

# 工业装备状态预测 暨模型训练加速方法研究

(申请清华大学工程硕士学位论文)

培 养 单 位 ： 软件工程系

学           科 ： 软件工程

研   究   生 ： 李 思 宇

指 导 教 师 ： 邓 仰 东   副教授

二〇二一年四月



# **Prediction of Industrial Equipment State and Research on Model Training Method**

Thesis Submitted to

**Tsinghua University**

in partial fulfillment of the requirement

for the degree of

**Master of Engineering**

in

**Software Engineering**

by

**Siyu Li**

Thesis Supervisor: Associate Professor Deng YangDong

**April, 2021**



学位论文指导小组、公开评阅人和答辩委员会名单

指导小组名单

李 XX	教授	清华大学
王 XX	副教授	清华大学
张 XX	助理教授	清华大学

公开评阅人名单

刘 XX	教授	清华大学
陈 XX	副教授	XXXX 大学
杨 XX	研究员	中国 XXXX 科学院 XXXXXXXX 研究所

答辩委员会名单

主席	赵 XX	教授	清华大学
委员	刘 XX	教授	清华大学
	杨 XX	研究员	中国 XXXX 科学院 XXXXXXXX 研究所
	黄 XX	教授	XXXX 大学
	周 XX	副教授	XXXX 大学
秘书	吴 XX	助理研究员	清华大学



# 关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：

清华大学拥有在著作权法规定范围内学位论文的使用权，其中包括：（1）已获学位的研究生必须按学校规定提交学位论文，学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文；（2）为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆、资料室等场所供校内师生阅读，或在校园网上供校内师生浏览部分内容；（3）按照上级教育主管部门督导、抽查等要求，报送相应的学位论文。

本人保证遵守上述规定。

（保密的论文在解密后遵守此规定）

作者签名：\_\_\_\_\_

导师签名：\_\_\_\_\_

日 期：\_\_\_\_\_

日 期：\_\_\_\_\_





## 摘 要

利用蓬勃发展的人工智能和大数据技术,针对工业装备特别是复杂装备实现预测式维护是工业互联网+的重要组成部分,能够大幅度降低装备的运行成本、全面保障装备安全稳定运行,对于我国制造业转型升级、跨越式发展具有极其重要的意义。由于工业装备数据以传感器采集的时间序列数据为主,因此以上问题很大程度上体现为针对时间序列的预测问题。

工业装备数据来源复杂、机理多样、场景众多,而且数据分布上一般具有正常运行样本量巨大单故障样本稀少的特点。如何根据正常运行样本进行训练并同时具备故障检测能力,已经成为亟待解决的问题。

同时,由于其数据特点,工业人工智能摸往往存在着模型数量大、训练时间长、模型泛化能力差的问题,如何加快模型训练速度、提供模型构造效率,也成为迫切需要解决的问题。

针对以上问题,本文提出了两个研究目标:

- 1) 利用正常数据训练出满足实际应用需求、并能发现故障的时间序列预测模型;
- 2) 提高时间序列模型训练效率。

本文的主要贡献包括如下内容:

1. 针对牵引系统设备温度监测,设计了基于历史信息和基于电气信号两种不同的工业装备预测模型;
2. 构造了基于以上模型的完整预测方案,首先应用 LSTM 和 CNN 搭建时间序列预测网络,利用经验模态分解进行信号分解,对不同频率信号进行分别预测;并且进一步对高低频数据采用不同的采样频率;尝试应用自编码器进行相关特征处理。
3. 研究利用重要性采样的方法对模型训练进行加速,设计和搭建了通用的加速框架,并且进行了实验验证,能够缩短训练过程 50%。

**关键词:** 时间序列预测; 深度神经网络; 重要性采样; 经验模态分解

## Abstract

An abstract of a dissertation is a summary and extraction of research work and contributions. Included in an abstract should be description of research topic and research objective, brief introduction to methodology and research process, and summarization of conclusion and contributions of the research. An abstract should be characterized by independence and clarity and carry identical information with the dissertation. It should be such that the general idea and major contributions of the dissertation are conveyed without reading the dissertation.

An abstract should be concise and to the point. It is a misunderstanding to make an abstract an outline of the dissertation and words “the first chapter”, “the second chapter” and the like should be avoided in the abstract.

Keywords are terms used in a dissertation for indexing, reflecting core information of the dissertation. An abstract may contain a maximum of 5 keywords, with semi-colons used in between to separate one another.

**Keywords:** keyword 1; keyword 2; keyword 3; keyword 4; keyword 5

# 目 录

摘 要.....	I
Abstract.....	II
目 录.....	III
插图和附表清单.....	VI
符号和缩略语说明.....	VIII
第 1 章 引言 .....	1
1.1 研究背景与意义.....	1
1.2 问题描述与主要挑战 .....	2
1.2.1 问题描述及研究目标.....	2
1.2.2 时间序列预测的挑战性.....	3
1.3 研究内容与主要贡献 .....	3
1.4 组织结构 .....	4
第 2 章 国内外研究现状 .....	5
2.1 引言 .....	5
2.2 时间序列预测方法 .....	5
2.2.1 时间序列预测常用模型.....	5
2.2.2 提高预测效果方法.....	8
2.3 对时间序列模型的训练加速 .....	8
2.3.1 模型训练加速方法研究现状.....	8
2.3.2 筛选有效样本（重要性采样） .....	8
2.3.3 重要性采样介绍.....	10
第 3 章 时间序列数据分析与处理 .....	12
3.1 引言 .....	12
3.2 数据及评价指标说明 .....	12
3.3 数据特点分析 .....	13
3.3.1 工业装备状态数据特点.....	13
3.3.2 时间序列数据平稳性分析.....	14

3.4 数据预处理 .....	14
3.4.1 数据导入及清洗 .....	15
3.4.2 数据重采样 .....	15
3.4.3 数据分段 .....	16
3.4.4 整理数据表并通过滑动窗口取模型输入 .....	17
3.4.5 数据标准化 .....	17
3.4.6 不同序列划分方式 .....	18
3.5 实验环境 .....	19
3.5.1 实验说明 .....	20
3.5.2 选用 Tensorflow2 编写的优势 .....	20
3.6 本章小结 .....	21
<b>第 4 章 工业装备状态预测 .....</b>	<b>22</b>
4.1 引言 .....	22
4.2 工业装备状态预测方案 .....	23
4.2.1 系统设计 .....	23
4.2.2 方法说明 .....	24
4.2.3 两种不同的状态预测方法 .....	26
4.3 实验结果 .....	26
4.3.1 往前预测多步结果 .....	26
4.3.2 防止过拟合的策略实验结果 .....	27
4.3.3 两种不同方法的工业装备状态预测的结果 .....	27
4.4 本章小结 .....	28
<b>第 5 章 利用重要性采样对训练过程加速 .....</b>	<b>30</b>
5.1 引言 .....	30
5.2 训练加速算法 .....	30
5.3 基于重要性采样的训练加速方法框架的设计 .....	31
5.3.1 模型结构设计 .....	31
5.4 训练加速实验效果 .....	32
5.5 本章小结 .....	34
<b>第 6 章 总结与展望 .....</b>	<b>35</b>
6.1 本文总结 .....	35
6.2 未来展望 .....	35

## 目 录

---

参考文献.....	36
附录 A 补充内容.....	38
致 谢.....	39
声 明.....	40
个人简历、在学期间完成的相关学术成果.....	41
指导小组学术评语.....	42
答辩委员会决议书.....	43

## 插图和附表清单

图 2.1	应用于时间序列预测的典型模型类型及其特点总结 .....	6
图 2.2	ForecastsNet 网络结构 .....	7
图 2.3	TCN 空洞卷积结构 .....	7
图 3.1	5103 号列车数据中环境温度变化图（横轴是时间，纵轴是环境温度，单位为摄氏度） .....	15
图 3.2	电网数据分段及划分处理流程图 .....	16
图 3.3	列车走行部数据采集时间间隔分布直方图（横坐标：间隔 (s)，纵坐标：频次） .....	16
图 3.4	数据标准化前预测效果示意图 .....	17
图 3.5	数据标准化流程 .....	18
图 3.6	数据标准化效果说明图 .....	18
图 3.7	出现过拟合问题的网络训练过程 loss 变化 .....	19
图 4.1	预测式故障检测示意图 .....	22
图 4.2	列车轴温预测系统流程框架图 .....	23
图 4.3	工业装备状态预测部分系统设计 .....	23
图 4.4	模型在高频处会预测效果差示意图 .....	24
图 4.5	往前预测多步均方误差（横轴坐标值代表往前预测的步数，纵轴是均方误差） .....	27
图 4.6	出现过拟合问题的网络训练过程 loss 变化 .....	27
图 4.7	过拟合策略效果图：加了 dropout 之后的模型 loss 曲线变化 .....	28
图 5.1	重要性采样模型结构 .....	31
图 5.2	三次对比实验训练加速平均效果：验证集的 loss 变化趋势图 .....	32
图 5.3	三次对比实验训练加速平均效果：验证集的 loss 变化趋势图（100 个 epoch 开始） .....	33
图 5.4	三次对比实验分别的训练加速效果：验证集的 loss 变化趋势图 .....	34
表 3.1	电网温度预测模型中用到的主要特征说明 .....	13
表 3.2	列车轴温预测模型中用到的主要特征说明 .....	14
表 3.3	实验环境 .....	19
表 3.4	软件环境 .....	20

---

表 4.1	基于历史数据, 150、192、222 三组不同载荷下, 电机温度预测均方误差结果 .....	28
表 4.2	基于历史数据, 150、192、222 三组不同载荷下, PU 温度预测均方误差结果 .....	29
表 4.3	基于电气信号, 150、192、222 三组不同载荷下, 电机温度预测均方误差结果 .....	29
表 4.4	基于电气信号, 150、192、222 三组不同载荷下, PU 温度预测均方误差结果 .....	29
表 5.1	重要性采样的效果: 数据随机划分下, 全连接网络测试集上的 MSE 结果	32
表 5.2	重要性采样的效果: 数据随机划分下, 长短时记忆网络测试集上的 MSE 结果 .....	33

## 符号和缩略语说明

IoT	物联网 (Internet of Things)
EMD	经验模态分解 (Empirical Mode Decomposition)
iid	独立同分布 (Independent and identically distributed)
MSE	均方误差 (Mean Squared error)
PCA	主成分分析 (Principal component analysis)
LSTM	长短时记忆网络 (Long short-term memory)



## 第 1 章 引言

### 1.1 研究背景与意义

当今世界的数字化程度不断提升，在 2020 年，我们每个人平均每秒会产生 1.7MB 的数据，我们每天正在产生 2.5 亿字节的数据，仅仅在过去的两年，就产生了迄今为止世界上 90% 的数据<sup>[1]</sup>。在快速发展的技术的推动下，我们身边产生的数据不断地增加。其中，工业物联网 (Industry Internet of Things) 技术的高速发展，为我们利用蓬勃发展的机器学习和大数据技术揭示设备性能变化趋势提供了全新手段。

工业装备在人类社会中扮演着极其重要的角色，因此其安全稳定运行的意义极为重大。当前，每年因为机器故障直接造成的损失就达到 6400 亿美元<sup>[2]</sup>，甚至超过了集成电路行业的总产值。利用工业人工智能技术，通过数据驱动的方式重建机理，将预测式分析手段应用于装备部件、系统甚至是整车级别的故障预测、推理和剩余寿命估计，全面地提高设备的健康管理水平准确预测设备故障，准确采取预防措施，能够大大地降低装备的运行成本，保障装备运行的稳定性和安全性，对制造业的跨越式的发展有着极其重要的意义。

然而，现代装备的复杂性和高集成性导致了故障表现形式的复杂化，而且由于设备装态受环境和工况影响大、运行场景多变，对人工智能技术的成果应用提出了重大挑战。一方面，工业数据往往存在故障样本稀少的问题，如何充分利用反映正常运行模式的数据建立故障检测和预测模型，仍然是突出的问题；同时，工业 AI 需要构造针对不同问题和场景的多种模型，如何提高模型构造速度，也具有极大的紧迫性。

