没有办法建立出合适的模型来应用于特定场景中。尤其是在多任务、长时间预测时，因使用不适用于给定时间序列的方法进行预测可能会导致企业季度或年度战略的失败。因此，不论是管理者还是学术研究人员，必须谨慎选择适合当前情况的预测方法(Purthan et al)。

在过去的几十年里，研究人员和企业经营者通常会采用例如HW、神经网络、ARIMA等方法进行相关的销售时间序列预测，在少量数据和特定的预测任务背景下，类似的预测模型能够让预测误差在可接受的范围内。Cranage和Andrew(1992)使用79个月的餐馆销售数据，将计量经济学方法与时间序列方法(SARIMA和HW)进行比较。他们的结果表明，SARIMA方法在初始期和七个月的预测期都有更好的表现。然而，这项研究只针对一家餐厅，并没有对多任务时间序列预测任务进行总结。Makatjane和Moroke(2016)使用南非19年的月度汽车销量数据，发现HW方法相对于SARIMA方法具有更强的预测能力。Udom(2014)使用移动平均法、HW法和SARIMA法对塑料行业中单个分销商的五种不同产品的销售额进行了预测，并以MAPE为评价指标，发现SARIMA方法对销售额的预测效果最好。Frank等人(2003)采用单季节指数平滑法(SSES)，HW和ANNs方法用于女装销售预测。他们发现神经网络显示出令人满意的拟合优度统计量(R2)，但HW方法在比较实际销售和预测销售方面表现得更好。但神经网络对序列数据的前一个时间步长没有记忆，其不能在较长的时间范围内捕获时间序列的特征，例RNN有时会出现梯度消失的问题。

虽然以上文献均采用销售数据进行预测，但得出的结果并不一致。在对不同的时间序列进行建模时，各种方法能够取得性能通常会和数据中的时间维度特征相关，而当机器学习在预测任务上通常会展现比传统模型更高的预测性能时，更多的人选择用此类方法来进行时间序列数据，例如：LSTM (Hochreiter和Schmidhuber, 1997)、XGBoost以及Prophet等。为了验证LSTM网络是否能够取得比传统模型更好的预测效果，Weytjens等人(2019)还将SARIMA和Prophet与ANN和LSTM方法进行比较，以预测现金流。在他们的工作中，他们引入了一种新的绩效衡量方法，即利益机会成本(IOC)。研究利用IOC和MSE作为成本函数进行模型的评价，最终得出结论:LSTM是预测现金流的最佳方法。而在多任务预测中，Yu等(2018)使用LSTM方法预测销售。他们分析了66种产品，包括45周的数据。研究结果表明LSTM为66种产品中的17种提供了准确的预测，而并非全部。考虑到这些数据几乎没有季节性这一特点，单用一个LSTM网络进行多任务的销售预测会面临时间序列长度不够的情形，追其本质，是单个LSTM网络并不能稳定地识别序列数据中的时序特征。

以上的研究工作虽然为销售预测提供了离线预测场景下的可靠方法，但随着云计算、分布式数据库存储技术的发展，企业会要求模型能够进行实时、在线地预测。并且，随着企业推广能力的增强，企业的营业点会分布在各个范围内。这就要求模型要能够进行在线的多任务预测，这使得模型要能够高效地识别出时序数据中的深层次特征，即数据的时空相关性。为适应这一场景，研究设计了一种基于CNN-LSTM网络的在线多任务销售预测模型。此外，本文还用此模型对比了传统的经济预测模型、统计机器学习和单个LSTM模型。目的为在线多任务预测提供理论支持。

1 数据处理

1.1数据预处理

机器学习模型所得出的良好的预测结果不仅仅依赖于模型本身的质量，还和数据有着很大的关系，通过数据的处理可以让算法模型预测出具有参考性的结果。采用的方法有数据的规划范、时间序列处理、缺失值填充法等。

为了得到准确的预测结果和清晰的展示，按照时间序列将商品销售数据规范化，分类整理商店和商品类别。采用特殊值填充法对测试集进行缺失值处理，将所有的销售额用“0”值填充。为了同步训练集和测试集的顺序，让训练集和测试集中的列名和顺序保持一致性。

1.2特征提取

日期按照one-hot方法进行日、周、月编码处理，以便确保我们的网络模型可以更好的处理时间序列数据，对季节性的的数据更加敏感。预测方法选择在线单步预测，因此数据的输入采用追加方式将数据在训练的过程中不断地输入到模型中来进行模型调节。由于神经网络的数据输入会造成一定的抖动，所以采用滑动窗口算法来消除这种抖动，窗口大小=∆N，步长=n，时间间隔=∆T。最后，使用min-max标准化(Min-max normalization)方法对数据集中的销售额进项归一化处理。



