高频交易下的股票涨跌预测

梅智敏

院 （系）： 计算学部 专 业：软件工程

学　　号： 1183710118 指导教师：张彦航

**2022年5月**



**毕业设计（论文）**

题 目 高频交易下的股票涨跌预测

专 业 软件工程

学 号 1183710118

学 生 梅智敏

指 导 教 师 张彦航

答 辩 日 期

# 摘 要

在数字经济时代，随着数据的规模越来越大，如何压缩和存储是一个基础且重要的课题。然而，传统的数据压缩算法只将重点放在如何提升压缩率，使之接近香农提出的熵的下界，这些算法不适用于全部的场景，特别是对于压缩后的数据有查询需求（如在避免解压全部数据的情况下访问部分数据）时。因此需要一种算法来适应存在着大量对压缩数据有实时查询需求的应用的系统。且随着物联网技术的不断发展、数据的不断增加，基于云的物联网解决方案渐渐无法满足企业的需求。此时边缘计算作为云的扩展，可以加快数据分析的速度，便于企业更快更好地做出决策，但边缘硬件也对算法提出了新的要求，如存储空间更小、不适合做批处理等。因此，本文在边缘硬件环境下实现了基于有损压缩、可对压缩数据进行分析式查询的Plato算法，并通过利用误差向量的正交投影性质，进一步改善了向量乘法运算的error-bound；实现了基于无损压缩、支持随机访问的GD算法，并通过改进预处理时的终止条件，避免了预估空间落入局部最优值。

关键词：时序数据；数据压缩；边缘硬件；误差度量；随机访问

# Abstract

In the era of digital economy, with the increasing scale of data. How to compress and store is a basic and important topic. However, the traditional data compression algorithms only focus on how to improve the compression rate and make is close to the lower bound of Shannon’s entropy.These algorithms are not suitable for all scenarios especially when the compressed data needs to be queried(such as accessing part of the data without decompressing all the data). Therefore, an algorithm is needed to adapt to the system with a large number of real-time query requirements for compressed data. With the continuous development of Internet of things technology and the increasing of data, cloud based Internet of things solutions are gradually unable to meet the needs of enterprises. At this time, as an extension of the cloud, edge computing can speed up the speed of data analysis, which is convenient for enterprises to make decisions faster and better. But edge hardware also puts forward new requirements for the algorithm, such as smaller storage space, not suitable for batch processing and so on. Therefore, in the edge hardware environment, this paper implements the Plato algorithm based on lossy compression, which can query compressed data analytically, and further improves the error bound of vector multiplication by using the orthogonal projection property of error vector; The GD algorithm based on lossless compression and random access is implemented. By improving the termination condition of preprocessing, the prediction space is avoided to fall into the local optimal value.

**Keywords:** time series, data compression, edge hardware, error measurement, random access

# 目 录

[摘 要 I](#_Toc104145642)

[Abstract II](#_Toc104145643)

[目 录 III](#_Toc104145644)

[第1章 绪 论 - 1 -](#_Toc104145645)

[1.1 课题背景及研究的目的和意义 - 1 -](#_Toc104145646)

[1.1.1 高频交易 - 1 -](#_Toc104145647)

[1.1.2 限价订单簿Limit Order Book - 1 -](#_Toc104145648)

[1.1.3 研究目的和意义 - 5 -](#_Toc104145649)

[1.2 国内外研究现状 - 5 -](#_Toc104145650)

[1.2.1 概述 - 5 -](#_Toc104145651)

[1.2.2 基于统计模型的方法介绍 - 6 -](#_Toc104145652)

[1.2.2 基于深度学习模型的方法介绍 - 7 -](#_Toc104145653)

[1.3 该领域存在的挑战及本文的主要研究内容 - 8 -](#_Toc104145654)

[1.3.1 该领域存在的挑战 - 9 -](#_Toc104145655)

[1.3.2 主要研究内容 - 9 -](#_Toc104145656)

[1.4 章节安排 - 10 -](#_Toc104145657)

[第2章 问题形式化及数据集描述 - 12 -](#_Toc104145658)

[2.1 研究问题的形式化 - 12 -](#_Toc104145659)

[2.1.1 MidPrice涨跌预测形式化 - 12 -](#_Toc104145660)

[2.1.2 交易策略形式化 - 13 -](#_Toc104145661)

[2.2 数据集描述 - 13 -](#_Toc104145662)

[2.2.1 时序数据分段 - 13 -](#_Toc104145663)

[2.2.2 估计函数选择与存储 - 14 -](#_Toc104145664)

[2.2.3 误差度量 - 15 -](#_Toc104145665)

[2.3 Plato数据查询 - 15 -](#_Toc104145666)

[2.3.1 误差传播 - 16 -](#_Toc104145667)

[2.3.2 时序数据操作的误差界分析 - 18 -](#_Toc104145668)

[2.3.3 使用正交投影性质改进误差界 - 19 -](#_Toc104145669)

[2.3.4 段不对齐的情况下改进误差界 - 20 -](#_Toc104145670)

[2.3.5 对误差界的进一步改进 - 22 -](#_Toc104145671)

[2.4 本章小结 - 23 -](#_Toc104145672)

[第3章 GD算法的实现与改进 - 24 -](#_Toc104145673)

[3.1 引言 - 24 -](#_Toc104145674)

[3.2 GD数据压缩 - 24 -](#_Toc104145675)

[3.2.1 预处理 - 25 -](#_Toc104145676)

[3.2.2 压缩 - 26 -](#_Toc104145677)

[3.2.3 压缩文件大小分析 - 28 -](#_Toc104145678)

[3.3 GD数据解压缩与随机访问 - 28 -](#_Toc104145679)

[3.3.1 GD数据解压缩 - 29 -](#_Toc104145680)

[3.3.2 GD随机访问及代价分析 - 30 -](#_Toc104145681)

[3.4 本章小结 - 32 -](#_Toc104145682)

[第4章 实验设计与分析 - 33 -](#_Toc104145683)

[4.1 实验环境 - 33 -](#_Toc104145684)

[4.2 Plato算法分析 - 34 -](#_Toc104145685)

[4.2.1 数据集及预处理 - 34 -](#_Toc104145686)

[4.2.2 实验分析 - 34 -](#_Toc104145687)

[4.3 GD算法分析 - 38 -](#_Toc104145688)

[4.3.1 数据集及预处理 - 38 -](#_Toc104145689)

[4.3.2 实验分析 - 38 -](#_Toc104145690)

[4.4 本章小结 - 41 -](#_Toc104145691)

[结 论 - 43 -](#_Toc104145692)

[参考文献 - 44 -](#_Toc104145693)

[哈尔滨工业大学本科毕业设计（论文）原创性声明 - 46 -](#_Toc104145694)

[致 谢 - 47 -](#_Toc104145695)

# 第1章 绪 论

## 1.1 课题背景及研究的目的和意义

### 1.1.1 高频交易

在过去的几十年里，新技术正在或者已经改变了股票以及其他金融产品的交易活动。在计算机出现之前，所有的股票交易都是在交易大厅里面人与人之间进行，后台其他人员的处理活动确保了交易的顺利完成。但是，后台处理和实际交易过程中已经被逐渐地自动化。而高频交易就是自动化交易的一个子集，它通常被定义为利用计算机交易算法来提交交易指令和取消交易指令。根据美国证券交易委员会的描述，高频交易具有以下的一些特征：（1）使用高性能的计算机以及复杂的计算机应用程序来产生，路由和执行交易指令。根据最新的行情信息和宏观消息计算产生交易信号是高频交易最核心的功能之一。此外，另有监控系统对交易策略进行实时监控，当异常事件发生时便实时发出警告。（2）将服务器直接架设在交易所，最小化由网络等原因造成的延时在极短的时间内创建和实施买卖交易。（3）采用逐笔数据进行计算。高频交易往往需要以毫秒级的逐笔数据作为计算基础，以消化最新的市场信息，并以此制定交易决策。与人工交易比起来，程序化交易能够更快、有效、理性地决策，这就使得程序化成为高频交易不可或缺的一部分。（4）买卖频繁，持仓时间短。

### 1.1.2 限价订单簿Limit Order Book

传统的金融市场是以报价驱动为基础而运作的，即交易员与交易商（也称“做市商”）在市场中进行沟通。交易商负责维护金融资产的库存量，公布出价和要价，并被要求按其报价进行交易。交易商的存在提供了市场流动性，但是报价驱动市场的透明度相对较低。

而如今，随着电子通信网络和高频交易的快速发展，订单驱动的金融市场成为了主流。它和传统的报价驱动市场最大的区别就在于市场中的交易员可以直接与其他交易员进行交易而没有中间交易商的干预。而订单驱动的金融市场的核心就是以限价订单簿Limit Order Book为中心的双重拍卖机制来促进交易，这一机制已经被全球各大交易所所广泛采用，例如纽约证券交易所（NYSE）、纳斯达克和伦敦证券交易所（LSE）。

在以Limit Order Book为中心的金融市场中，交易者可以提交两大类交易指令：限价指令Limit Order和市场价指令Market Order。前者会要求该交易的成交价格必须好于或者等于所指定的值，如果当前市场中最好的价格仍然不能满足要求，这个交易指令就会继续等待直到市场中出现了满足要求的价格为止，所以此类指令具有“执行不确定性”，即他有可能不会被执行，一直在等待状态。后者则要求该交易立刻以当前市场中最好的价格来完成订单，此类订单一定会被执行，但是不能保证成交价格足够好，所以它具有“价格不确定性”。另外，除了提交这两类指令，交易者也可以取消他们之前所提交但是暂时处于等待状态的Limit Order。

本质上来说，Limit Order Book（也称为 LOB）是当前金融市场行情的一个快照，其基本功能是匹配市场中的买卖双方，它由股票交易所统一管理并呈现给每一位交易者，以帮助交易者们做出交易决策。LOB包含了一组可以代表市场状态及其变化方式的数据且会动态更新，它是当今大多数电子交易所在金融和加密货币市场中使用的机制。图1是LOB的示意图，可以帮助我们直观了解。