针对历史 IoT 数据进行预测具有十分重要的意义。一方面可以通过数据预测未来趋势，从而进行优化决策，例如针对风力发电的风力预测；另一方面，时间序列预测也可以用于实时故障检测，由于工业数据往往存在故障样本稀少的问题，可以通过建立基于正常数据的预测模型，然后通过分析预测模型（反映正常运行模式）和实际数据的残差进行故障预测。通过数据驱动的方式重建机理，将预测式分析手段应用于装备部件、系统甚至是整车级别的故障预测、推理和剩余寿命估计，能全面地提高设备的健康管理水平，大大地提高运维效率，和装备的出勤率，节约运维和设备运行的成本和效率。

## 1.2 问题描述与主要挑战

### 1.2.1 问题描述及研究目标

#### 问题描述

本课题针对工业装备状态这一类时间序列的预测问题，研究如何训练高准确度的模型，及训练过程的加速问题。

时间序列是按照时间顺序对特定过程进行观测而形成的数值序列。时间序列分析的目的在于通过挖掘反映数据变化规律的模式，在理解时间序列的基础上实现分类和预测，从而支持针对相应自然和社会现象的决策。高效能时间序列分析在金融、生物信息、自然灾害预测、过程控制等方面具有极其重要的作用。

#### 研究目标

时间序列的预测问题的典型解决方法是应用传统的概率统计模型，由于其准确率高、模型相对简单、鲁棒性好等优点，应用广泛。但是传统方法特征提取自动化程度低、验证依靠经验和人工分析。面对海量数据，深度学习具有自动化特征提取、对机理和人工依赖程度较低的优点，成为未来的发展趋势。因此，本文采用基于深度学习的时间序列预测框架。

然而，现有的深度学习方法的有以下三个问题：

1. 难以跟踪快速跳变数值。深度神经网络在回归问题上往往难以适应快速变化的数据，导致平均精度高、但是极值点附近误差较大的问题，然而极值点附近又往往是预测的重点。
2. 模型训练速度慢。在用深度学习处理时间序列问题中，模型优化的计算量大，需要的样本数量多，训练速度慢。导致在很多场景下，深度学习在时间序列上的应用相较于传统方法，并没有明显的优势。
3. 模型定制要求高。和深度学习应用到图像分类问题中不同，时间序列问题中数据之间的分布并没有显著的相似性。如果训练一个全局的深度学习模型，在具体的某一时间序列数据上，模型的效果可能会特别差<sup>[3]</sup>。时间序列问题的模型迁移能力差，无法将神经网络应用于各类不同的任务。工业数据情况复杂多变，往往需要针对问题定制模型，甚至每台机器都需要单独的模型。

深度神经网络的训练时间很长，加之需要多个不同的模型。训练速度极大地限制了深度学习在时间序列问题上的发展。针对以上提出的现有的深度学习方法存在的问题，本课题提出了两个的研究目标：

- (1) 利用正常数据训练出满足实际应用需求、并能发现故障的时间序列预测模型；
- (2) 提高时间序列模型训练效率。

### 1.2.2 时间序列预测的挑战性

自然和社会现象的错综复杂决定了时间序列的复杂性，本课题针对的工业装备 IoT 数据更具有以下鲜明特点：

- (1) 动态性：工业数据往往具有时变和非稳态的特点，即数据背后的统计分布和规律随时间变化，同时各种突发事件、偶然因素的影响也会造成非趋势性和非周期性的不规则变动。
- (2) 多样性：需要多种模型。时间序列问题的数据之间的分布并没有显著的相似性。如果训练一个全局的模型，在具体的某一时间序列数据上，模型的效果可能会特别差。时间序列问题的模型迁移能力差，无法将神经网络应用于各类不同的任务。工业数据情况复杂多变，往往需要针对问题定制模型，甚至每台机器都需要单独的模型。
- (3) 小样本：工业装备 IoT 数据的典型特点是故障类型呈长尾分布，正常运行样本极为丰富，特点故障样本稀少。样本的严重不均衡导致故障和非故障的分类十分困难，传统的监督式分类方法在这种情况下并不适用。
- (4) 高维度：一方面，随着各方面硬件技术的不断发展，实际应用中数据的采样频率不断提高，因此时间序列的长度也不断变大，仅仅把时间序列看作单纯的一维向量数据来处理不可避免地会带来维数灾难等问题；另一方面，很多实际应用中的时间序列数据往往包含多个变量（multivariate），这些变量之间往往存在复杂依赖关系。

### 1.3 研究内容与主要贡献

以电网数据和告诉机车数据为研究对象，针对时间序列回归预测问题，研究高准确度预测模型以及加速训练技术和方法。针对章节1.2.1中提出的两个研究目标，研究内容分为以下几个部分。

- (1) 针对时间序列预测问题，调研和应用了深度神经网络模型结构和方法。
- (2) 设计合理的模型训练框架，对时间序列模型的训练进行了加速。
- (3) 对具体的电网问题和轨道车辆走行部问题，对数据进行了分析和处理，搭建了时间序列模型训练和预测的系统。

完成了具体的研究内容，本文的主要贡献如下：

1. 针对牵引系统设备温度监测，设计了基于历史信息和基于电气信号两种不同的工业装备预测模型；
2. 构造了基于以上模型的完整预测方案，首先应用 LSTM 和 CNN 搭建时间序列预测网络，利用经验模态分解进行信号分解，对不同频率信号进行分别预

测；并且进一步对高低频数据采用不同的采样频率；尝试应用自编码器进行相关特征处理。

3. 研究利用重要性采样的方法对模型训练进行加速，设计和搭建了通用的加速框架，并且进行了实验验证，能够缩短训练过程 50%。

## 1.4 组织结构

本文总共分为六个章节。

第 1 个章节引言中，对问题本身进行了阐述，包括研究背景与意义，和问题描述与挑战分析。

第 2 个章节国内外研究现状，主要是对预备知识的阐述和研究现状的调研与分析。

第 3、4、5 是文章的主要工作内容的介绍。这三个章节中，第 4、5 章介绍了本文的两大块的主要内容，分别是工业装备的预测，和利用重要性采样对训练过程进行加速。在第 3 中则对这两部分内容中实验的设定和数据部分进行了说明，数据部分包含数据说明、数据特点分析和数据预处理几大部分。

在最后第 6 章中，对全文的研究做了总结，并提出了未来的展望。

## 第2章 国内外研究现状

### 2.1 引言

本章节对研究内容相关的文献进行了调研，对研究现状进行了阐述、分析和总结。同时为了方便理解研究内容、方便阅读，也对涉及到的一些相关的背景知识进行了介绍。

### 2.2 时间序列预测方法

针对章节1.2.1中提出的第一个研究目标：训练出准确度能满足实际应用需求的时间序列预测模型。这里调研了常用的时间序列预测的模型和方法，和一些提高预测效果的手段，同时也对不同模型的特点进行了分析。

#### 2.2.1 时间序列预测常用模型

处理时间序列的常用方法包括两类，一类是传统的统计模型例如 ETS 和 ARIMA，第二类是基于深度神经网络。

基于深度神经网络的模型预测方法近年来越来越具有竞争力。传统的统计学模型依然具有其兼顾准确率高、模型相对简单、鲁棒性好、高效等优势。然而，更复杂，更高维度和以及包含噪声的现实世界中的时间序列数据无法用带有参数的解析方程来描述，因为动力学太复杂且未知，传统浅层方法因为只包含一个小的非线性操作的数量，没有能力准确地模拟这种复杂的数据。从未标记数据中学习特征的优点是可以利用丰富的未标记数据，并且可以学习比手工制作的特征更好的特征。这两个优点都减少了对数据专业知识的需求。

近年来，在时间序列预测领域，也有关于标准框架的基础工作支持深度学习在时间序列预测问题上的发展。出现了为时间序列预测研究设计的 GluonTS 开源框架<sup>[4]</sup>。

常用的深度神经网络包括循环神经网络（如 LSTM 和 GRU）、序列模型、卷积神经网络、Transformer 和 Attention 模型。M4 竞赛结果表明<sup>[5]</sup>，准确率最好的模型都结合了传统统计模型和深度学习模型，或多种深度学习模型。在时间序列预测问题上，没有哪一种绝对优势的模型，各种模型都有其结构上的优势特点。不同模型的优势特点如表2.1。深度学习的模型对特征的拟合能力比较好。其中全连接网络是最基础的一类模型，原理比较简单，全连接的结构没有对问题有额外的

传统统计模型	ETS和ARIMA等	简单、鲁棒性好、高效		
深度学习模型	全连接网络	不同变量的相关信息，提取特征	原理简单	模型参数量大，训练效果不好
	循环神经网络		记忆能力	不具有长时依赖
	卷积时间网络		时间不变性	固定步长
	序列模型		内在相关性	特征维度不高、特征之间相关性不强时作用不大
	Transformer和Attention模型		长时依赖	模型复杂

图 2.1 应用于时间序列预测的典型模型类型及其特点总结

前提假设，比较普适，网络的拟合能力也好，但是问题是全连接的结构参数量相对来说会比较大，容易过拟合，训练起来也会比较困难。循环神经网络具有一定的记忆能力，但是在长时依赖的处理上往往不够好。卷积神经网络能够捕捉特征的时间不变性，但是往往受限于固定的步长。序列模型能够捕捉特征的内在相关性，但是在特征维度不高、特征之间相关性不强时作用不大。Transformer 和 Attention 的模型是最近几年新提出来的模型，它在众多任务上的效果都很好，但是模型参数量大，对训练资源的要求比较高。

### 循环神经网络在时间序列预测问题上的应用效果及问题

DeepAR 提出基于 LSTM 的时间序列概率自回归预测模型<sup>[6]</sup>，用 LSTM 模型建模未来时间步中的贝叶斯模型。近年提出的模型中，结合了指平滑方法的混合 RNN 模型<sup>[7]</sup>在 M4 预测竞赛中成为最终的优胜模型<sup>[5]</sup>。循环神经网络由于其独特的记忆机制<sup>[8]</sup>，一直都是处理时间序列问题的最通用选择之一，常用的模型结构如长短时记忆 LSTM 和门控循环单元 GRU。

GRU-ODE-Bayes 网络<sup>[9]</sup>构建了一个连续版本的门控循环神经网络，利用贝叶斯的方法更新，在多项任务中取得了很好的效果。

但是循环神经网络也有其缺点，比如对并行训练的支持不好、不能捕捉过长的时序依赖关系等。所以人们也一直没有放弃对其他网络结构在时间序列预测问题上的探索。

### 序列模型能有效捕捉相关性及构造时变模型

常见的序列模型一般包括编码器和解码器两部分。能够捕捉时间序列数据之间的相关性。目前效果较好的序列模型是由多个 RNN 组成。

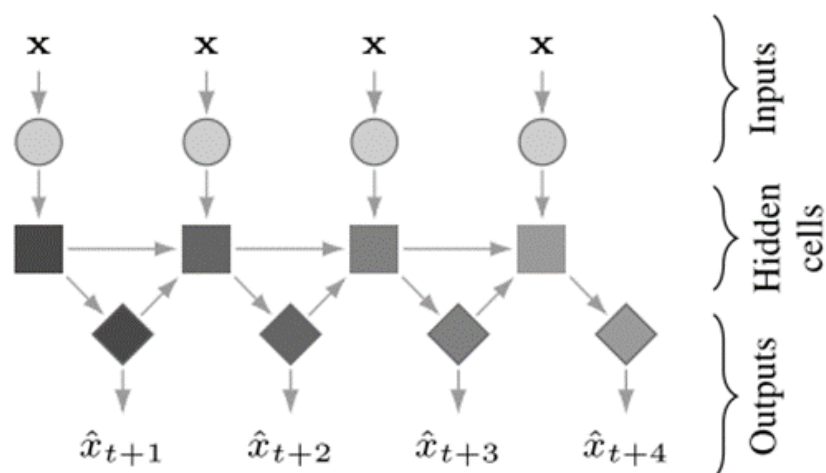


图 2.2 ForecastsNet 网络结构

ForecastNet<sup>[10]</sup> 中采用 Dense 连接的方式, 如图2.2所示, 实现了时变的模型结构, 同时通过交错输出的方式, 有效地缓解了梯度消失的问题, 在多步的时间序列预测问题上得到了不错的准确度。