2基于滑动窗口的在线预测算法

在线预测与传统的机器学习方法不同，传统的机器学习方法是离线训练，通过批量的样本输入来训练模型，模型更新周期较长，实时效果反馈效果差，在小规模数据集时会有一定效果，但是，当大量数据输入模型时会无法实时响应和反馈模型。在线预测算法采用小批量或单量的输入数据方法，模型参数实时更新，预测结果在线实时反馈。

本文提到的在线单步预测的神经网络模型在回归预测时输入数据会造成一定的抖动，为了解决这种不稳定性并提供在线实时预测，我们采取滑动窗口算法来输入数据。

2.1算法流程



算法1. 基于滑动窗口的在线预测算法

输入：变量xi∈D预处理的数据集

输出：实时预测结果

1. 将预处理好的销售数据进行特征的提取，并初始化模型参数的权重ω
2. 根据滑动窗口算法选取训练样本实例，使训练样本持续到来
3. 使用实时训练的模型预测出当前结果
4. 将预测值和实测值进行比较，依据损失函数得出损失值(loss)
5. 利用损失函数的梯度更新当前的决策，反向传播更新模型的权重，直到损失值很小或者不再下降
6. 输出实时结果



3神经网络模型

3.1 CNN模型

卷积神经网络（Convolutional Neural Networks, CNN），是一种前馈多层的神经网络，由大量的可学习权重和偏置神经元组成。神经元在CNN的网络体系中以神经网络层的形式展现，它由一个输入层，许多隐藏层和一个输出层组成。本文构建的CNN模型由卷积层和池化层构成。

卷积（CONV）层是CNN模型的基本构建块。在这一层中，输入卷积被卷积使用一组可学习的过滤器，从而生成与每个过滤器相对应的几个特征图。每个神经元链接到输入体积的特定区域，并且所有神经元具有相同的连接权重。

池化（Pooling）层通常是压缩数据的神经网络层，对特征图的空间大小进行下采样，从而减少了可学习的参数数量和计算成本，因此有助于控制过拟合问题。CNN中最广泛使用的两种池化策略是最大池化（max pooling）和平均池化（average pooling）。 最大池化是许多研究者在CNN网络中使用最多的策略，本文CNN模型的池化层选择最大池化策略。

3.2 LSTM模型

长短时记忆网络（Long Short-Term Memory，LSTM），由循环神经网络（Recurrent neural network，RNN）演变而来。1997年，S Hochreiter和J Schmidhuber在RNN的基础上提出了LSTM模型[4]，LSTM使用存储单元记忆历史信息，每个存储单元包含一个存储细胞和三个门控制器（输入门，输出门和忘记门）。存储细胞中的输入门控制输入信息，输出门控制输出信息。遗忘门控制需要丢弃哪些信息并重置存储细胞。

3.3 CNN-LSTM模型

通过将CNN和LSTM的组合，构建出CNN-LSTM混合模型。充分利用CNN对局部特征提取的优势，将CNN处理后的数据送入到LSTM网络中进行时间序列的预测。混合模型如图所示



搭建混合模型时，首先将预处理后的数据根据滑动窗口算法来持续输入到CNN模型中，在CNN模型中的卷积层设置1X1维度的卷积核，再将经过3层卷积层局部特征提取后的数据通过最大池化层（Max Pooling）压缩数据和参数。最后将CNN模型中提取出的特征送入LSTM模型中进行实践序列的预测。