**独立统计：**每一支股票的LOB数据分开统计，每一个LOB只反映其对应的那支股票的信息。

**买卖两方**：Ask代表卖方，Bid代表买方。

**价格数量对应关系**：它表明当前市场中有人愿意以多少价格出售/买入多少数量的股票，以Ask方的20.6价格为例，它对应的Volume值为2，代表目前市场中有人愿意以“不低于20.6”的价格出售2支股票。需要注意的是，展示在Limit Order Book中的价格都是Limit Order中所要求的价格，而Limit Order是给出某个价格底线，然后要求成交价“好于或者等于该底线价格”。故Bid方20.5价格对应Volume值为3的含义就是：当前市场中有人愿意以“不高于20.5”的价格买入3支股票。

**买卖双方不同的Level：**仅仅依据交易者的预期价格而无需考虑对应的Volume来对他们提交的“交易需求”进行分级，Level-1是最高级，当市场中的交易发生时，会优先匹配高等级的“交易需求”再到低等级的需求。首先对于Bid方，价格越高则所处的等级越高，故20.5的价格处于Level-1，20.4的价格处于Level-2。接着是Ask方，价格越低则所处的等级越高，20.6的价格处于Level1，20.7的价格处于Level2。

**LOB的更新：**每当市场中有交易员提交了新交易指令或者取消了之前的某个Limit Order，LOB都会更新以呈现出当前市场的真实交易情况给每一位交易者。图2和图3详细展示了两种情况下的LOB更新过程：（1）当有交易者提交了新的交易指令时，LOB会发生更新，图2中展示的是限价为20.55的Limit Ask Order取代20.6成为了新的Best Ask。注意：若有交易者提交新的Market Order时，则会立刻与LOB中记录的Limit Order发生匹配，也会导致LOB的更新。（2）当有交易者取消了尚未执行Limit Order也会引起LOB的更新，图3中展示的是限价为20.6的Limit Ask Order被它的提交者给取消了，所以LOB就会删除这一指令的记录，导致20.7从Level-2 Ask变成了Level-1 Ask（Best Ask）。

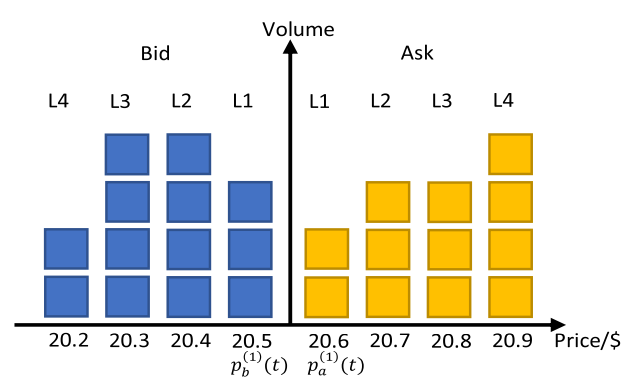


图1 LOB数据示意图

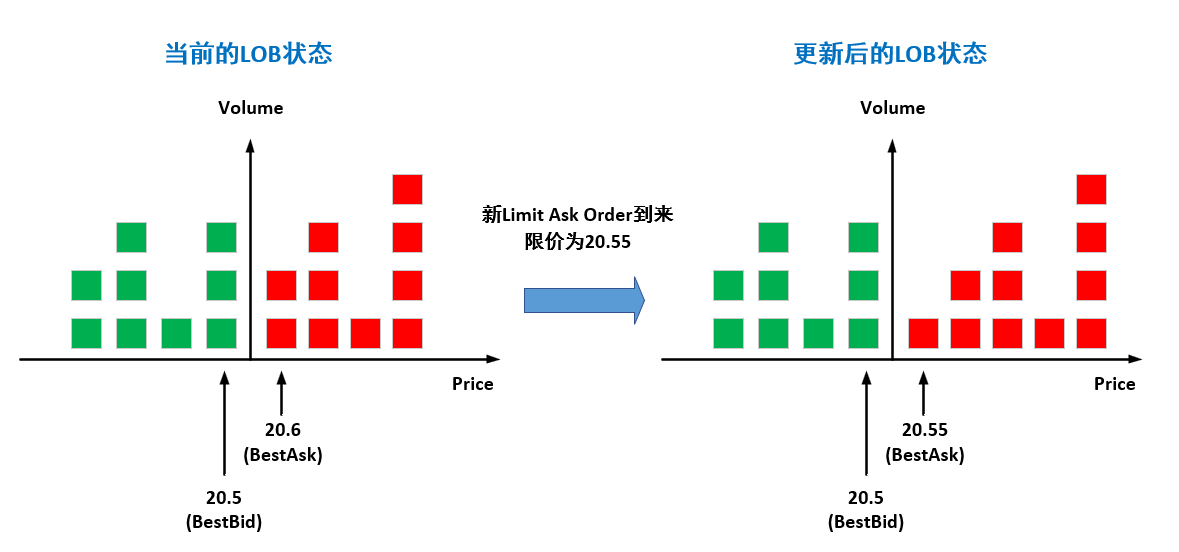


图2 新Limit Order到来导致LOB更新

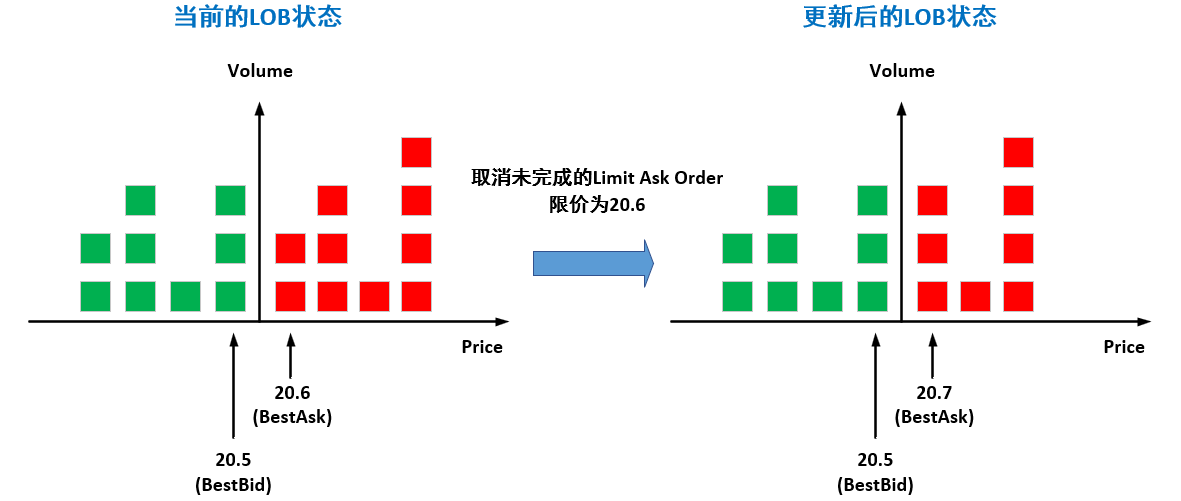


图3 取消未完成的Limit Order导致LOB更新

### 1.1.3 研究目的和意义

在基于LOB匹配机制的金融市场中，**Best Ask**是市场中交易者愿意出售股票的最低价格。也就是说，你可以买到该股票的最低价格。同理，**Best Bid**就是市场中交易者愿意卖出该股票的最高价格。在此引入一个新的指标**Mid Price：**





图4 LOB中Mid-Price的意义

在高频交易市场中，Mid Price是一项重要的指标，它可以表示资产的一般市场价值，所以人们经常使用这一指标来代表资产价值，例如图4中苹果AAPL股票价格其实就是对应LOB中的Mid Price。

因此，对于量化金融研究者而言，如何准确预测短期内LOB中Mid Price的涨跌情况成为了近年来的热点问题。目前的高频交易研究主要包含两大类：价格预测和优化策略，而价格预测也正是优化策略的基础。因此借助LOB进行短期Mid Price变化预测可以有效指导交易者们合理优化自己的交易策略以获得最大收益，具有重要的实际意义。

## 1.2 国内外研究现状

### 1.2.1 概述

已有的方法主要可以分为两大类：统计模型方法以及数据驱动的机器学习方法。其中前者是传统的主流方法，优点是不需要利用大量的数据来训练，复现较为简单；缺点是这类方法经常会对数据的分布进行一些和实际情况不相符合的假设，从而导致应用到实际的高随机性的股票数据时，预测效果往往不好。例如[1]中所提出的VAR模型以及[2]中提出的ARIMA模型。而今年来随着深度学习的兴起，研究者们更倾向于使用数据驱动的深度学习方法来完成高频交易下短期股票涨跌趋势预测任务，本课题也挑选了部分已有的方法进行了复现。

### 1.2.2 基于统计模型的方法介绍

试图利用统计模型或者随机模型来捕获Limit Order Book中的动态特征，并在此基础上准确预测短期内股票价格涨跌趋势是很多统计学派研究者们所希望做到的。

例如，Gourieroux（1999年）[3]、Bouchaud（2002年）[4]和Smith（2003年）[5]通过套用多种统计模型进行试验挖掘出了Limit Order Book中20多种动态特征，并给出广泛统计特征列表。然而，他们的工作仍然不能将特征统一到单个统计模型中加以使用。可喜的是，Cont和de Larrard在2011年的工作[6]中，填补了上述空白，他们提出使用马尔可夫模型来模拟以Limit Order Book为基础的金融市场运作过程，该模型捕捉了Market Order以及Limit Order的主要特征并在此基础上模拟这些特征对价格走势的影响。但是他们的模型仅仅考虑了LOB中Level-1的特征，即大部分剩余特征并不在考虑范围内。值得一提的是，相同的作者Cont在他的另一个工作[7]中确实考虑到了多个Level的Price以及对应的Volume信息，他假定所有即将到来的市场事件（Limit Order，Market Order以及Cancel Order）都遵循独立的泊松过程，以此来预测未来的MidPrice。受到Cont和de Larrard使用马尔可夫模型成功的启发，越来越多的学者开始研究具有随机驱动项的类马尔可夫模型，如向量自回归模型（VAR）[1]和自回归综合移动平均模型（ARIMA）[2]。但是这些模型为了避免过多的参数空间，通常依赖于人工创建的数据特征，但是以此类特征为基础的统计模型大多不具备泛用性，它们往往在某几支股票上表现很好但是在其他股票上表现很差。