### 应用于时间序列预测的卷积神经网络

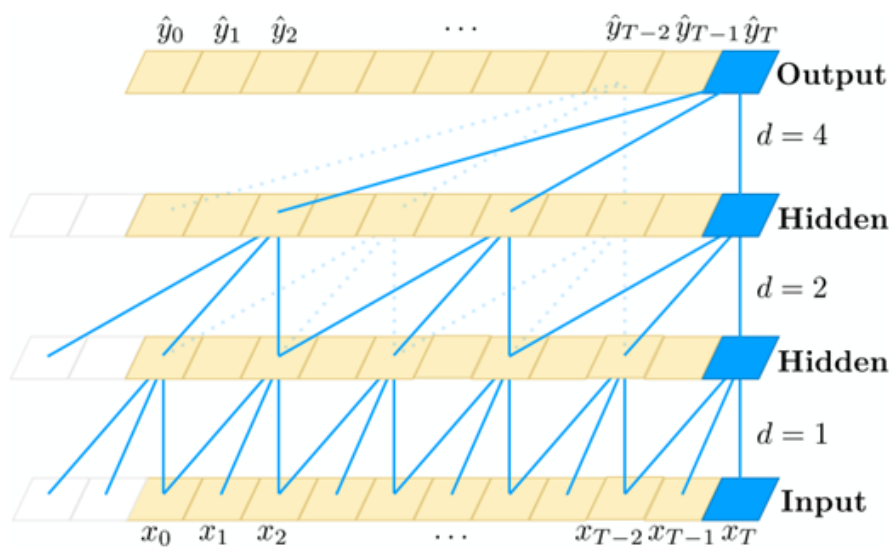


图 2.3 TCN 空洞卷积结构

早在上个世纪 80 年代末, 卷积的结构就被应用于序列问题上<sup>[11]</sup>。卷积神经网络通过卷积和参数共享来实现位移不变性的效果<sup>[12]</sup>, 能够应用在时间序列模型上, 具有时间不变性的特性。在建模序列数据问题上, 通过多个评价系统<sup>[13]</sup> 的测试, 达到了相当不错的效果, 明显优于循环神经网络<sup>[14]</sup>。卷积神经网络相比循环神经网络, 结构上支持并行, 计算速度方面能够得到有效地保证。也有相关工作<sup>[15]</sup> 结

合了卷积神经网络和循环神经网络，用卷积层替换了长短时记忆网络中原有的全连接层。

以 Temporal Convolution Networks (TCN) 为代表的更先进的卷积神经网络，结合了更大的卷积范围和残差跳跃连接，在序列建模任务的训练上，取得了很好的效果<sup>[16-17]</sup>。TCN 应用了空洞卷积的卷积结构，如图 2 中的感受野足够大，能够捕捉序列中的长时依赖信息，而且利用了残差连接，有效地提升了模型的准确率。

## 2.2.2 提高预测效果方法

### 经验模态分解

经验模态分解 (Empirical Mode Decomposition, EMD) 方法被认为是自 2000 年来，以傅立叶变换为基础的线性和稳态频谱分析的一个重大突破。

对于非线性数据尤其是非平稳数据，直接处理会非常难，但是这类数据就比较适合用经验模态分解来处理，经验模态分解在这类数据的处理上会具有明显的优势。

经验模态分解也常常会被用在信噪比低的数据上提升数据的信噪比。

## 2.3 对时间序列模型的训练加速

### 2.3.1 模型训练加速方法研究现状

时间序列模型的训练速度严重制约了深度学习在时间序列上的应用。模型的训练加速一般有特征、样本、模型几个方向的思路。对特征进行降维、筛选有效样本、利用多级模型在保证准确率的情况下降低使用的模型的复杂度。

另外图像领域对训练加速的研究对时间序列模型也有一定的借鉴意义。为了保证阐述的完整性，这里列出了其他的一些可以参考的图像问题加速的方法。以常用的卷积神经网络为例，常用的加速方法包括：(1) 张量分解，(2) 低精度的运算和其他量化方法，(3) 对大网络进行剪枝，(4) 用 Teacher-Student 方法通过训练小网络得到大网络，(5) 设计高效的网络结构，用启发式的方法找到最精简的网络结构，(6) 网络结构的自动搜索等<sup>[3]</sup>。近年来，研究者们将模型训练的效果提升和过程加速的注意力转移到重要性采样上来。

### 2.3.2 筛选有效样本 (重要性采样)

重要性采样 (Importance Sampling) 是统计学中估计某一分布性质时使用的一种方法。该方法从与原分布不同的另一个分布中采样，而对原先分布的性质进行估计。



重要性采样被应用于提升凸优化问题的随机优化方法的收敛速度上,例如构造核分类器<sup>[18]</sup>、用随机过程优化正则化损失<sup>[19]</sup>等。另外一方面,在深度神经网络领域,样本筛选的方法也被应用于解决困难样本的处理问题,例如在嵌入学习问题中<sup>[20]</sup>,也被应用于解决样本不平衡的问题。

深度神经网络花费了很多重复的计算量在处理很容易拟合的样本上,这些样本在训练一段时间后,就能被忽略掉,并不会影响最终的模型效果,在训练过程的某些时间开始忽略这样的样本,能大大地节省模型优化的计算量。也就是说,在训练过程中,不同的样本对收敛过程起到的作用效果是不一样的,很多样本在很少的几个轮次之后就能被很好地分类或者拟合,通过对样本的筛选,能够在较少影响准确率的情况下有效地加速训练过程。

近年来的一些工作表明<sup>[21]</sup>,重要性采样的方法的加速原理是降低随机梯度下降过程中梯度估计的方差,最优的采样分布应该和样本的梯度范数成正比。所以重要性采样一般围绕梯度范数展开,但是由于梯度范数的计算复杂度过高,另外一种常用的思路是通过训练过程的损失来对梯度范数进行拟合,从而计算样本重要性。通过损失来计算样本重要性的方法通常有很多超参数需要调优,并且对梯度范数拟合得不够。为了探索一种更好的训练数据选择策略,包括课程学习(curriculum)和自步学习(Self-paced Learning)在内的先前工作都采用了简单的启发式规则,例如打乱序列长度来训练语言模型,或放弃损失值大于手动定义的阈值的训练样本。这种人为定义的规则在一定程度上受限于某些任务,不能推广到更广泛的学习场景,因为不同的学习任务可能会产生不同的最佳数据选择规则,甚至一个学习任务可能也需要具有各种属性的数据才能在不同的训练阶段进行优化。

Yang Fang 等人应用强化学习的方法来选取样本<sup>[22]</sup>。通过强化学习来设计一个训练数据的过滤器(Neural Data Filter, NDF)插入到网络输入层之前,在训练的过程中动态筛除一部分对训练益处不大的数据,从而减少训练数据量。该方法有两个直观的原则:一方面,数据选择策略应该是通用的,这样就可以自然地将其应用于不同的学习场景,而无需进一步的人工设计。另一方面,该策略应具有前瞻性,因为它在训练的每个步骤中的选择都会带来更好的长期回报,而不是暂时适应当前阶段。

NDF的基本工作流程是:给定当前训练中的网络的状态和当前 mini-batch 数据的状态(states),输出一组动作(action),是一个 0,1 向量,它的维度和 mini-batch 的大小相同,每个元素代表了 mini-batch 中对应的数据是否应该被剔除。利用强化学习的方法来建立过滤器过程包括以下几个步骤,首先通过现有的采样的 policy,输入当前的模型状态和待采样样本状态,输出采样的结果,然后将样本输入模型

中，得到实际的梯度下降的效果的 Reward，继而用 Reward 对采样的 policy 进行优化。

### 2.3.3 重要性采样介绍

#### 蒙特卡洛积分

提到重要性采样首先要介绍蒙特卡洛积分的概念。

蒙特卡洛是一类算法的总称，是用随机抽样和统计模拟的方法，来进行数值计算。它的基本做法是，做大量重复实验来统计频率，根据伯努利大数定律，当样本数足够多时，频率会无限接近于概率，所以理所当然，可以通过频率来估计概率。

对于求解积分  $\int_a^b f(x)dx$ ，经典的方法是我们需要找出  $f(x)$  的原函数  $F(x)$ 。但是，在求积分的过程中，积分的原函数在很多情况下都不是很容易获得，那么我们就无法应用经典的求解积分的方法。

蒙特卡洛积分是蒙特卡洛算法的具体应用。蒙特卡洛方法在估计  $\int_a^b f(x)dx$  积分时，将其表示为一个均匀随机变量的期望，如下，

$$\theta = \int_a^b f(x)dx = (b-a) \int_a^b f(x) \frac{1}{b-a} dx = (b-a)E(f(X)), X \sim U(a, b) \quad (2-1)$$

其中  $U(a, b)$  代表在  $[a, b]$  之间的均匀分布。从而可以通过如下算法来得到积分估计的结果：

1. 从分布  $U(a, b)$  中产生 i.i.d 样本  $x_1, x_2, x_3, \dots, x_n$ ;
2. 计算  $f(X)$  期望的估计值  $\overline{f(X)} = \frac{1}{n} \sum_{i=1}^n f(x_i)$ ;
3. 得到  $\hat{\theta} = (b-a)\overline{f(X)}$ 。

容易得到，估计值  $\hat{\theta}$  的期望和方差分别为，

$$E\hat{\theta} = \theta, \text{Var}(\hat{\theta}) = (b-a)^2 \text{Var}(\overline{f(X)}) = \frac{(b-a)^2}{n} \text{Var}(f(X))$$

但是基于均匀分布的估计方法，不能应用于无穷积分的估计，而且当被积函数在积分区间上的分布不是很均匀时，抽样的效率会比较低。

#### 重要性采样

前面的蒙特卡洛积分经典计算方法采用均匀分布作为加权函数，会有抽样效率的问题，重要性采样就是一种利用合理的加权函数，提高抽样样本效率的计算方法。和经典的计算方法相比，重要性采样的加权函数不再是均匀分布。设随机

变量  $X$  的概率密度函数是  $g(x)$ 。记  $Y = \frac{f(x)}{g(x)}$

$$\theta = \int_a^b f(x)dx = \int_a^b \frac{f(x)}{g(x)} g(x)dx = EY \quad (2-2)$$

再通过简单的蒙特卡洛积分方法估计  $EY$ ：

$$\hat{\theta}' = E\hat{Y} = \frac{1}{n} \sum_{i=1}^n y_i = \frac{1}{n} \sum_{i=1}^n \frac{f(x_i)}{g(x_i)} \quad (2-3)$$

此处  $x_1, x_2, x_3, \dots, x_n$  为从  $g(x)$  中抽取的样本。用这个方法估计的参数的方差为  $Var(\hat{\theta}') = Var(Y)/n$ 。当  $Y$  为常数时方差为 0。所以， $f(x)$  的选择目标应该是尽量接近  $g(x)$ 。

## 第3章 时间序列数据分析与处理

### 3.1 引言

在本章节中，对实验的情况进行了具体的说明，交待了实验环境，实验参数等。

同时对数据相关的问题进行了阐述。对实验中用到的数据进行了说明，同时说明了数据预处理的流程和具体的方法，以及采用相关方法的动机、实际效果等。

### 3.2 数据及评价指标说明

本文中应用的数据主要包括电网数据和列车走行部数据，都是实时采集的实际场景中的工业装备状态数据。下面对这两种数据的具体情况分别进行了说明。

#### 电网数据说明

本文实验中采用的电网数据，是基于青岛永磁实验台，按线路条件（包括牵引、匀速、制动），跑典型工况，采集的相关数据。

数据在 150t、192t、222t 三种不同载荷下分别采集，在为 10Hz、100Hz、100kHz 三种不同采集频率的下，采集到不同特征列数据。

在表3.1中对各特征列进行了详细的说明。!!! 补充完整

#### 列车走行部数据说明

本文实验采用的数据包含列车底盘（又名走行部）的监测数据，包含各轴承温度、驾驶档位、输出功率、环境温度、当前轨道的地理位置情况等相关数据。

需要通过预测轴承温度的方式来判断轴承的运行状态。

本文中实验用的是车号为 5103 的列车的数据，在表3.2中对各特征列进行了详细的说明。在清洗完成后，还剩 88401 条可用数据，跨时 234 天，平均时间间隔为 228.6 秒，但是采样间隔分布比较不均匀，例如列车停运时采样间隔会比较长。