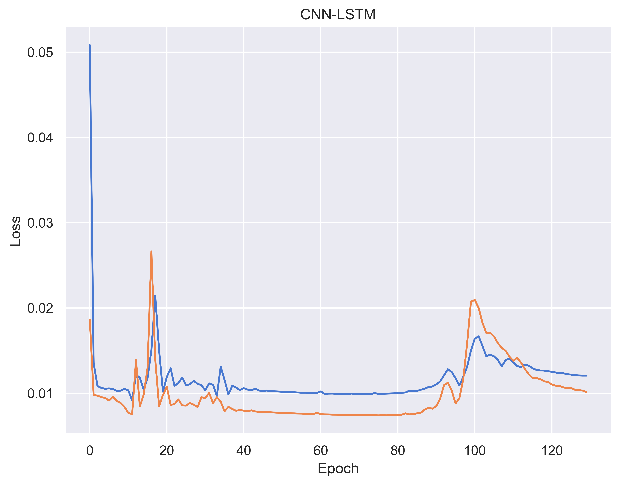
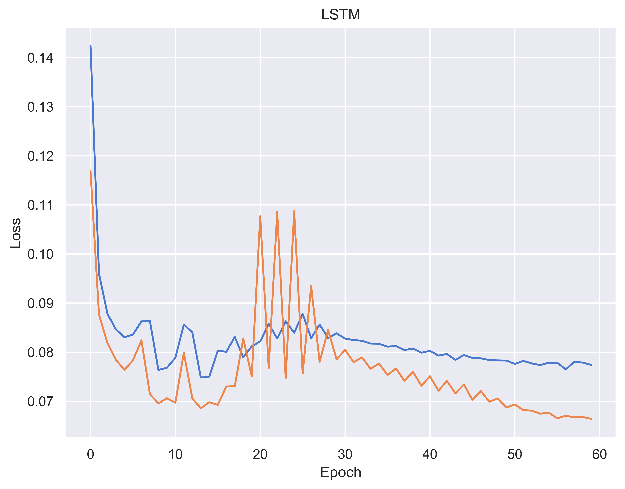
LSTM对时间序列下的销售预测有着较好的效果，但是随着数据量的增大，LSTM模型难以处理过于复杂的特征，导致难以给出更为准确的预测。基于此，提出CNN-LSTM混合深度模型，传统的CNN模型可以有效的提取数据的局部特征[5]，但是对时间序列的敏感性较低，所以我们在CNN的基础上加入LSTM模型。

4 实验结果

本节使用LSTM模型和CNN-LSTM模型对2020年宁波M公司的第一季度销售数据进行预测，通过不同周期下的时间序列对比展示、离线和在线算法的对比，并且利用SMAPE方法对预测结果进行了误差评定。

4.1Loss对比

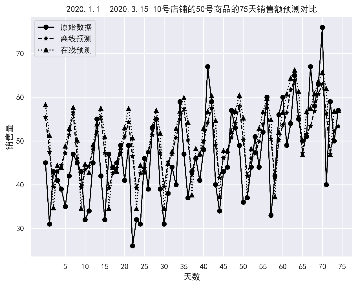
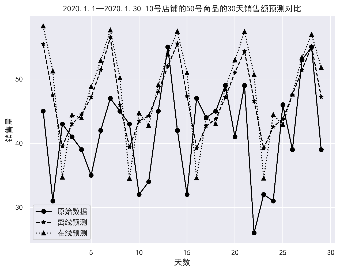
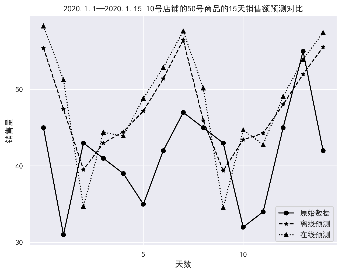
按照上文所提到的算法思路建立了LSTM和CNN-LSTM模型，将多任务算法下预处理的时间序列数据分别输入到2个模型中进行训练，在训练过程中得到单一模型和混合模型的loss计算回归图，结果如图所示。

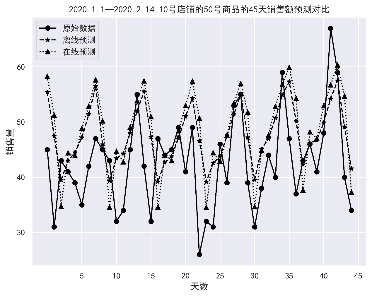
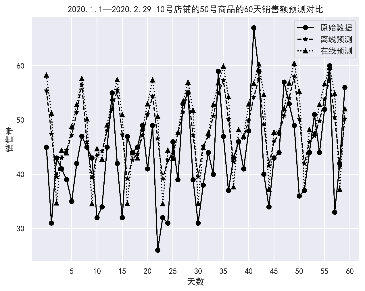
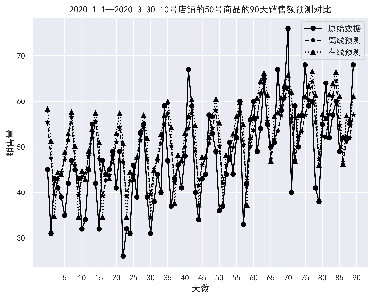