总的来说，在2015年之前，使用传统的统计模型来理解Limit Order Book动态过程的研究工作还较多，近年来则是大大减少了。其根本原因就在于这类方法要么大量使用一些人工创建的特征，要么对数据或者市场中事件的分布进行一些和实际情况不相符合的假设，从而导致模型难以应用到实际的高随机性的股票数据。

### 1.2.2 基于深度学习模型的方法介绍

近年来随着深度学习的快速发展以及传统统计模型难以有效捕获LOB的动态特征，越来越多的深度学习模型被应用到金融数据中以尝试利用Limit Order Book中的多维市场交易信息来预测短期内金融资产（股票、证券以及外汇等）的价格走势。其中MLP和SVM是两类常用的分类模型，例如在[8]中，作者训练了一个SVM模型来预测未来30s内日经225指数的价格涨跌走势。在[9]中，作者使用12个维度的特征向量作为输入，来预测韩国综合股价指数的日内价格走势。并且在两个不同大小的预测窗口下（一个为长期，另一个为短期）对MLP和SVM的性能进行了比较。在[10]中，作者通过一个预测期货未来价值的任务来对MLP，SVM和RBF-NNs进行了实验比较。

除了在不同的金融数据集中比较分类器的性能外，也有很多学者将研究重心放在如何从Limit Order Book中提取合适的特征来帮助后续预测任务。由于金融数据的高频高随机特性，没有预处理或者特征提取的话会给后续模型训练带来极大困难。为此，[11]中的作者在模型训练之前使用主成分分析PCA来对源数据进行降维从而使得相同的模型性能得到了提升。但是，这种将特征提取方法是静态的预处理步骤，没有随着模型训练过程实时优化以最大限度地提升模型的整体性能。为了弥补这一缺陷，一些学者提出将特征提取功能设计成模型中的某个单独的网络层，这样在模型训练过程中就可以实时利用反向传播来更新自身参数以达到优化的目的。相关的代表工作有[12]中的BoF模型以及[13]中的CNN模型，在[12]中，作者在公开数据集FI-2010上进行了实验，这是基于Limit Order Book进行股票价格预测的标准benchmark数据集，包含连续 10 天从纳斯达克北欧股票市场中提取的五只股票的LOB的标准化数据表示，约4,000,000 个样本点。在[13]中，作者使用卷积神经网络 (CNN) 来进行特征提取，创新性地将该方法应用到高频股票数据中。该方法使用来自金融交易所的非公开大规模、高频LOB数据作为输入来预测股票的价格走势，并且在实验中与其他不做特征提取的方法（MLP）进行比较。这类工作表明了从大量数据中提取代表性特征对于理解Limit Order Book的动态过程具有重要价值。

还有一类模型近年来也在金融数据领域备受关注，那便是LSTM[14]。该模型最初是为了解决循环神经网络RNN对于长时依赖的消失梯度问题[15]，现在已经广泛应用于NLP领域，也常常在seq2seq [16]模型架构中充当encoder或者decoder的编码器。近年来越来越多的研究者使用LSTM来抓取股票历史数据之间的“时序依赖信息”从而帮助预测MidPrice走势。代表工作有[17]，作者们使用1000只股票的Limit Order Book数据来测试他们设计的四层LSTM模型。结果显示，随着时间的推移，样本外预测精度并未明显下降，这体现了在金融数据中LSTM具有不错的潜力。

总的来说，近年来深度学习的研究愈发火热，基于DL模型来理解LOB动态过程的研究大量涌现。其主要围绕三大主题：如何从LOB中提取合适的特征，如何从时间维度捕获LOB数据之间的依赖关系以及如何让模型在不同大小的预测范围下仍保持良好的表现。

## 1.3 该领域存在的挑战及本文的主要研究内容

### 1.3.1 该领域存在的挑战

综合考虑前面的领域内研究调查，我将该领域内的挑战总结为三点：

(1) 对于统计模型，难以在足够小的参数空间和合理的分布假设之间做到平衡，往往是顾此失彼。

(2) 对于深度学习模型，如何从LOB中提取合适的特征，如何从时间维度捕获LOB数据之间的依赖关系以及如何让模型在不同大小的预测范围下仍保持良好的表现都是当前面临的难点。

(3) 需要充分的实验来佐证预测模型的优良性能，主要包括两方面的测试：数据集上的模型指标测试以及回测收益测试。前者是利用一些评价指标如Accuracy、Precision、Recall以及F1值来比较不同模型在同一个数据集上的表现好坏；后者则是依据预测模型的结果设计一个交易策略，然后在历史数据上模拟执行此交易策略，根据最终得到的收益或者损失Profit&Loss（PnL）来从另一个角度衡量不同策略的好坏。

### 1.3.2 主要研究内容

本次毕设的主要研究内容为在高频交易的应用场景下借助Limit Order Book数据来对股票的涨跌走势进行预测，同时根据预测结果制定可以获利的交易策略。研究主要分为以下几个阶段：

(1) 首先需要广泛阅读文献，了解目前的研究者们主要采用统计模型还是深度学习模型、提取了Limit Order Book中的哪些特征、设计模型的依据、一般在哪些数据集上进行测试。

(2) 其次，需要从已有文献中获得启发，在贴合应用场景的基础上设计出新的分类模型用于预测短期内MidPrice的涨跌走势，模型尽量不要对数据的分布做出不合理的假设。

(3) 另外，从两个大方向来优化自己的模型：模型设计与特征提取。模型设计这块可以尝试引入seq2seq架构让模型可以同时输出多个预测范围内的价格走势结果；特征提取这块可以先尝试手动设计几个特征，然后利用和标签之间的相关性检验来证明特征的有效性。

(4) 最后，需要进行充分的实验来评估设计的模型，实验分为两大块：评价指标测试和回测盈亏。每一块都包含3大阶段：baseline方法的表现、初步模型的表现以及优化后模型的表现。需要注意的是实验中需要对预测范围T取不同值进行多次验证。

## 1.4 章节安排

本论文共5章，各章内容概述如下：

第1章为绪论，主要分为三个模块。首先介绍了课题的应用场景以及现实意义，详细描述了现代以Limit Order Book作为匹配机制的金融市场运作方式，指出基于LOB的短期MidPrice涨跌趋势预测是一大热点；其次对领域内已有的研究方法撰写了文献综述，传统的基于统计模型的方法和新兴的基于深度学习模型的方法都有介绍；最后从前人的方法中得到启发，确定了本课题模型优化方向以及需要的两大类验证实验。

第2章为问题形式化及数据集描述，该章分为两节。第一节对本课题研究的问题进行形式化并统一相关符号以便于论文后续表达；第二节对本课题所使用的公开数据集FI-2010进行详细描述，包括特征集以及标签集。

第3章为初步模型的设计与实现，该章分为四节。第一节介绍深度学习模型中的CNN和LSTM，本课题的初步模型主要用到了这两大模型；第二节详细描述初步模型的设计思路与模型结构；第三节结合数据集和模型结构给出数据流向图，详细描述了从模型输入到模型输出的向量维度和形状变化情况；第四节基于初步模型设计了一个交易策略，介绍了设计思路与实现方法。

第4章为对初步模型的改进思路与实现，该章分为三节。第一节阐述从模型角度的改进，即引入seq2seq结构和注意力机制的动机与设计思路；第二节介绍从特征提取角度的改进，即引入两个新特征OI和OIR的原因、设计思路和有效性验证；第三节综合前面两节的内容对初步模型加以优化，给出实现细节和优化后的模型。

第5章为实验设计与实现，该章分为两节，每一节都包含模型指标测试和回测盈亏两部分实验，模型指标测试会比较模型的Accuracy、Precision、Recall和F1值而回测盈亏会比较基于不同模型的交易策略在历史股票数据上的盈亏情况。第一节首先介绍实验设计，然后将初步模型和文献综述中提到的一些已有深度学习方法进行比较；第二节将初步模型和改进后的模型进行了比较，具体行文结构与第一节类似。

在最后的结论中，本论文对完成的工作进行了总结：阐述了初步模型和改进后模型的设计思路中的创新点所在；对两阶段实验得到的结果进行了综合分析与评价；同时针对高频交易中的股票涨跌预测问题对未来进行展望，提出了几种可能的研究方法。

# 第2章 问题形式化及数据集描述

## 2.1 研究问题的形式化

第一章中已经对本课题所要研究的问题进行了简略描述，此处给出更为准确的形式化表达。分为两个阶段，第一阶段是预测MidPrice的涨跌趋势，第二阶段是基于涨跌预测的交易策略。

### 2.1.1 MidPrice涨跌预测形式化

根据第一章中对LOB的详细描述，我们可以将时刻的LOB状态用向量表示：



其中不同的代表LOB中不同的Level，每一个Level包含买方价格、买方价格对应的量、卖方价格、卖方价格对应的量这4个维度的信息，分别用、、、表示，示意图如下：