#### 评价指标

均方误差（MSE, Mean squared error）是比较常用的反映估计量和被估计量之间差异程度的一种度量。

但是均方误差作为评价指标也有它存在的问题。比较大的一个问题就是对异常值比较敏感，如果样本中有个别异常值出现，会对均方误差的影响比较大，导致

表 3.1 电网温度预测模型中用到的主要特征说明

特征	通道名称	采集频率	采集方式	特征说明
电机温度	AI_Motor2Temp1_gui	10Hz	PTU 采集	待预测特征之一
PU 温度	AI_PUTemp_gui	10Hz	PTU 采集	待预测特征之一
网流	AI_BusNegCurrent_gui	10Hz	PTU 采集	输入特征之一
U 相电流瞬时值	AI_PhaseUCurrent2_gui	10Hz	PTU 采集	输入特征之一
U 相电流有效值	AI_PhaseUCurrent2_RMS_gui	10Hz	PTU 采集	输入特征之一
W 相电流瞬时值	AI_PhaseWCurrent2_gui	10Hz	PTU 采集	输入特征之一
W 相电流有效值	AI_PhaseWCurrent2_RMS_gui	10Hz	PTU 采集	输入特征之一
环境温度	PT2_	100Hz	数采系统	输入特征之一
电机速度	Mot2_speed_rpm_gui	10Hz	PTU 采集	输入特征之一

评估的结果并不鲁棒。

针对这个问题，在数据预处理的数据清洗阶段做出了一些处理，去掉了一些明显的异常点。

### 3.3 数据特点分析

#### 3.3.1 工业装备状态数据特点

工业装备数据一般采集于实际的工业装备系统，由于系统本身十分复杂，机理很难分析，很难对其进行建模分析。而且由于数据本身会存在噪声干扰，特性也比较复杂，通过传统的时间序列的分析方法也很难对其进行建模和抽取特征。

#### 电网数据特点

1. 电网数据来源于几种不同的采集方法，不同采集方法的采样频率会有差异。
2. 如电压电流这类特征会有很明显的周期性。

表 3.2 列车轴温预测模型中用到的主要特征说明

特征	特征说明
采集时间	用于数据分段和序列处理
车号	
16个轴的测试温度	待预测特征。共6个轴，每个轴六个测试点。
环境温度 1、2	两个点采集到的环境温度，用于预测轴温的特征
主发电机温度	用于预测轴温的特征
风机 1、2 温度	用于预测轴温的特征

### 列车轴温数据特点

1. 数据波动十分剧烈。这份数据采集所在的区域所在的环境温度波动剧烈，如图3.1所示，昼夜温差很大，低温低于 15 摄氏度，高温高于 42 摄氏度。
2. 虽然在数据表中包含了地理位置信息，但是当前轨道的地理情况（坡度、弯度等）很难精确捕捉。
3. 根据采集时间来看，列车的运行并不是一个连续的过程，会存在停运的情况，而且数据采集的过程也不连续。所以不能直接作为一个长时间序列来进行处理，需要对数据进行分段。

### 3.3.2 时间序列数据平稳性分析

#### 平稳性分析的意义

平稳性分析能够确定模型是否具有统计分析的意义。虽然利用深度学习建模并不是基于统计来对数据进行处理，但是平稳性分析能够检测序列是否具备被预测的基础，例如白噪声就不能被预测。

这一个步骤主要是为了对待预测的数据的性质和预测难度有一个了解，方便后续选择合适的预测方法。

## 3.4 数据预处理

数据的预处理包括如下的几个主要部分：数据读入、数据清洗、数据重采样、数据分段、数据标准化、经验模态分解、数据窗口处理、划分训练集和测试集这些部分。下面会对这些内容分别进行阐述。视数据特点的不同，个别的流程可能可

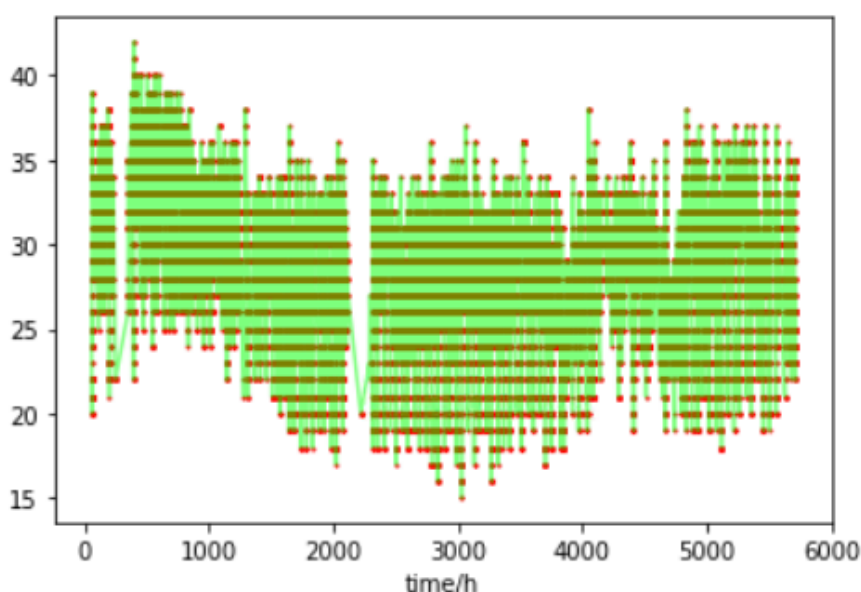


图 3.1 5103 号列车数据中环境温度变化图（横轴是时间，纵轴是环境温度，单位为摄氏度）

以省略。

### 3.4.1 数据导入及清洗

电网原始数据存储在 `tdms` 文件和 `csv` 文件中，把数据读入 `pandas` 之后进行处理。列车轴温原始数据存储在 `dmp` 文件中，需要通过 `oracle` 数据库工具进行导入。

在列车轴温数据中，导入后的数据存在问题。首先是存在着部分列的数据格式问题，打印出来发现是因为个别数据显示为“异常温度”或者是“-”。另外会有些数据的时间标记是默认的最小值的数据。用简单的处理数据缺失的方法来处理，由于数据量可观，可以直接在数据表中删除异常的样本点。

### 3.4.2 数据重采样

需要注意的是，这一步对电网数据和列车轴温数据需要进行的操作是不一样的。

在电网数据中，数据本身有固定的采样频率，采样间隔都是统一的，只是不同特征的采样频率不一样。

但是在列车走行部数据中，如图3.3中给出的采样时间间隔分布所示，采样点并不均匀的，就必须进行均匀间隔的重采样。

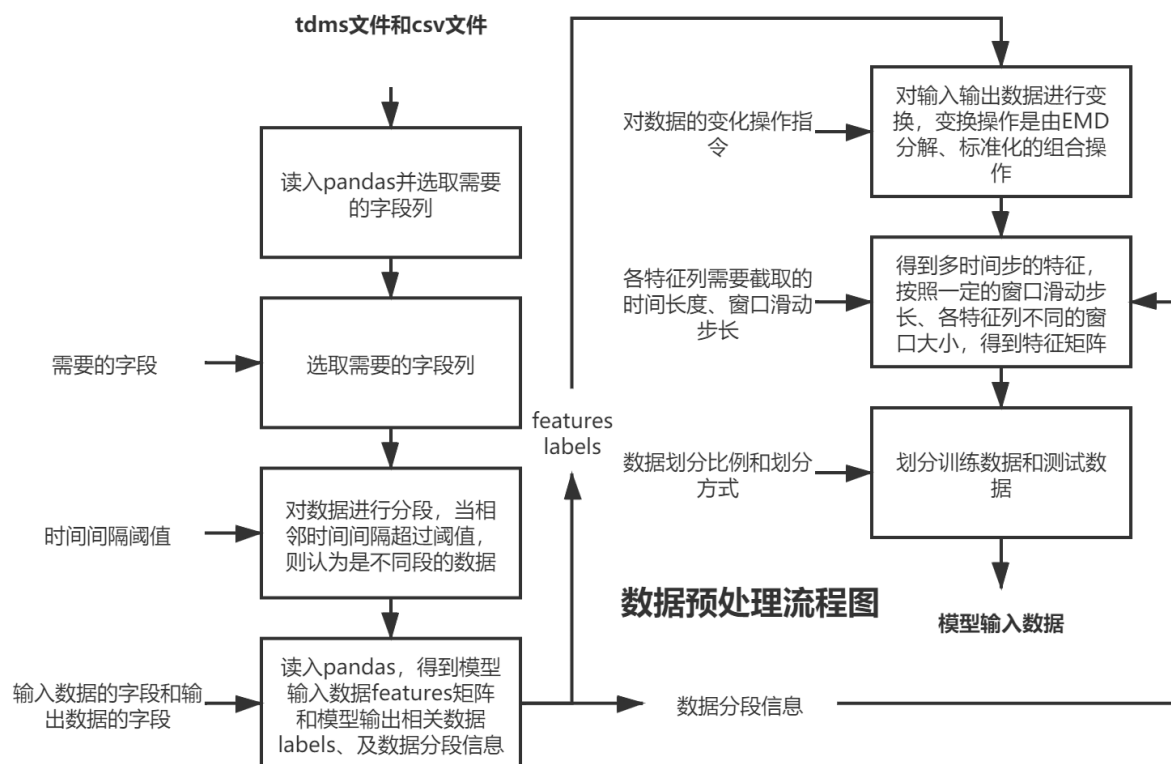


图 3.2 电网数据分段及划分处理流程图

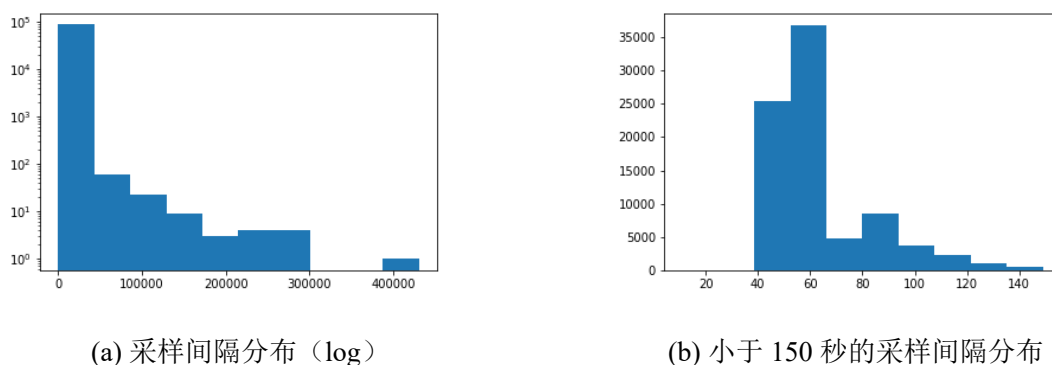


图 3.3 列车走行部数据采样时间间隔分布直方图（横坐标：间隔 (s)，纵坐标：频次）

### 3.4.3 数据分段

首先介绍为什么需要对数据进行分段，在列车走行部数据中，如图3.3中给出的采样时间间隔分布所示，采样点是十分不均匀的，间隔可能长达 40000 秒（约 1 天左右），也可能很短，50 秒左右。如果只是进行简单的重采样，如果重采样的采样间隔过短，那在原始数据采样间隔很长的地方，会造成大量的数据冗余，对模型



的训练也并没有好处；如果重采样的采样间隔过长，又会导致信息的过度丢失。

所以这里提出一个数据分段的解决方案，通过设定一个合理的时间间隔阈值，如果两个点之间的时间间隔过长，我们则认为它们属于两个不同的数据段。

关于时间间隔阈值的选择问题。可以将所有数据的时间间隔分布用直方图统计出来，再根据统计出来的结果选择出一条比较明显的分界线。

为了找到重采样的合理阈值，这里对采样间隔进行了如3.3直方图分析，大部分的采样间隔都集中在150秒以下，通过如3.3(b)中对150秒以下的间隔的进一步分析可以得到，70秒是一个比较合理的阈值。