由图2整体结果可见，CNN-LSTM混合模型相比于单一模型LSTM的loss值更低，且收敛的速度更快。单一模型在epoch 0到30的时候，loss值有很明显的震荡趋势，难以达到收敛，这是因为LSTM在数据较大的情况下难以有效的提取特征，而CNN-LSTM混合在数据送入LSTM模型中之前，CNN就已经提取好了关键特征，从而让模型在训练的时候可以快速收敛，CNN的加入大大降低了单一模型时过拟合的风险。

4.2在线与离线算法对比

针对文中第2节提到的在线预测算法，我们利用训练好的模型对2020年第一季度90天的销售额数据进行了实时周期性的预测，并且我们对比了原始数据、离线预测和在线预测的实时预测结果。我们选择每15天为一个周期进行实时预测，如图3所示我们分别预测并展示了15天、30天、45天、60天、75天和90天等6个周期的销售数据。

****

****

由上图两种的预测方法所得出的结果可知，相比于离线模型的预测结果，在线模型对销售额数据的预测与原始实际数据更加贴近，且随着周期的增加销售走势也更加一致。

**4.3模型性能分析**

为了证明本文提出的CNN-LSTM混合深度模型的优越性，采用SMAPE（对称的平均绝对百分比误差）对不同算法，如LSTM、Xgboost、灰色模型和ARIMA进行对比分析。SMAPE表达公式如下所示。

（1）

对比结果如表所示：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | CNN-LSTM | LSTM | Xgboost | 灰色模型 | ARIMA |
| SMAPE | 9.686975 | 13.824194 | 15.801907 | 15.917542 | 16.893254 |

实验结果证明，CNN-LSTM混合深度模型，其预测的结果的误差值9.68%相比其他算法误差值较小，通过对其他常用算法的误差值进行比较，可以看出混合深度模型的结果更加精准，并且对时间序列相关的销售预测有着较好的效果。

**4.4模型差异对比**

正态性检验是判断数据总体是否服从正态分布的一种检验方法，它对做统计工作之前有着重要意义，采用偏度-峰度检验法检验数据的正态分布性。

图片包含 游戏机

描述已自动生成手机屏幕的截图

描述已自动生成

有表中数据可知，CNN处理后的数据峰度和偏度更加接近于0，可以证明CNN处理后的数据相比于元数据更好的服从正态分布。

|  |  |  |
| --- | --- | --- |
|  | 峰度 | 偏度 |
| 原始数据 | 5.340550192030676 | -1.0892404692253963 |
| CNN处理后的数据 | 0.37508950888516956 | -0.6558730006217957 |

时间序列的平稳性对于有关时间序列的预测有着重要意义，非平稳的时间序列数据来预测未来的销售数据十分困难，所以将非平稳的时间序列转换为平稳的时间对预测研究有着重大意义。

对原始数据和CNN处理后的数据进行比较发现，原始数据的自相关系数在很长的延迟时期里一直保持为整数，且数值较大；CNN处理后的数据自相关系数数值较小，并且序列保持在零轴附近波动，证明了CNN处理后的数据具有较强的平稳性和相关图特征。

社交网站的手机截图

描述已自动生成手机屏幕截图

描述已自动生成

原始数据和CNN处理后的数据进行单方跟检验对比发现，原始数据检验统计量大于1%和5%的置信区间临界值，而CNN处理后的数据原始数据检验统计量远大于1%置信区间的临界值，且p值约等于0，因此可以非常好地拒绝该假设，可以很好的证明CNN处理后的数据具有一定的平稳性，对未来的预测有帮助。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据 | 检验统计量 | P值 | 1%置信区间 | 5%置信区间 | 10%置信区间 |
| 原始数据 | -2.60611 | 0.09174 | -3.44316 | -2.86719 | -2.56977 |
| CNN处理后的数据 | -6.56076 | 8.39281 e-09 | -3.44384 | -2.86749 | -2.56994 |

5 总结

［1］KATI S,TEEMU L,PETRI S，et al．Forecasting sales in industrial services [J］．Journal of Service Management，2018，29（2）：277－300．

[2]周小溪,徐行,孟剑飞,苏旭中.服装销售预测方法研究进展[J].针织工业,2020(03):68-72.

[3]江雪.基于时间序列模型的商业企业销售量预测方法研究[J].创新科技,2015(07):41-44.

[4] S. Hochreiter, J. Schmidhuber. Long Short-term memory. Neural Computation, 9(8), 1735-1780,  
1997.

[5]孙波,杨磊,郭秀梅,陈冉,张童,贾昊.基于CNN和SVM混合模型的心电信号识别方法[J/OL].山东农业大学学报(自然科学版):1-6[2020-05-14].http://kns.cnki.net/kcms/detail/37.1132.S.20200509.2342.030.html.