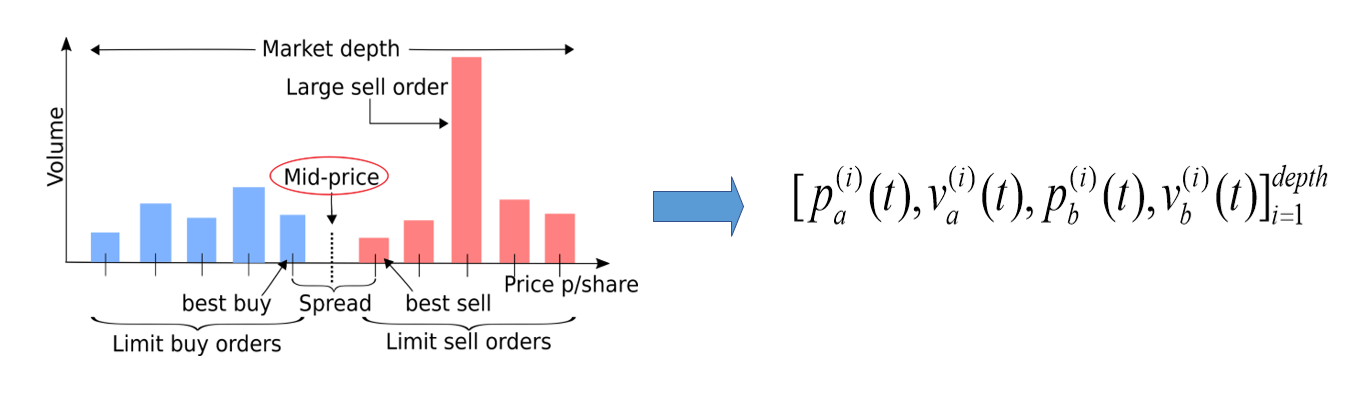


图2-1 LOB状态表示

每当市场中交易者提交新Limit Order、提交新Market Order或者取消尚未执行的Limit Order，作为市场买卖行情快照的LOB就会自动更新一次。关注每次更新的LOB状态，并用相同的向量化表示它们，就可以得到一串LOB状态序列。每一个都对应着一个MidPrice，记作：



根据第一章的描述，可以衡量该时刻的股票价值，那么我们就可以通过观察MidPrice序列的变化情况来判断股票价值的涨跌。但是由于股票数据具有很强的随机性，如果我们仅仅比较和来判断股票价值的涨跌，那么最终得到的结果将会同样随机而难以找到规律。因此，在很多已有的研究中[12][18]，人们引入了一定时间跨度内的增长率来表征股票价值的涨跌情况，具体公式如下：





其中代表未来个样本点的平均MidPrice，而代表过去个样本点以及当前样本点的平均MidPrice，接下来引入增长率的概念：



表示未来时间范围内的股票平均价值相较于过去时间范围内的股票平均价值涨幅。另外，股票价值可以分为涨、跌、保持相对平稳这三大类，于是可以自定义标签如下：



其中，是判断阈值，一般取0.001到0.002之间。在本课题中，沿用前人在[13]中的取值0.002。

前人的研究[17]发现可以使用LSTM来捕获之间的时序依赖关系从而帮助预测，我们可以沿用这一思路，但是观察对象是能提供更多信息的LOB状态序列。于是我们将这个预测问题最终形式化如下图：

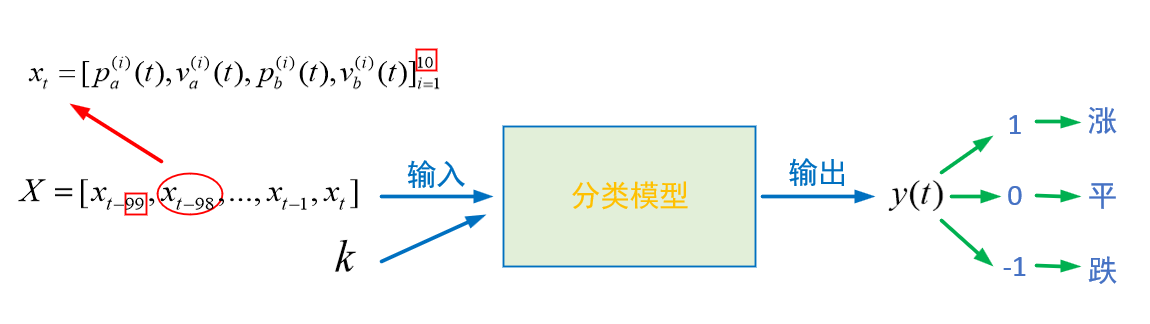


图2-2 预测问题形式化

我们观察包含当前在内的共计100个序列，为了控制输入向量的维度，每个使用前10个Level的信息，即每个包含40个维度的特征。分类模型的输入的维度是100\*40，另外需要输入预测范围，经过分类模型处理之后输出对的预测结果：涨或跌或平。

### 2.1.2 交易策略形式化

依据分类模型的预测结果可以设计出一种简易的交易策略：若意味着短期内股票价格上涨，我们需要尽可能买入；若意味着短期内股票价格下跌，我们需要尽可能卖出；若意味着短期内股票价格保持平稳，我们就不做任何操作。由此可以定义动作,可以取值，形式化表达如下图。假设预测模型在处的输出，则，我们需要向交易所提交Market Buy Order以尽快满仓买入。为了方便后续的回测盈亏实验，我们直接以BestAsk作为成交价；同时考虑到模型输出和提交指令的时间耗费，当我们的Market Buy Order到达交易所时，LOB可能已经更新了几次，所以成交价采用而不是，为延迟因子，在本课题中取。同理，若预测模型在处的输出，则，我们需要向交易所提交Market Sell Order以尽快空仓，成交价使用。

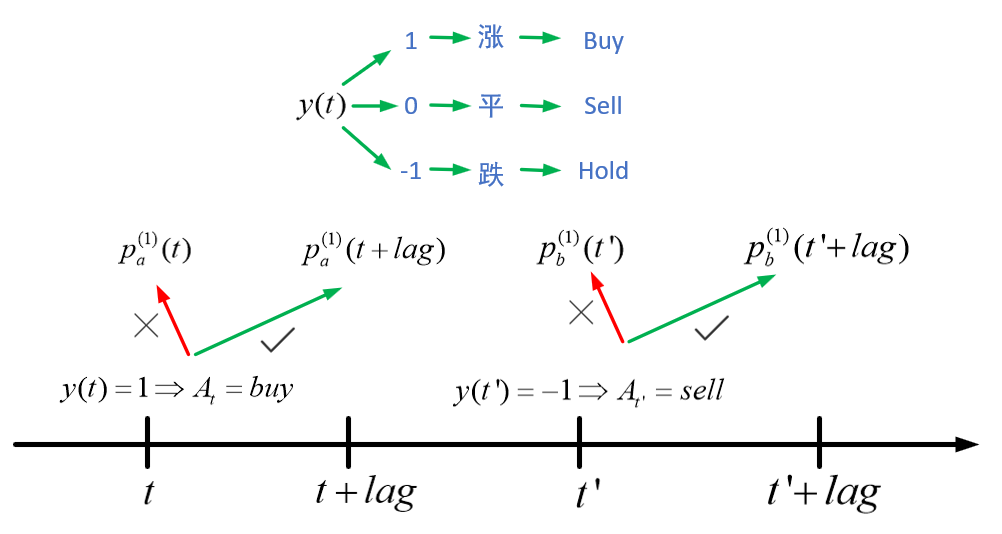


图2-3 交易策略形式化

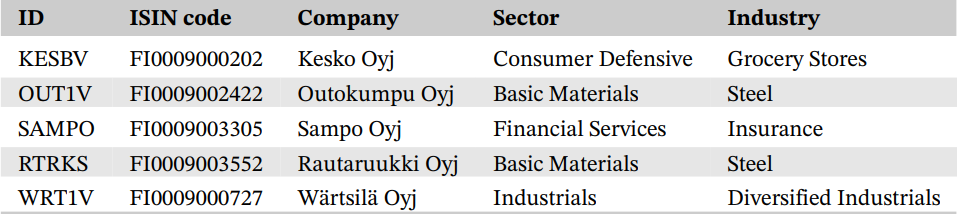
## 2.2 数据集描述

### 2.2.1 来源及包含的股票

本课题使用数据集是一个公开数据集，名为FI-2010。它来源于纳斯达克北欧金融市场，共包含五只股票在连续10个交易日内的大约400万个市场行情快照，即400万个LOB状态点，这些按照更新的时间顺序排列且五只股票分开记录。

具体来说，该数据集选用的都是五支股票在芬兰Helsinki交易所从2010年6月1号到2010年6月14号期间共10个交易日的LOB状态信息。数据集通过只选择一个股票市场交易所来避免了与分散市场相关的问题，根据一些学者的研究[19]，在分散市场的情况下，给定金融资产的Limit Order分布在多个交易所之间，这会给后续数据分析带来各种问题。在Helsinki交易所，交易时间为10:00-18:25，中午不休市。但是刚开盘的一段时间和即将收盘的一段时间内，数据波动性极大，因此，为了防止这2个特殊时间段内的数据对模型造成影响，该数据集仅保留10:30-18:00之间的所有LOB状态点。所选用的五支股票详细信息如下：