#### 3.4.4 整理数据表并通过滑动窗口取模型输入

在清洗完数据并对数据分好段之后，把需要用到的数据整理成一张二维的数据表。每一列是一个属性，每一行是一个采样点的数据。!!! 滑动窗口示意图

#### 3.4.5 数据标准化

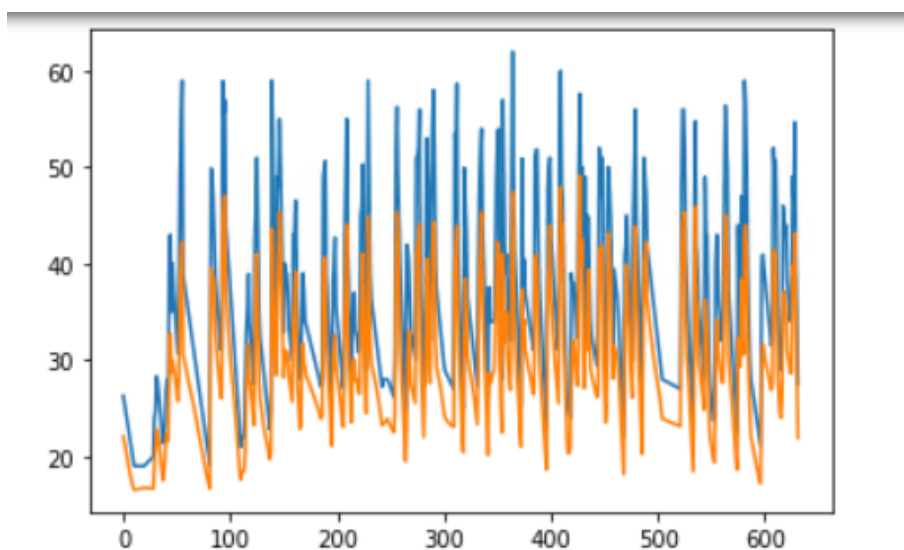


图 3.4 数据标准化前预测效果示意图

数据预处理过程需要对数据进行标准化。如果没有标准化，会出现的问题是，预测趋势还是对的，但是对高温和低温的极值附近的预测很差，Accuracy 结果比较好但是 MSE 差，这也是由于 MSE 的结果会比较受个别值的影响。如图3.4所示。

采用的标准化的方式是普遍被应用的控制均值和控制标准差的方式。具体的公式如3-1所示，数据标准化的处理流程如3.5所示，需要将预测结果反标准化之后

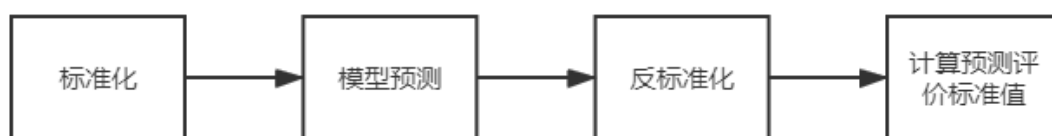


图 3.5 数据标准化流程

再进行应用和计算评价标准的值。

$$z = (x - u)/s \quad (3-1)$$

其中， $u$  是  $x$  的平均值， $s$  是  $x$  的标准差。

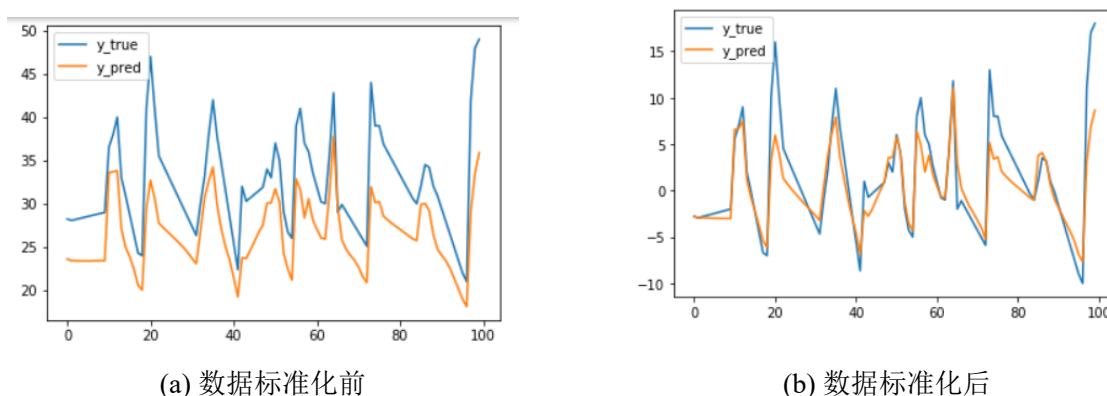


图 3.6 数据标准化效果说明图

为了更清晰的对比，在3.6中，截取了预测结果的片段进行对比。标准化前预测效果如3.6(a)所示。在加入标准化处理后，预测效果有了明显地改善。标准化后预测效果如3.6(b)所示。

### 3.4.6 不同序列划分方式

训练集和测试集有如下两种不同的划分方式：

1. 第一种是序列化的顺序划分方式。以 7:3 的划分比例为例，即按照时间顺序，前 70% 的数据作为训练集，后 30% 的数据作为测试集。
2. 第二钟是随机划分的方式。同样以 7:3 的划分比例为例，随机取 70% 的样本点作为训练集，剩下 30% 的样本点作为测试集。

#### 序列化的划分方式下的过拟合问题

不同的序列划分方式会导致实验结果有很大的差别。在序列化的划分时，很容易出现在训练集上效果很好，在测试集上效果不好的情况。图3.7是在序列化的

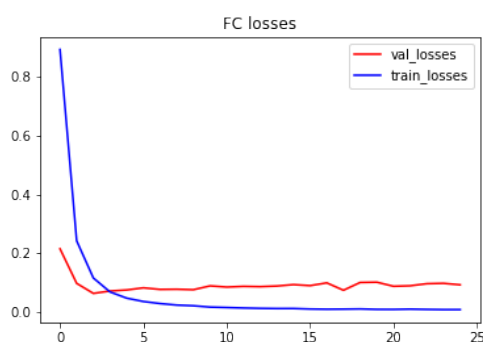


图 3.7 出现过拟合问题的网络训练过程 loss 变化

样本划分方式下，应用全连接网络，训练得到的训练和验证集上的 loss 变化情况。如图3.7，很明显训练出现了过拟合问题。分析原因是因为序列化的划分方式中的训练集和测试集，会由于工况的区别，有着很大的差异，给模型效果带来了挑战。

但是在随机化的划分方式中则不容易出现这种问题。这也很容易理解，这和训练集和测试集的样本分布的相似度有关，随机划分的训练集和测试集的样本基本同分布。

### 采样两种划分方式实验的原因

这里之所以采用了两种划分方式进行实验，也是因为这两种划分方式各有其特点和不可替代的实际意义。序列化的划分方式更加接近实际中使用的场景，我们的模型在实际使用中，不可能将未来的数据拿来训练。但是由于资源的限制，实验用到的数据有限，而研究对象的情况会比较复杂，所以实际情况中会使用更多数据来训练模型，理论上当数据足够多时，会涵盖更多的数据情况，模型的应用效果也会更加接近随机划分训练集和测试集。

实际的情况下模型的应用效果可能介于两种划分方式之间，所以这里对两种数据划分方式都展开了研究。

## 3.5 实验环境

表 3.3 实验环境

环境	版本
操作系统	Ubuntu 18.04.3 LTS

本文的实验都是在 Linux 系统下搭建完成，语言是 Python，利用 GPU 进行模

表 3.4 软件环境

软件环境名称	安装包版本信息	安装来源
Cuda	V10.0.130	官网
Python	3.8	Conda
Tensorflow	2.4.1	pip
Pandas	1.2.2	pip
Numpy	1.19.5	pip
Sklearn	0.24.1	pip
EMD-signal	0.2.13	pip

型的训练与测试。

深度学习框架方面，本文的实验都基于 Tensorflow 版本 2.4.1 进行，应用了 Tensorflow2 最新的 Keras 接口。具体的实验环境和安装包配置如表3.3和表3.4所示。

### 3.5.1 实验说明

需要注意的是，为了消除随机因素的影响，如果没有特别说明，那文中给出的实验结果都是基于三次重复实验的平均结果。

### 3.5.2 选用 Tensorflow2 编写的优势

首先对比下动态图和静态图的特点。旧版本的 Tensorflow 一直采用的是静态图的机制，而被学术界研究人员广泛应用的 Pytorch 是基于动态图的机制。静态图是指，在图被构建之后，在模型运行之时，计算过程中，图是无法修改的。动态图反之，在计算过程中可以对图进行修改。由于静态图计算过程中图无法修改，所以可以在运行前进行一些优化例如融合部分 Operation。但是也会因为图融合等原因，静态图在调试方面具有较大的难度，无法提供断点和单步调试功能，而且也会非常的不直观。

Tensorflow 不仅在部署上相对于其他框架有着独特的优势，而且 Operation 全面，生态良好。在升级 Eager 模式后，能够兼顾静态图运行效率快，和动态图代码简洁方便调试的优点。

## 3.6 本章小结

## 第4章 工业装备状态预测

### 4.1 引言

本章节介绍了工业装备状态预测的相关内容，包括整体方案的设计，和应用的具体方法，详细地说明了设计的动机、和实际的效果。

工业装备状态预测的一个重要的应用就是将其应用到预测式故障检测中。

故障检测的一个很直接的思路是，将问题定义成一个分类问题，分类结果分别是正常和存在故障两种状态，这样也能通过历史数据训练神经网络模型，得到我们想要的结果。但是在工业装备状态预测下，作为分类问题建模存在的问题是样本极度地不平衡。设备发生故障的情况相对正常情况会很少，甚至是从来没有出现过故障。在这样的情况下，分类的难度会特别大。

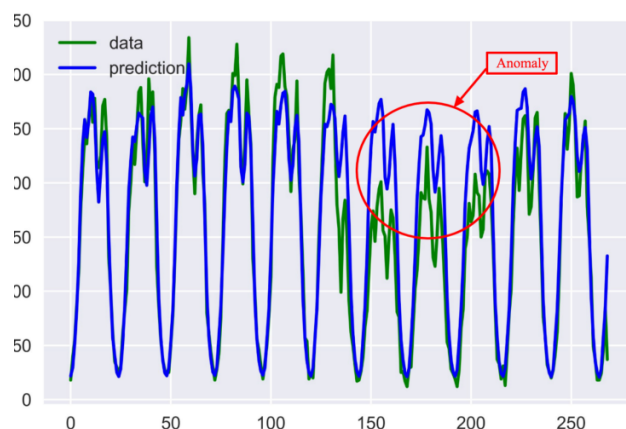


图 4.1 预测式故障检测示意图

如图4.1中所示，当实际采集到的工业装备状态数据和预测数据发生较大的偏差时（图中红圈部分），我们就可以认为是装备发生了故障。这个前提是需要被检测到的故障会引起相应检测状态的变化，例如在很多装备或者电子元器件中，如果发生温度偏离正常运行状态过高，就可以认为是发生了故障。

以列车走行部作为状态预测为例，可以设计如图4.2的故障预测系统。分为数据整理、数据预处理与分析、模型预测、应用到实际系统调试几个模块。

为了能够达到能够上车调试的效果，需要对模型进行不断地迭代和优化，直到预测的效果能够达到某一个特定的阈值。这部分我们只关注预测的结果，可以和上车调试部分独立出来，成为一个单独的模块。在下面的部分，我们只讨论工业装备状态预测的部分，主要关注如何提升预测的效果，减小预测结果和实际结果之间的偏差。

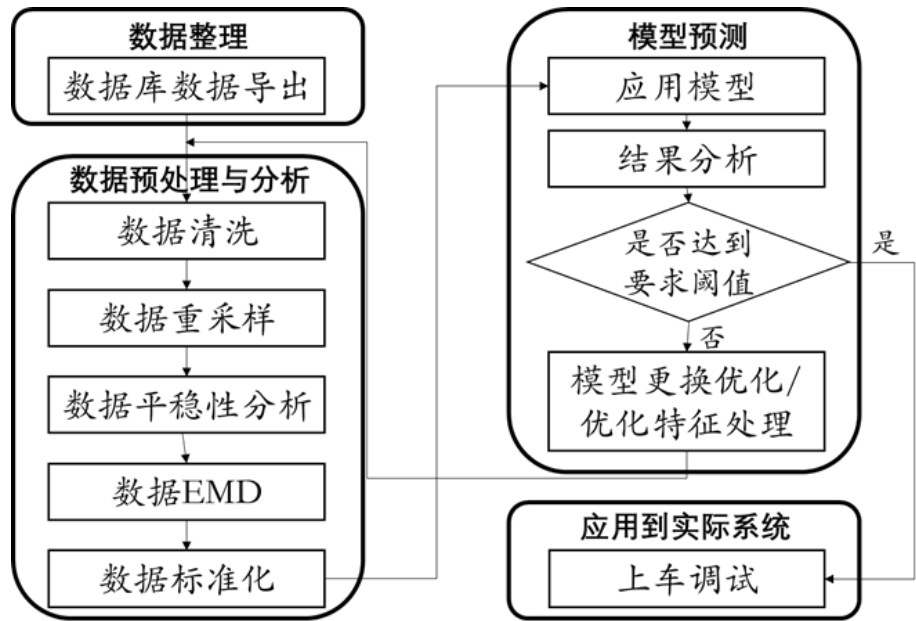


图 4.2 列车轴温预测系统流程框架图

4.2 工业装备状态预测方案

4.2.1 系统设计

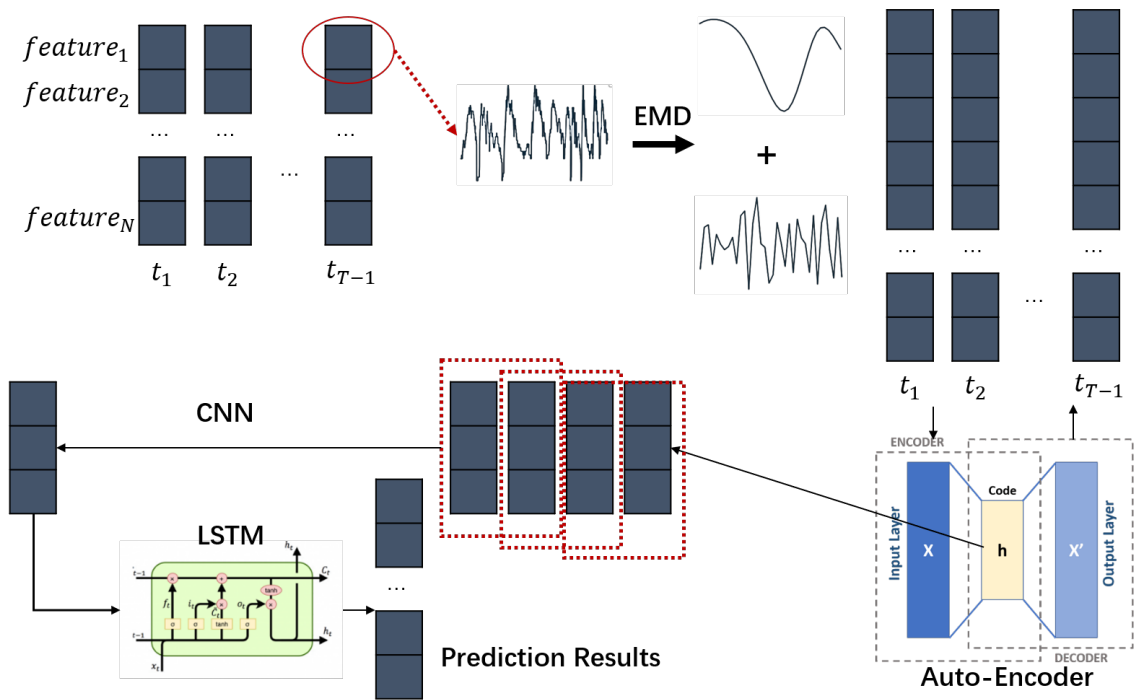


图 4.3 工业装备状态预测部分系统设计

如4.3图中所示，这里给出了整体预测方案的系统设计方案。

1. 针对极值部分数据突变的预测技术。由于极值部分变化剧烈，往往预测效果不好。解决这个问题一个方案就是对信号进行频率分解，这样输入的特征

中会包含不同频率的信号特征，能够更好地预测极值部分的数据值。所以应用经验模态分解对时间序列进行信号处理，提取出不同频率的信号分量。

2. 针对序列难建模的特点，提出应用 CNN+LSTM 组合的模型结构。
3. 针对工业序列噪声大、特征之间相关程度高的特点，利用 Auto-Encoder 部分进行特征相关性的提取。

### 4.2.2 方法说明

在这一个小的章节中，将会对工业装备状态预测方案中采取的具体方法进行分别的说明。

#### 经验模态分解信号处理

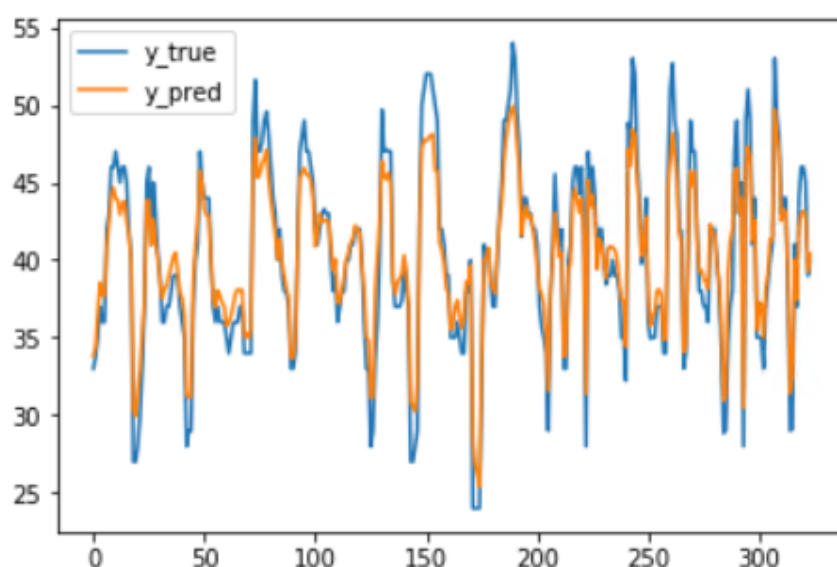


图 4.4 模型在高频处会预测效果差示意图

通过实验观察发现，如图4.4中所示的模型预测结果示意图（其中  $y\_true$  是真实值， $y\_pred$  是预测值），模型预测结果的低频分量效果较好，但是容易在尖峰等轴温快速变化的位置预测效果相对较差。于是想通过 EMD 分解，可以提取不同频率的信号特征，进而提升模型预测效果。

#### 对高低频数据不同采样频率

为了进一步优化模型，在简单 EMD 方法的基础上，又进行了一些改进。由于 EMD 方法会将数据分解成多个不同的频率的数据，对于低频数据，采样太过密集会导致数据过于冗余，而对于高频数据，采样过于稀疏会导致高频部分数据失真。所以很容易想到，我们可以对不同频率的数据在输入模型前进行不同的采样频率



的处理，在减少特征维度的同时，尽可能地提升数据的质量和模型预测效果。

### 使用 Layer Normalization

需要注意的是，由于后续需要利用重要性采样进行训练加速，为了统一，模型中统一使用 Layer Normalization 取代 Batch Normalization。这是因为，在重要性采样中会根据样本的重要性进行采样，那么一个 batch 的样本不再是从原分布中随机产生，不再满足和同分布的特点，那么 Batch Normalization 对于不同的 batch 的效果将不会是一致的。会导致训练出现问题。

### 不同时间序列预测的模型方案对比

本文对比了几种不同模型在实验数据上的预测效果，下面分别介绍这几种模型的结构。

本文被用来对比的实验基准网络是一个全连接网络，

本文实现了长短时记忆网络，

通过实验测试了这几种不同网络的实际效果，

### 自编码器在特征提取上的应用

以上考虑了多个时间步上特征提取的模型结构问题，但是在单个时间步上的多个特征之间，可能存在很强的相关性，而特征的相互关系并不能用以上的模型结构表述，需要进一步处理。这里提出利用 Auto-Encoder 结构捕捉特征之间的相关性。

需要注意的是，Auto-Encoder 不需要完整地嵌入网络中，只用将 Encoder 部分加入预测网络中，将 Encoder 的输出值输入其余部分的网络。而且，这部分的网络无需参与后续的整个网络的梯度更新过程，只需要单独进行训练。

应用自编码器主要有两方面的目的，一方面自编码器自带的结构特征可以用来进行特征提取和降维，另外一方面，自编码器具有降噪的效果，工业装备状态数据是实际采集到的数据，难免会存在很多噪声，通过自编码器的应用，能够有效地减少噪声对模型的干扰。

那这里为什么选择用自编码器来进行降维，和常用的降维方法例如 PCA (Principal component analysis, 主成分分析) 方法对比。PCA 方法只适用于线性降维，其他的降维方法也会受到数据的局限性，但是自编码器降维的好处是利用深度神经网络来进行建模，对数据的拟合能力很强。但是，自编码器降维在数据量比较小的时候容易出现过拟合的问题，所以相对于 PCA 方法而言，自动编码器更适用于复杂的大型数据集。本论文讨论的主要针对工业装备状态这种长时间数据，采集

到的数据很多，所以不用担心过拟合的问题。

### 自编码器效果

通过实验我们发现，不是所有的工业装备状态预测的数据都适合用自编码器在特征维度进行降维。例如在火车走行部的轴温预测中，各个不同的轴的温度会同步变换，之间有比较强的相关性，而且这种相关性独立地存在于当前各个轴的温度状态中，不需要额外的历史信息，解码器就能恢复出数据。

所以判断是否能够应用自编码器进行特征提取，我们可以通过训练的自编码器的输入和输出的比较，如果差别比较小，说明编码器部分的信息损失比较小。

当然当出现自编码器效果不好时，也可以通过尝试别的不同结构的自编码器进行处理，例如这里可以尝试编码器和解码器都是更加适用于序列处理的网络结构，例如 LSTM 模块，但是在这里不作过多展开和研究。

### 防止过拟合的策略

针对在3.4.6中提出的，在序列化的划分方式下出现的过拟合问题，采用了加 dropout 的方式进行解决。在后面应用 LSTM 时，也采用了同样的加 dropout 方式，同样也不存在过拟合问题。

### 4.2.3 两种不同的状态预测方法

对于状态预测有两种方式，一种是通过历史数据回归，另外一种结合历史数据和当前的其他可获得的电气信号预测。这两种预测方式都有其各自的特点和适用的应用场景。

第一种通过历史数据回归的状态预测方法，它的应用场景更为广泛，第二种需要获得当前的其他电气信号，有的场景下受到传感器数据收集速度或者数据传输速度的限制，即时的信号并不那么容易获取，第一种方法就不会有这样的限制。

但是第一种方法也会存在相应的问题，有的时候会发生工况等状态的变化，这样情况下简单通过历史的信息回归信息是不完备的。

## 4.3 实验结果

### 4.3.1 往前预测多步结果

预测的效果和设定的往前预测的时间长度相关。自回归往前预测第  $i$  步 ( $i$  取 1-160) 的均方误差如图4.5中所示，可以看到，往前预测得越远，均方误差越大，预测效果越差。