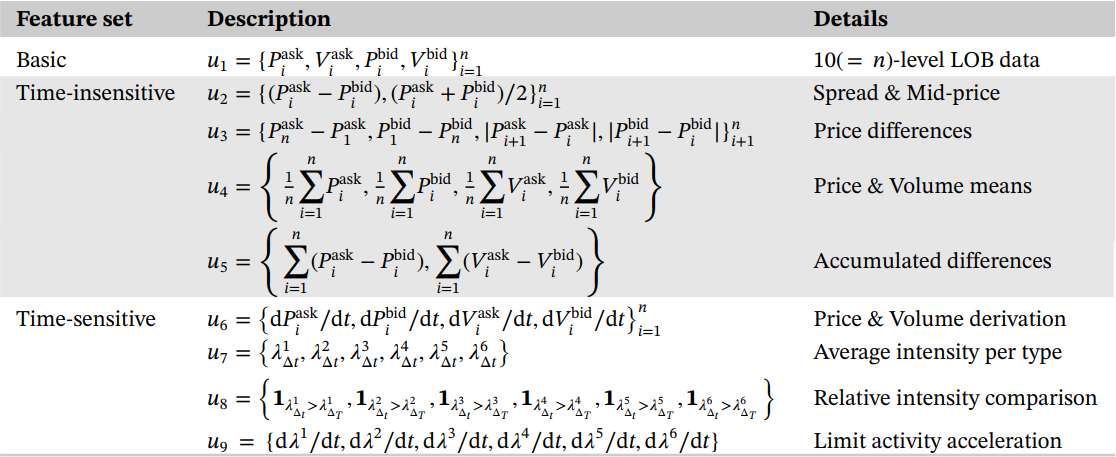
表2-1 选用的五支股票信息



### 2.2.2 特征集

2.1节中已经对LOB状态进行了形式化，此数据集则对采用更多的特征来描述LOB状态，共144个特征，可分为3大类9小类。第一类是LOB中的基本特征，和2.1节的定义完全一致，共包含10个Level的信息，即40个特征。第二类是由基本特征进行简单四则运算得到的一些进阶特征，且不考虑和的时序关系，此类特征共20+20+4+2=46个。第三类是考虑了时序关系之后的进阶特征，此类特征共40+6+6+6=58个。具体细节如下表：

表2-2 数据集的特征集合



### 2.2.3 标签集

该数据集自带5个标签，代表预测范围，定义同2.1节。

# 第3章 初步模型的设计与实现

## 3.1 相关技术介绍

### 3.1.1 CNN

CNN[20]最初是应用于图像处理领域，核心功能是挖掘出图像中的某些模式，从而帮助人们更好地去理解图像。对于常见的彩色图像，它由RGB三个颜色通道组成，每个通道中包含width×height个数值，即一张彩色图像的向量维度为，CNN的作用就是将图像简化为易于处理的形式，而不会丢失对获得良好预测至关重要的特征。CNN用于手写数字识别的示意图如下：

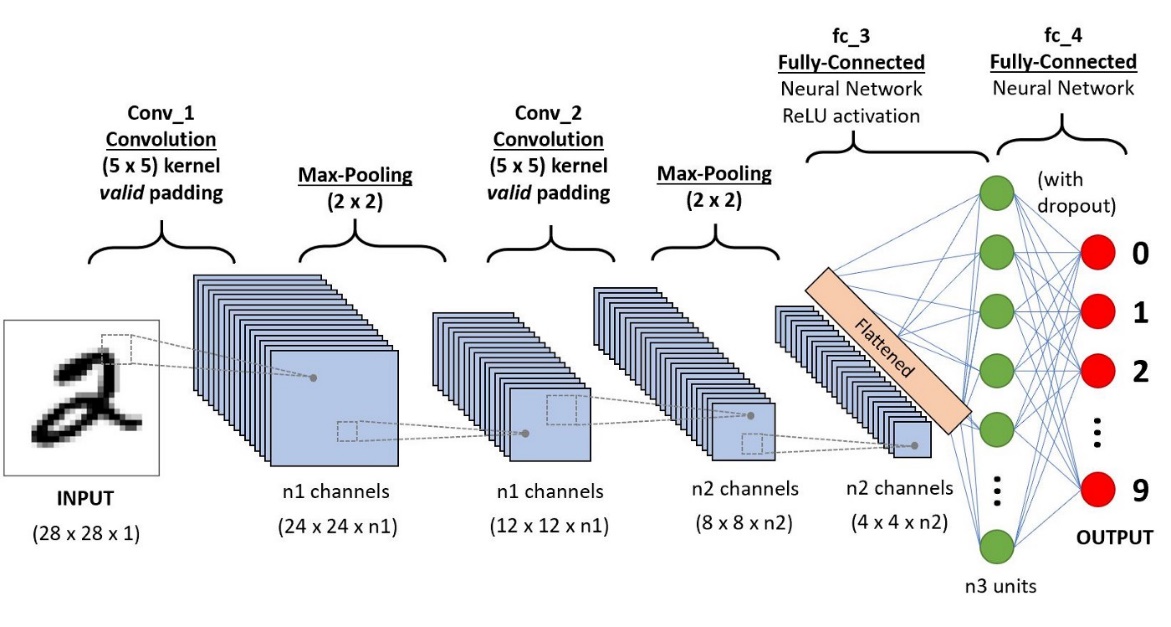


图3-1 CNN用于手写数字识别的示意图

CNN可以接收输入图像，将可学习的weights和bias分配给图像中的各个像素，在此基础上让模型可以识别出一些特征。与其他分类算法相比，CNN所需的预处理步骤要少得多。在许多传统方法中，用于提取图像特征的过滤器是手工设计的；相比较而言，CNN经过足够多次的训练，有能力将过滤器中的参数调整到较优的状态，从而更准确地捕获图像中的特征。

CNN的设计灵感来源与人脑中的视觉皮层组织，单个神经元仅仅对于有限范围内的刺激做出反应。因此，CNN的核心就在于“过滤器”和“感受区域”的设计。“过滤器”用于捕获图像中的某种特征，“感受区域”就是一块由人工设定好的范围，例如3×3，5×5等。“过滤器”每次与图像中一块“感受区域”进行卷积运算，得到的结果存入Feature Map，当“过滤器”扫完一遍图像范围后即完成了一次特征提取任务，在此之后还可以增加池化操作以及全连接层映射以完成分类等任务。

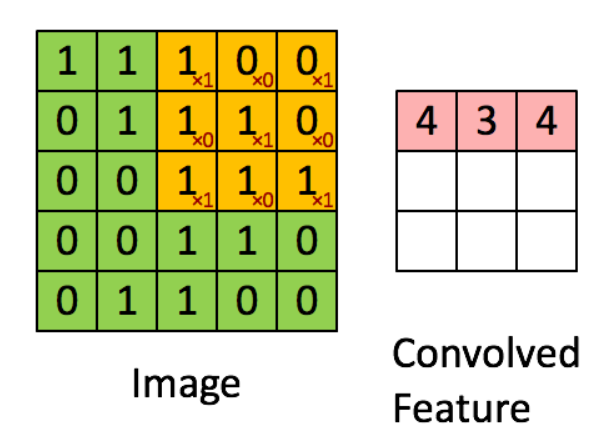


图3-2 过滤器在感受范围上做卷积运算

上图的绿色部分就是一个5×5×1的输入图片，“感受范围”大小为3×3×1，黄色部分就是“过滤器”，维度也是3×3×1，右下角乘数就是“过滤器”中的权重参数。“过滤器”和“感受范围”做完一次卷积运算后得到结果4，就是Convolved Feature中右上角的数值。

至于池化操作，可以简单理解为对一个大矩阵的信息浓缩从而得到一个小矩阵，而信息浓缩的方式可以自定义，例如max pooling就是选择提取池子中最大的数字来代表整个池子的信息而average pooling则是使用池子中所有数值的平均数来代表整个池子的信息。具体细节可参考下图：

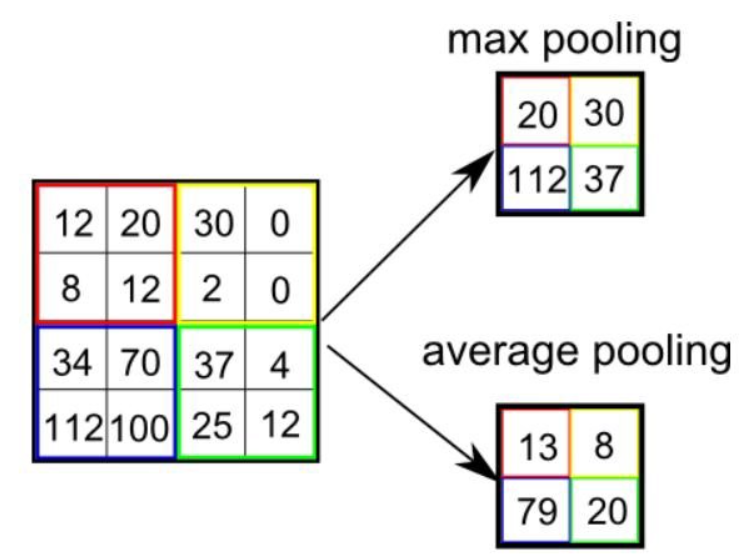


图3-3 池化操作

值得注意的是，池化操作并不是必要的，随着现在GPU算力的增强，许多学者已经在减少池化操作的使用了。总之，经过卷积操作和池化操作之后，模型已经可以更好地理解图像的特征，随后可以将得到的特征向量拉平，再经过全连接层映射到不同的类别上从而完成分类任务。

### 3.1.2 LSTM

LSTM[14]起初是为了解决传统RNN网络在面对长时依赖时出现的梯度消失问题而设计的改进版RNN。能够长时间地记住信息实际上是LSTM网络的默认行为，而不是难以学习的东西。