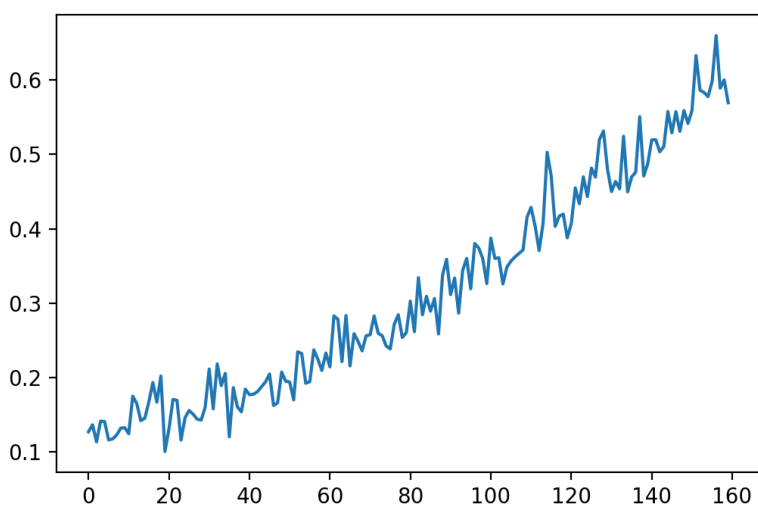


图 4.5 往前预测多步均方误差（横轴坐标值代表往前预测的步数，纵轴是均方误差）

### 4.3.2 防止过拟合的策略实验结果

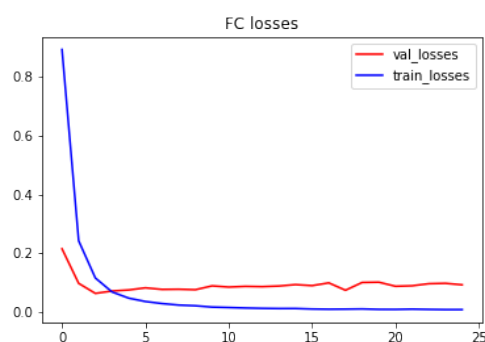


图 4.6 出现过拟合问题的网络训练过程 loss 变化

如图4.7(a)中所示，对比前面图4.6中的结果，不再出现训练一段时间后训练集 loss 虽然一直在下降，但是验证集的 loss 反而再上升，而且训练集的效果远远好于测试集效果这样的情况。在模型过拟合方面有了明显改进。说明加 dropout 模块的方法十分有效。

### 4.3.3 两种不同方法的工业装备状态预测的结果

下面对两种不同方法分别进行实验，每种方法分别应用到电机温度预测和 PU 温度预测两个场景下，给出了在三种不同载荷情况下，训练集的 MSE 和测试集的 MSE 的最终结果。

4.1和4.2 给出的是第一种方法—基于历史信息的预测方法的结果，4.3和4.4 给

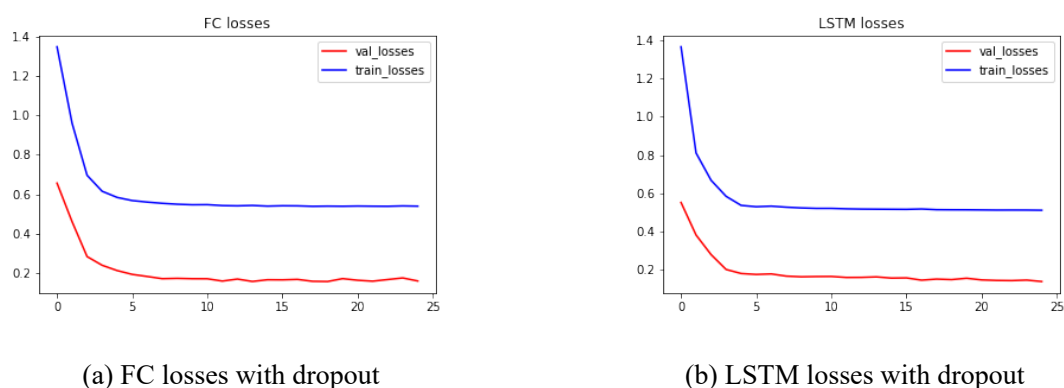


图 4.7 过拟合策略效果图：加了 dropout 之后的模型 loss 曲线变化

表 4.1 基于历史数据，150、192、222 三组不同载荷下，电机温度预测均方误差结果

载荷	训练集 MSE	测试集 MSE
150	0.357	0.334
192	0.328	0.358
222	0.197	0.190

出的是第二种方法—基于电气信号的预测方法的结果。

通过上面的结果可以看到，基于电气信号的方法的训练集 MSE 明显更小，而且差别比较悬殊，差了 10 倍以上，说明这样的方法的确提供的信息更完备。

但是测试集的效果却反而不是很好，在基于电气信号的方法中，虽然数据提供的信息量更多，但是模型效果反而更差。分析原因是因为过拟合导致的，数据中的噪声也会导致这样的问题，加上 MSE 的数值本身就容易受到个别异常点的影响，所以导致了测试集 MSE 的结果比较差。

#### 4.4 本章小结

在本章中讨论了关于工业装备状态预测的相关问题，主要目标是搭建一个预测效果好的深度神经网络模型，对不同的模型结构进行了研究和对比，同时提出了一些提高模型预测效果的方案。

表 4.2 基于历史数据，150、192、222 三组不同载荷下，PU 温度预测均方误差结果

载荷	训练集 MSE	测试集 MSE
150	0.011	0.017
192	0.020	0.022
222	0.038	0.037

表 4.3 基于电气信号，150、192、222 三组不同载荷下，电机温度预测均方误差结果

载荷	训练集 MSE	测试集 MSE
150	0.014	0.670
192	0.004	0.034
222	0.042	1.925

表 4.4 基于电气信号，150、192、222 三组不同载荷下，PU 温度预测均方误差结果

载荷	训练集 MSE	测试集 MSE
150	0.001	0.031
192	0.004	0.032
222	0.026	0.610

## 第5章 利用重要性采样对训练过程加速

### 5.1 引言

这个章节主要讲了将重要性采用的方法应用于4中设计的模型的训练过程的加速。参考相关的工作，搭建了一个，基于最新版本的深度学习框架的，对于各类神经网络通用的训练加速的框架，介绍了框架的具体实现的方式，及其加速的效果。

### 5.2 训练加速算法

需要指出的是，不同的样本在不同的研究阶段对模型的训练效果是不一样的。基于训练加速这一研究目标，这里从样本的角度出发，研究如何尽可能高效地利用样本，在每个训练阶段选择对当前训练最有效果的样本来进行训练。那么需要定义怎样算对训练有效果，以及怎么选择合适的选择样本的策略。

---

#### 算法 5.1 重要性采样算法

---

```

输入:  $B$ 、 $b$ 、 $\tau_{th}$ 
 $t \leftarrow 1$ 
 $\tau \leftarrow 0$ 
repeat
  if  $\tau > \tau_{th}$  then
     $P \leftarrow$  从标准分布中采样  $B$  个点
    计算  $P$  中所有点的分数  $S$ 
    重采样:  $Q \leftarrow$  从这  $B$  个点中抽取  $b$  个点
    权重:  $w_i \leftarrow \frac{1}{B \cdot S_i}, \forall i \in P$ 
     $\theta_t \leftarrow \text{sgd\_step}(w_i, Q, \theta_{t-1})$ 
  else
     $Q \leftarrow$  从标准分布中采样  $b$  个点
    权重:  $w_i \leftarrow 1, \forall i \in Q$ 
     $\theta_t \leftarrow \text{sgd\_step}(w_i, Q, \theta_{t-1})$ 
  end if
  update  $\tau$ 
until convergence

```

---

### 5.3 基于重要性采样的训练加速方法框架的设计

本文中提到的训练加速框架是基于新版本 Tensorflow 2.4 的基础上完成，应用了最新的 Tensorflow 的 Keras 接口。新版本 Tensorflow 中集成了 Tensorflow 的灵活性，和 Keras 接口的高度抽象简单易用的特点。以往的框架往往是基于旧版本的 Tensorflow 或 Keras 编写而成，对于新版本的模型没办法调用框架来进行加速。相对于以往的框架而言，本系统结合了动态图和静态图的优点，模块逻辑清晰而且运行效率高。

而且框架适用于各类神经网络，对具体神经网络的结构没有限制和要求。

#### 5.3.1 模型结构设计

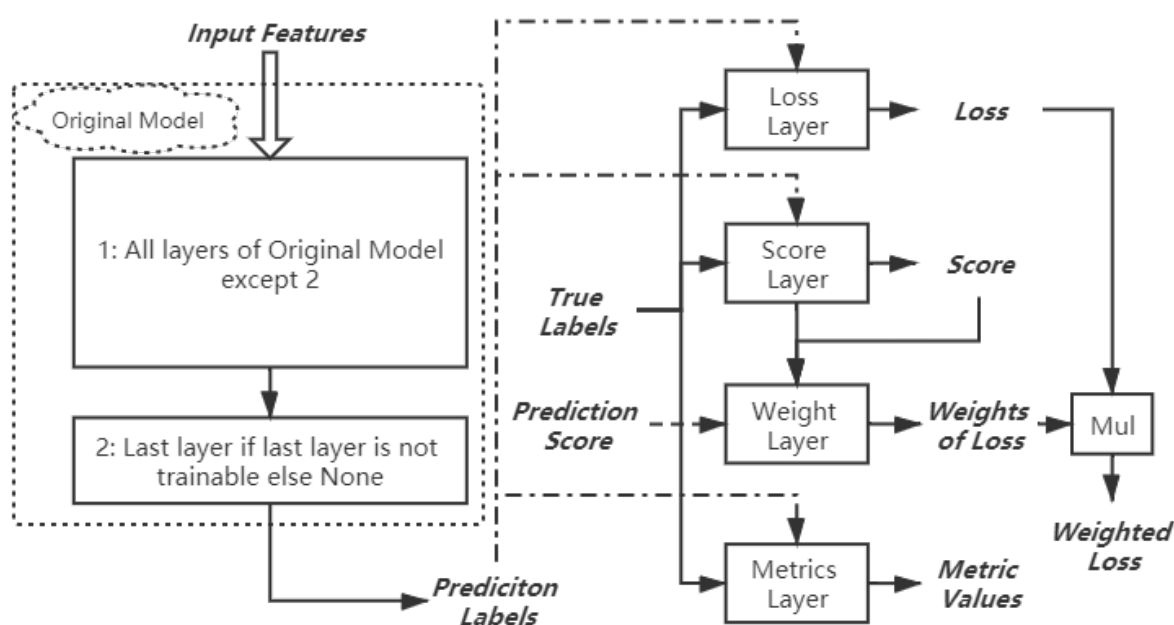


图 5.1 重要性采样模型结构

如图5.1画出了应用重要性采样的模型构建。整体模型架构是在原有的模型的基础上搭建完成的，在原来的模型的层后面又加入了额外的层，这些层用到了原来模型的输出作为输入，同时引入了一些其他的数据如标签数据作为输入，得到额外的输出。

主要的附加层是这四种，下面对它们进行分别的介绍。

1. Loss layer: 得到的样本的损失函数值。
2. Score layer: 得到的结果用来衡量样本的重要性，用于筛选样本，及决定样本的梯度下降的权重。有几种可以选择的类型：“loss”、“gnorm”、“full\_gnorm”、

”acc”

3. Weight layer: 会根据每个样本的分数计算得到样本的损失权重。
4. Metrics Layer: 计算得到评价指标。

## 5.4 训练加速实验效果

在下面的实验中，应用前面提到的重要性采样的方法，对模型进行加速，并和原始模型的结果进行对比。为了消除随机性的影响，同时验证方法的鲁棒性，每组实验进行三次，会给出平均的结果，和三组分别的结果，进行比对。为了保证公平，同一次实验中，对比的两种训练方式会加载同样的模型初始化参数。