循环神经网络设计成重复模块链的形式，在标准RNN 中，重复模块其实是很简单的，如下图所示仅包含一个tanh激活函数：

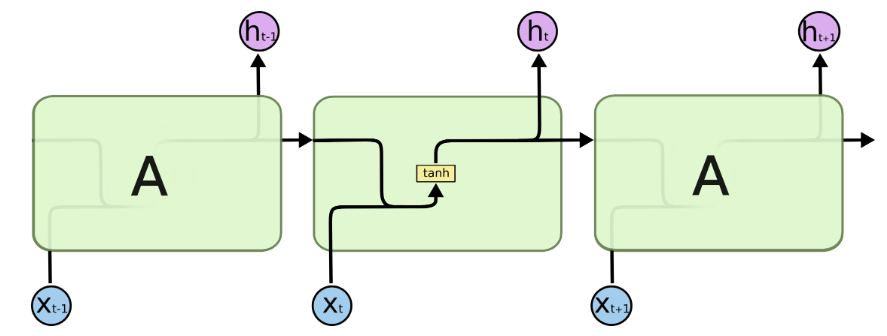


图3-4 标准RNN结构

但是LSTM中的重复模块则要复杂地多，共有4个神经网络层在以非常奇特的方式进行交互：

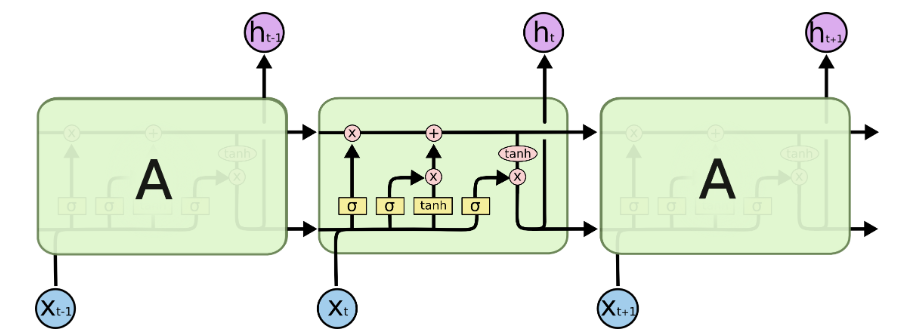


图3-5 LSTM结构

将核心的重复模块单独拿出来进行理解，输入的是当前时间步的向量，模块所记忆/携带的状态称为，最后会输出当前时间步的隐含状态，由于会沿着重复模块链一直传递下去，所以LSTM可以很自然地记住长时依赖关系。

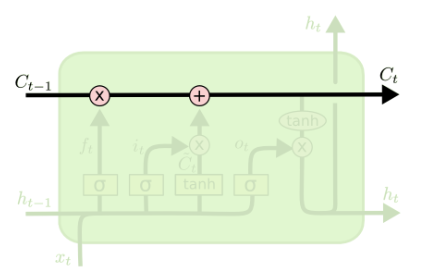


图3-6 携带的状态

至于如何从得到，LSTM利用了巧妙的gate设计来决定gate前的输入有多少可以保留到gate之后，这种选择性地让信息通过的方式可以对模型进行精细调节。首先是forget gate的设计，这一步用于决定从携带的中丢弃多少信息：



然后是input gate，这一步用于决定往中存储哪些新信息：





接下来就依据前面2个gate的输入来更新：



最后才是output gate，它用于决定将过滤多少得到最终的输出：



总结来说，LSTM通过巧妙的gate设计与传递实现了对于序列数据的长期依赖关系挖掘。

## 3.2 模型设计思路

经过2.1节的形式化和2.2节的数据集介绍，模型的输入为一个历史LOB状态序列和预测范围，模型的输出为对应的，预测短期内MidPrice涨跌情况。考虑前100个历史LOB状态，每个LOB状态仅使用数据集中的基本40个特征，故的维度为（100,40）：





从1.2节的文献综述和3.1节的CNN、LSTM技术介绍得到启发：CNN常常在深度学习模型中用于特征提取以帮助完成后续任务，LSTM可以帮助挖掘序列数据之间的时序依赖关系。那么在我们的任务中，可以将先用CNN提取中的隐藏特征，再经过一些中间层处理之后放入一个LSTM网络以抓取不同之间的时序依赖，最后再利用全连接层映射到三个类别上。

### 3.2.1 CNN层设计

由于CNN可以多次叠加使用来达到提取不同粒度范围的特征，因此本课题共使用3大卷积层，每一个卷积层的out\_channels统一设置成32。

**第一卷积层：**卷积核大小设置成（1，2），步长设置为（1，2）。结合的结构，可以发现使用大小为（1，2）的卷积核可以总结每个Level中对的信息。这里同样需要手动设置步长，因为CNN的一个重要特性是参数共享，这一特性很有吸引力，它可以避免学习过多参数导致过拟合。如果仅仅使用默认的（1，1）步长，那就意味着和是共享参数的，这显然是错误的，因为价格和数量的动态行为是不同的。

**第二卷积层：**卷积核大小和步长与第一层保持一致，由于第一卷积层已经捕获了每个Level中的独立信息，现在希望第二卷积层可以捕获跨Level的信息，设置（1，2）的卷积核与步长正好可以满足这点需求，经过这两层卷积操作之后得到的Feature Maps实际上已经包含了LOB中的micro-price特征，已有的工作[21]证明这一特征对于预测MidPrice具有帮助，但与[21]中不同的是，本课题考虑从Level-1到Level-10共10个micro-price。Level-1下的micro-price的定义为：





**第三卷积层：**这一层的作用是将特征浓缩，在我们的实验中，经过前面两个卷积层得到的Feature Maps维度为（100，10），因此这一层的卷积核大小设置为（1，10），这样就可以将Feature Maps维度转变为（100，1）

### 3.2.2 三通道中间层设计

近年来的深度学习模型中有一种较为流行的Trick，那就是将同一个输入到多个通道中分别进行处理得到多通道结果，然后将它们堆叠到一起再当作后续模型的输入。这个Trick的意义就在于多通道结果相当于对于源数据进行了多次探索性特征提取，最后的向量堆叠则是汇总这些探索结果，这大大降低了每个通道进行探索的压力。

受到这种Trick的启发，中间层的设计为：将CNN层的输出复制3份，分别放到3个通道中进行探索，最后再进行向量堆叠。前面的CNN层的卷积核第一个维度值都是1，也就是不考虑不同之间的时序关系，也没有进行池化操作；而最后的LSTM网络可以帮助我们抓取不同的长期和短期依赖关系。因此确定三通道中间层的作用如下：

**通道一：**包含2个小卷积层，卷积核大小分别为（1，1）和（3，1），即可以协助捕获前后1个时间步和3个时间步的时序关系，缓解后续LSTM层的压力。

**通道二：**同样包含2个小卷积层，卷积核大小分别为（1，1）和（5，1），功能和通道一相似，只不过是另一种探索。

**通道三：**进行池化操作，池子大小不需要设置太大，本课题使用（3，1），不然可能会丢失一些细粒度的特征。

### 3.2.3 LSTM层设计

输入向量经过CNN层处理之后分别输入到中间层的三个通道中，处理后的结果进行堆叠再输入到LSTM层。本课题的任务本质上是个分类问题，而分类问题最简单的解决办法是使用全连接层，但是这往往会需要学习大量的参数而导致训练过程特别困难。因此LSTM的主要作用有2个：首先是捕获时序特征，其次是相较于全连接层能够学习更少的参数。

为了捕获提取到的特征中存在的时序依赖关系，将全连接层替换为LSTM网络层。LSTM中每个重复模块所携带的状态会反馈给自身，并且传递给下一个重复模块，因此可以对特征的时序动态变化进行建模。在本课题中，我将隐状态的维度设置为64，然后将最后一个时间步对应的保留，在其后面接一个全连接层FC，FC的输出状态数量为3，代表涨、跌、平3种状态类别。最后加一softmax层表示3种价格变动类别的概率。

## 3.3 模型实现

根据3.2节的详细设计思路，使用Pytorch深度学习框架来实现初步模型，模型整体结构图如下：

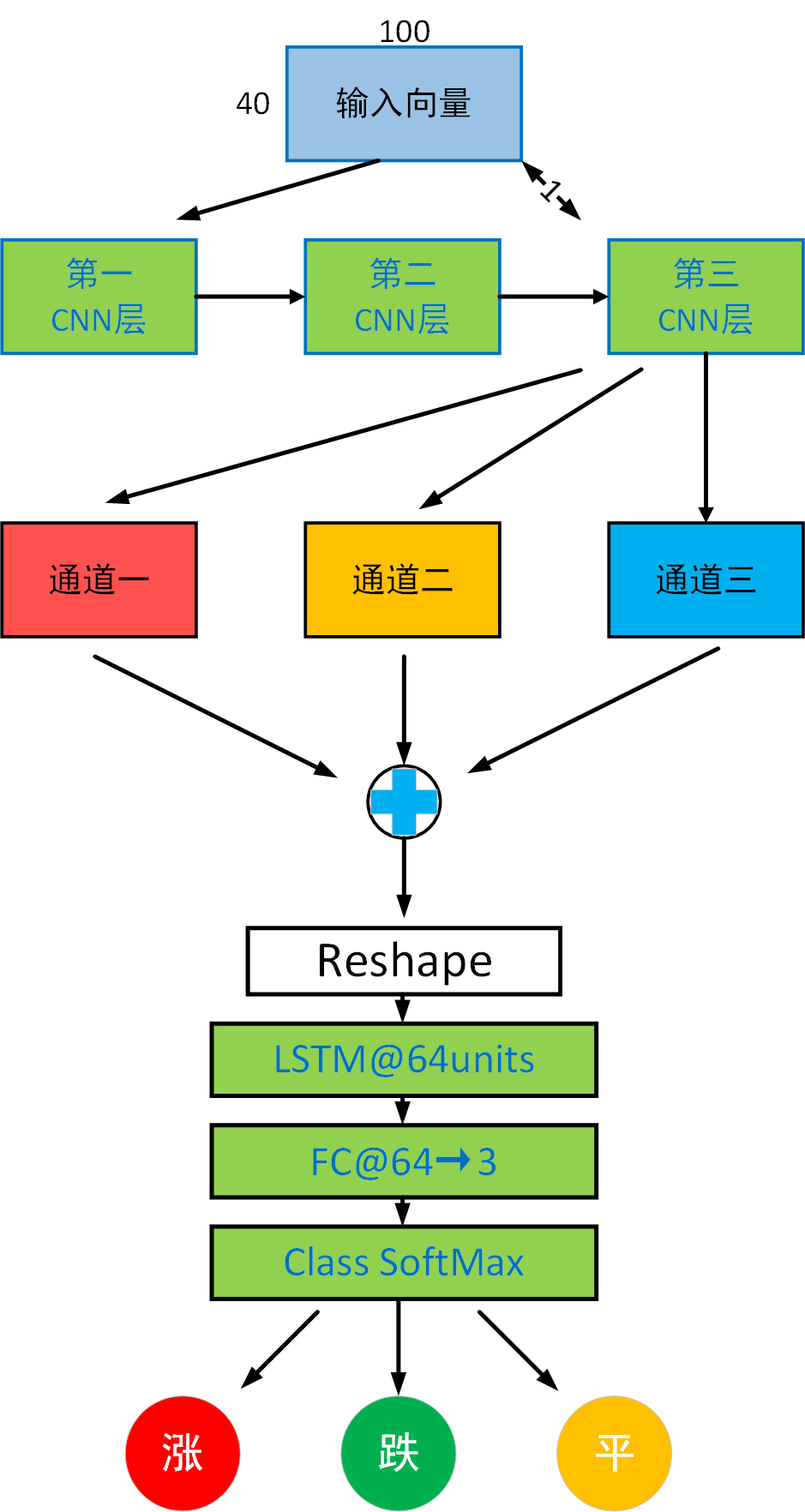


图3-7 模型整体结构图

输入向量为：





由于Pytorch中tensor的统一形状为，示意图隐去了BatchSize这一维度，仅仅展示tensor的后三个维度，输入向量为拓展了一个维度之后的结果。

输入向量首先依次经过三个CNN层；然后被复制三份分别进入通道一、通道二和通道三；将三个通道出口处得到的结果tensor进行，再经过reshape之后输入到LSTM网络中，LSTM隐状态设置为64；再经过一个全连接层和SoftMax层将结果映射到三个类别的概率分布上。

下面详细展示重点网络层的结构，首先是三个CNN层：

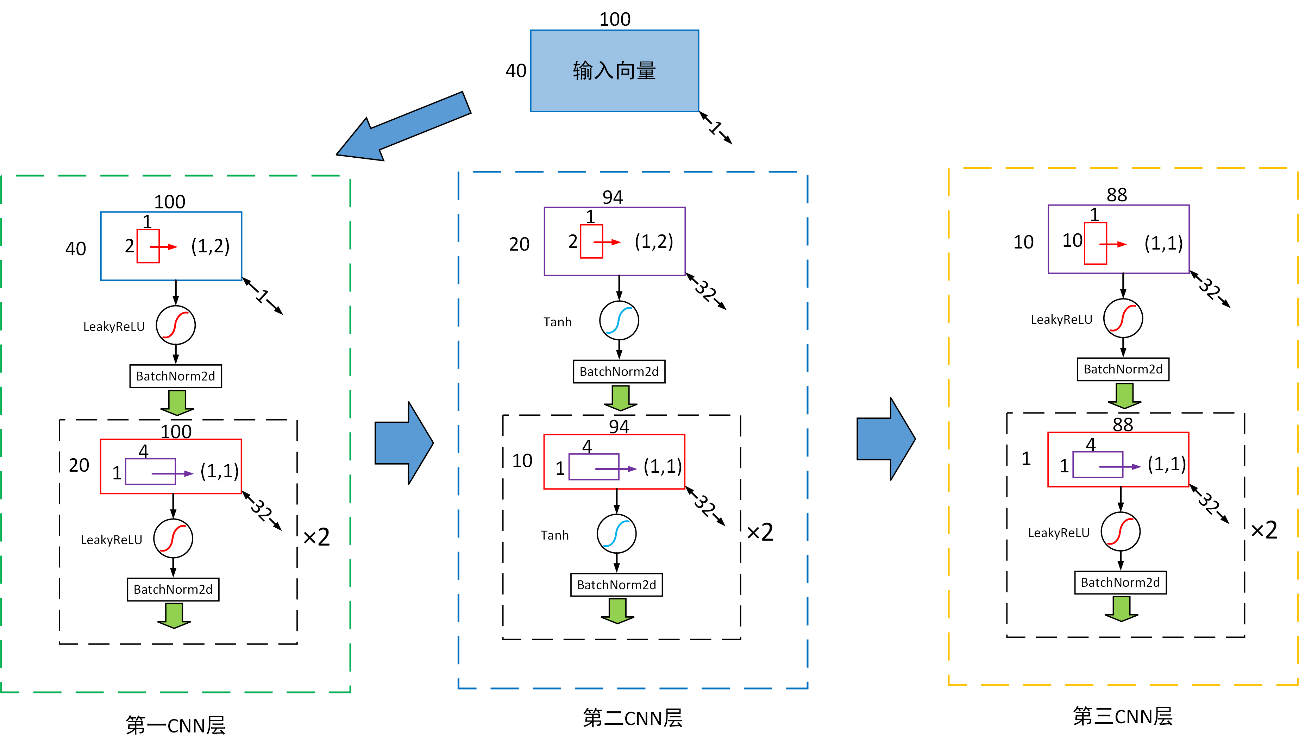


图3-7 CNN部分结构图

示意图中矩形的宽、高以及右下角双向箭头中间的数字分别代表tensor的；箭头代表数据流向；大矩形中不同颜色的小矩形大小代表卷积核Kernal的大小，小矩形右侧的箭头指向的元组代表步长Stride；圆圈加上一段曲线代表激活函数。

**第一CNN层：**输入向量经过第一次卷积操作，卷积核大小为（1，2），步长也为（1，2），之后得到的Feature Maps形状为（32，100，20），紧跟着的是一个激活函数LeakyReLU和归一化操作BatchNorm2d；然后经过两个重复的子网络，即先有一次卷积，卷积核大小与步长分别为（4，1）和（1，1），然后同样紧跟LeakyReLU和BatchNorm2d。此层的输出Feature Maps形状为（32，94，20）。

**第二CNN层：**此层的构造和第一层基本一致，唯一的区别就在于激活函数换成了Tanh。此层的输入Feature Maps形状为（32，88，10）。

**第三CNN层：**此层的构造也和第一层基本一致，唯一的区别在于将第一次卷积操作的卷积核大小与步长切换为（1，10）和（1，1）以便于将特征浓缩。此层的输入Feature Maps形状为（32，82，1）。

接下来是三通道结构图：

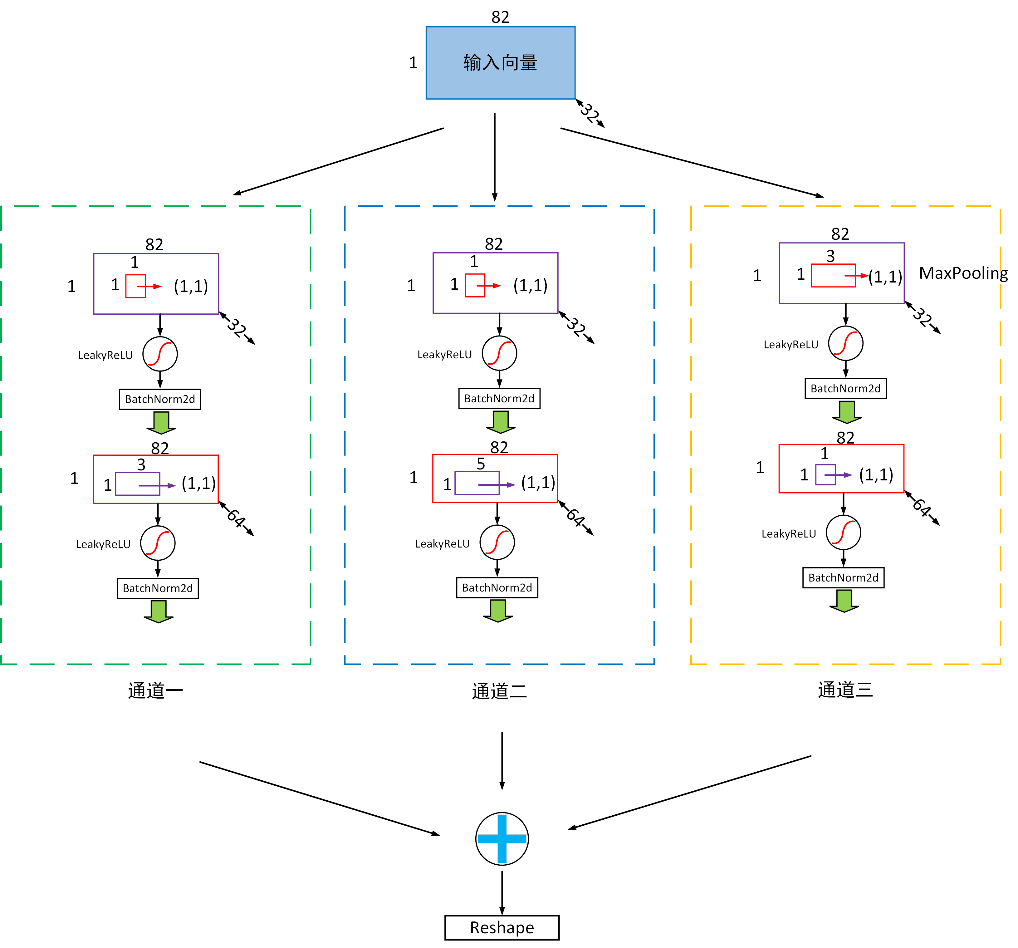


图3-8 三通道结构图

**通道一：**CNN层的输出向量先经过第一次卷积操作，输出channels为64，卷积核大小为（1，1），步长也为（1，1），得到的Feature Maps形状为（64，82，1），紧跟着LeakyReLU和BatchNorm2d；然后经过一次类似流程，但是卷积核大小变为（3，1）。此通道的输出Feature Maps形状为（64，82，1）。