### 加速效果

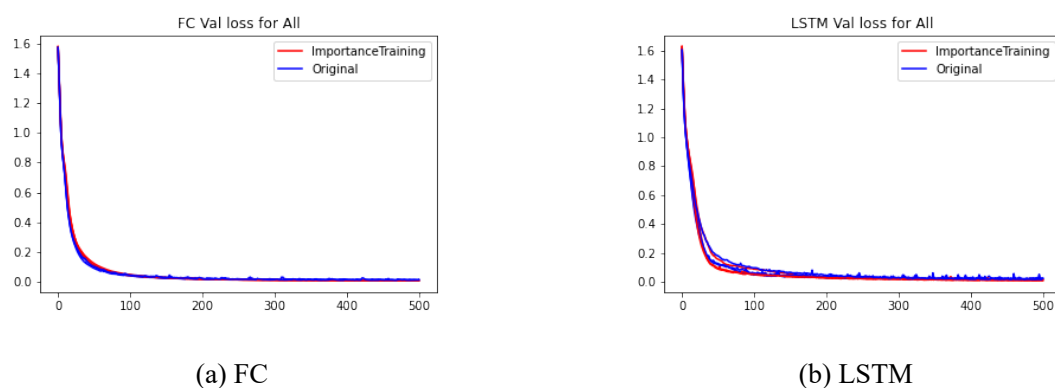


图 5.2 三次对比实验训练加速平均效果：验证集的 loss 变化趋势图

表 5.1 重要性采样的效果：数据随机划分下，全连接网络测试集上的 MSE 结果

结果类型	重要性采样方法	对比实验	MSE 比例 (<1 为有效果)
三次实验平均	0.053	0.069	0.759
实验 1	0.055	0.084	0.651
实验 2	0.054	0.057	0.948
实验 3	0.049	0.067	0.733

图5.2中给出了，应用不同模型，在应用和不应用重要性采样的方法下，多次实验，验证集 loss 的变化平均趋势对比，直接这样看看不出加速效果。

从5.2(a)和5.2(b)中加速前后的对比，我们可以看出，应用方法的初期，采样其



表 5.2 重要性采样的效果：数据随机划分下，长短时记忆网络测试集上的 MSE 结果

结果类型	重要性采样方法	对比实验	MSE 比例 (<1 为有效果)
三次实验平均	0.096	0.147	0.652
实验 1	0.097	0.142	0.681
实验 2	0.109	0.124	0.882
实验 3	0.082	0.175	0.467

实比较接近标准的采样方式，其实不会有太大的加速效果。

但是从5.1和5.2中的结果来看，训练同样的 epochs，明显应用了重要性采样方法进行训练加速的最终结果要好很多，最终的 MSE 的值的比例在加速后：加速前=0.6~0.8 左右。加速效果比较明显。

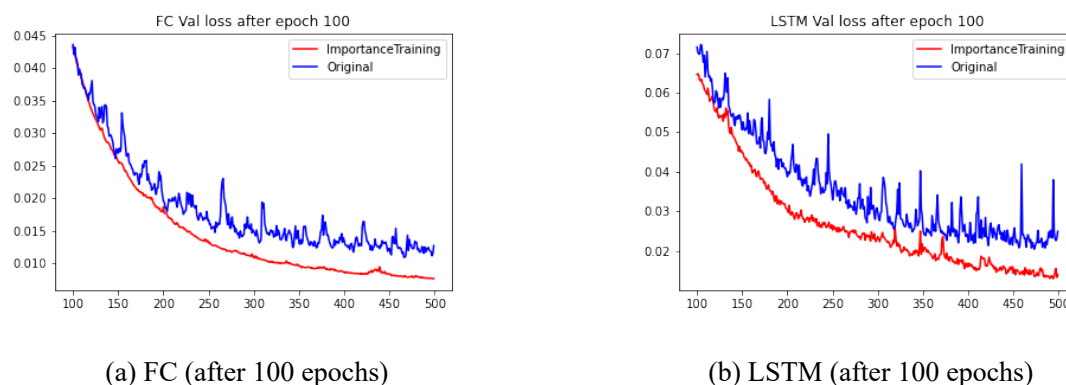


图 5.3 三次对比实验训练加速平均效果：验证集的 loss 变化趋势图（100 个 epoch 开始）

### 加速效果的稳定性

需要注意的是，由于前 100 个 epoch 的对比不是很明显，5.3中的图5.3(a)和图5.3(b)对 100 个 epoch 以后的数据单独 plot 了出来。明显应用了重要性采样加速的验证集 loss 下降更快。

本文中应用的重要性采样的方法会有比较稳定的加速效果，如图5.4中所示，重要性采样方法处理后的模型，相比原来原来的模型，应用不同的模型分别进行的三次实验中，loss 下降的速度都要更快。多次实验结果比较一致，反映了加速效果会比较稳定，说明整个加速框架比较鲁棒，不太会受到随机因素的影响。

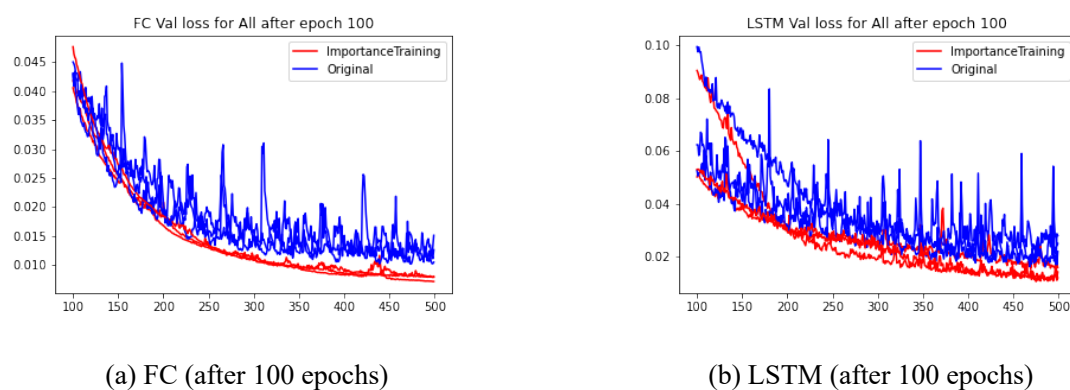


图 5.4 三次对比实验分别的训练加速效果：验证集的 loss 变化趋势图

## 5.5 本章小结

本章主要介绍了应用重要性采样方法对模型进行训练加速的相关内容。从实验的结果可以看出，整体的加速效果较好，而且加速效果比较稳定，说明加速系统比较鲁棒。

## 第 6 章 总结与展望

### 6.1 本文总结

### 6.2 未来展望

## 参考文献

- [1] Siegel E. Predictive analytics: The power to predict who will click, buy, lie, or die: volume 10[M]. Wiley Hoboken, 2013.
- [2] 国际自动化协会报告 [EB/OL]. <https://www.isa.org>.
- [3] Lebedev V, Lempitsky V. Speeding-up convolutional neural networks: A survey[J]. Bulletin of the Polish Academy of Sciences. Technical Sciences, 2018, 66(6).
- [4] Alexandrov A, Benidis K, Bohlke-Schneider M, et al. Gluonts: Probabilistic time series models in python[J/OL]. CoRR, 2019, abs/1906.05264. <http://arxiv.org/abs/1906.05264>.
- [5] Makridakis S, Spiliotis E, Assimakopoulos V. The m4 competition: 100,000 time series and 61 forecasting methods[J/OL]. International Journal of Forecasting, 2020, 36(1): 54-74. <https://www.sciencedirect.com/science/article/pii/S0169207019301128>. DOI: <https://doi.org/10.1016/j.ijforecast.2019.04.014>.
- [6] Salinas D, Flunkert V, Gasthaus J, et al. Deepar: Probabilistic forecasting with autoregressive recurrent networks[J]. International Journal of Forecasting, 2020, 36(3): 1181-1191.
- [7] Smyl S. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting[J]. International Journal of Forecasting, 2020, 36(1): 75-85.
- [8] Hewamalage H, Bergmeir C, Bandara K. Recurrent neural networks for time series forecasting: Current status and future directions[J]. International Journal of Forecasting, 2021, 37(1): 388-427.
- [9] De Brouwer E, Simm J, Arany A, et al. Gru-ode-bayes: Continuous modeling of sporadically-observed time series[J]. arXiv preprint arXiv:1905.12374, 2019.
- [10] Dabrowski J J, Zhang Y, Rahman A. Forecastnet: A time-variant deep feed-forward neural network architecture for multi-step-ahead time-series forecasting[C]// International Conference on Neural Information Processing. Springer, 2020: 579-591.
- [11] LeCun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural computation, 1989, 1(4): 541-551.
- [12] Amari S, et al. The handbook of brain theory and neural networks[M]. MIT press, 2003.
- [13] Chung J, Gulcehre C, Cho K, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. arXiv preprint arXiv:1412.3555, 2014.
- [14] Binkowski M, Marti G, Donnat P. Autoregressive convolutional neural networks for asynchronous time series[C]// International Conference on Machine Learning. PMLR, 2018: 580-589.
- [15] Shi X, Chen Z, Wang H, et al. Convolutional lstm network: A machine learning approach for precipitation nowcasting[J]. arXiv preprint arXiv:1506.04214, 2015.
- [16] Bai S, Kolter J Z, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling[J]. arXiv preprint arXiv:1803.01271, 2018.

- [17] Pascanu R, Gulcehre C, Cho K, et al. How to construct deep recurrent neural networks[J]. arXiv preprint arXiv:1312.6026, 2013.
- [18] Bordes A, Ertekin S, Weston J, et al. Fast kernel classifiers with online and active learning.[J]. Journal of Machine Learning Research, 2005, 6(9).
- [19] Zhao P, Zhang T. Stochastic optimization with importance sampling for regularized loss minimization[C]// international conference on machine learning. PMLR, 2015: 1-9.
- [20] Wu C Y, Manmatha R, Smola A J, et al. Sampling matters in deep embedding learning[C]// Proceedings of the IEEE International Conference on Computer Vision. 2017: 2840-2848.
- [21] Needell D, Ward R, Srebro N. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm[J]. Advances in neural information processing systems, 2014, 27: 1017-1025.
- [22] Fan Y, Tian F, Qin T, et al. Neural data filter for bootstrapping stochastic gradient descent[J]. 2016.

## 附录 A 补充内容

附录是与论文内容密切相关、但编入正文又影响整篇论文编排的条理和逻辑性的资料，例如某些重要的数据表格、计算程序、统计表等，是论文主体的补充内容，可根据需要设置。

### A.1 电网数据补充说明

基于青岛永磁实验台，按线路条件（包括牵引、匀速、制动），跑典型工况，进行相关数据采集。牵引主电路图如下：!!! 参考数据说明文档

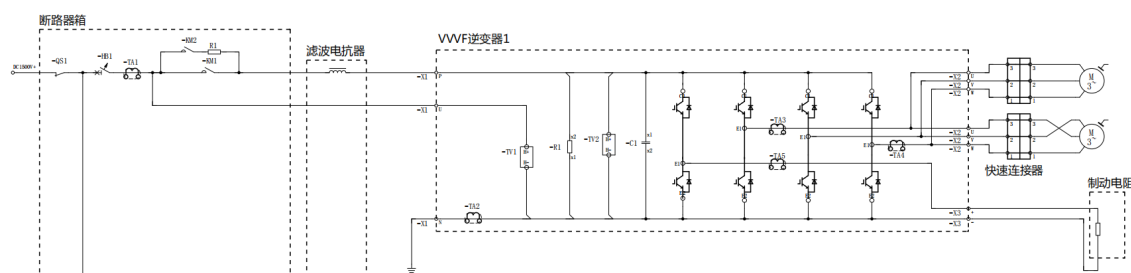


图 A.1 牵引主电路图

## 致 谢

衷心感谢导师邓仰东副教授对本人的精心指导。他的言传身教将使我终生受益。

感谢实验室全体老师和同窗们学的热情帮助和支持！

本课题基于科技部国家重点研发计划，对于科技部提供的帮助，特此致谢。

## 声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名：\_\_\_\_\_ 日 期：\_\_\_\_\_



## 个人简历、在学期间完成的相关学术成果

### 个人简历

1997 年 7 月 3 日出生于湖南常德市澧县。

2014 年 9 月考入清华大学自动化系自动化专业，2018 年 7 月本科毕业并获得理学学士学位。

2018 年 9 月免试进入清华大学软件工程系攻读软件工程硕士至今。

### 在学期间完成的相关学术成果

专利：

[1]

## 指导小组学术评语

论文提出了……

## 答辩委员会决议书

论文提出了……

论文取得的主要创新性成果包括：

1. ……

2. ……

3. ……

论文工作表明作者在 ××××× 具有 ××××× 知识，具有 ×××× 能力，论文 ××××，  
答辩 ××××。

答辩委员会表决，（× 票/一致）同意通过论文答辩，并建议授予 ×××（姓名）  
×××（门类）学博士/硕士学位。