**通道二：**和通道一类似，仅将第二次卷积核大小变为（5，1），此通道的输出Feature Maps形状为（64，82，1）。

**通道三：**和通道一类似，将第一次卷积操作变为池化操作MaxPooling并将第二次卷积核大小变为（1，1），此通道的输出Feature Maps形状为（64，82，1）。

最后将三个通道的输出进行叠加并reshape，得到的输出向量形状为（82，192）以符合后续LSTM对于输入向量形状的需求。

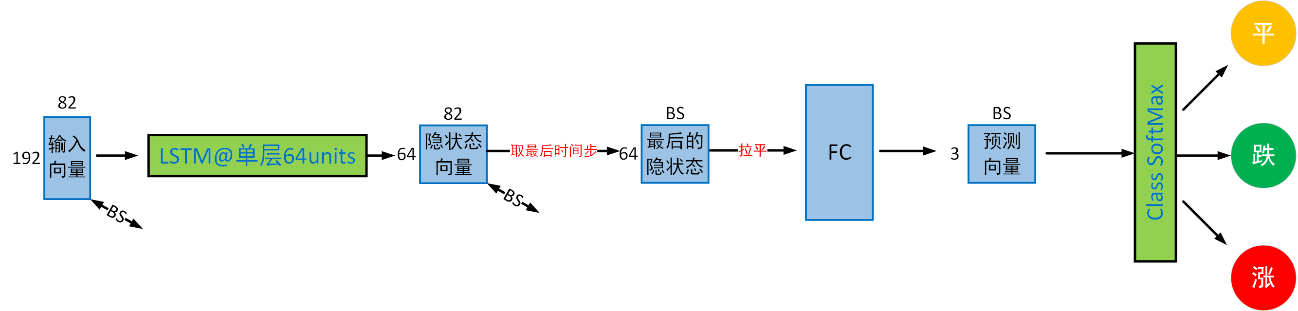


图3-9 LSTM部分结构图

上面的LSTM部分结构图将BatchSize维度展示出来，输入向量的形状就是（BatchSize，82，192）。LSTM网络设置为单层，隐状态数量设置为64，即64个LSTM单元，得到隐状态向量之后取最后一个时间步，得到的向量形状为（BatchSize，64），再经过后续处理得到三个类别的概率分布，取最大概率所在的类别作为预测的类别输出。

## 3.4 交易策略

依据2.1节的交易策略形式化以及3.3节的预测模型实现，可以设计一个简单的交易策略以供回测盈亏实验使用：

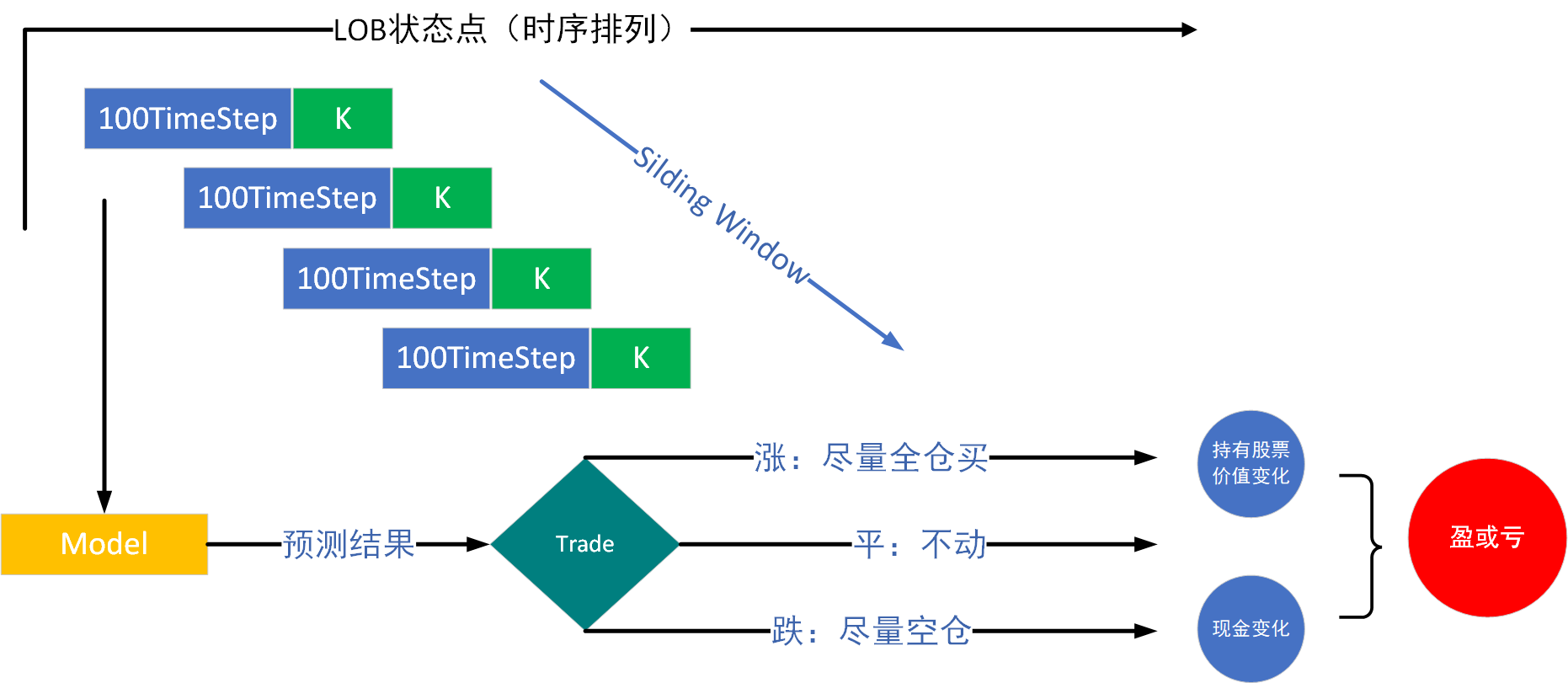


图3-10 交易策略示意图

如上图所示，交易策略的执行方式类似于“滑动窗口”：读取一段数据，执行相应的动作，然后再次读取下一段数据来执行动作。按照时序读取LOB的状态点，每一次都使用过去100个时间步的LOB状态序列作为预测模型的输入，预测未来个时间步内MidPrice的涨跌。根据预测结果分别做出不同的交易动作，而这些交易动作会引起账号的现金变化以及手中持有股票的价值变化，连续模拟交易一段时间之后便可以得到盈亏情况。

# 第4章 改进模型设计与实现

## 4.1 从模型角度的改进

## 从特征角度的改进

### Volume Order Imblance

一些已有的研究[22][23]表明Volume Order Imblance对于预测MidPrice趋势有帮助，这一指标用于衡量当前市场中买卖双方的不同压力，它的定义如下：







其中，分别代表中BestAsk与其对应的Volume，BestBid与其对应的Volume。

用来表征相较于增加的买方压力：若则表明要么有交易者取消了他的Limit Buy Order要么是有一个Order以成交，仅从这两个数据无法判断真实情况到底是属于哪一类，所以将设为0。若，就可以使用来代表增加的买方压力。最后，若，可以将其解释为由于交易者打算以更高的价格购买而导致的价格上涨势头，因此使用来表征买方压力即可。用来表征相较于增加的卖方压力，对于它的理解与类似。

### Order Imblance Ratio

### 应用到初版模型

## 最终模型

# 第5章 实验设计与实现

## 5.1 比较初步模型和baseline

### 5.1.1 模型指标实验

### 5.1.2 回测盈亏实验

## 5.2 比较改进后模型和初步模型

### 5.2.1 模型指标实验

### 5.2.2 回测盈亏实验

# 结 论

在本次毕业设计论文中，我们在边缘硬件上实现了基于有损压缩思想，可对压缩时序数据进行error-bound统计查询的Plato算法和基于无损压缩思想，可对压缩时序数据进行低代价随机访问的GD算法。其中：

1. 对于Plato算法，我们利用了作者提出的误差向量正交投影性质对时序数据乘法并聚集的误差界做了进一步改进。
2. 对于GD算法，通过改进预处理时的算法终止条件避免了预估空间落入局部最优值。

最后通过实验证明了Plato算法给出的误差界具有合理性，且对误差界的改进使均值相差较大的时序数据相关系数查询操作的误差界有很大程度的降低；GD算法的压缩效果比现存的优秀数据压缩算法略差，但牺牲少量压缩率换取的随机访问效果非常可观，且在压缩器状态恢复后的随机访问代价进一步降低。

我们还提出了一些对时序数据压缩查询未来的展望：

1. 随着边缘硬件在智慧城市中的大规模部署，对数据的压缩和查询需求不再拘泥于单机处理，可对分布式情况的压缩与查询做进一步探索。
2. 对于有损压缩，支持近似查询的算法，可以尝试探索拟合函数的更好的性质，对误差界做进一步改进。
3. 对于无损压缩，支持随机访问的算法，可以尝试改进原有启发式算法，使其得出的参数更加合理。

# 参考文献

[1] E. Zivot and J. Wang, “Vector autoregressive models for multivariate time series,” Modeling Financial Time Series S-PLUS, pp. 385–429, 2006.

[2] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, “Stock price prediction using the ARIMA model,” in Proc. 16th IEEE Int. Conf. Comput. Model. Simulation 2014, pp. 106–112.

[3] Gourieroux, C., Jasiak, J. and Fol, G. L. (1999). Intra-day market activity. *Journal of Financial Markets* **2**, 193–226.

[4] Bouchaud, J. P., Mezard, M. and Potters, M. (2002). Statistical properties of stock order books: Empirical results and models. *Quantitative Finance* **2**, 251–256.

[5] Smith, E., Farmer, J. D., Gillemot, L. and Krishnamurthy, S. (2003). Statistical theory of the continuous double auction. *Quantitative Finance* **3**, 481–514.

[6] Cont, R. and de Larrard, A. (2011). Price dynamics in a Markovian limit order market. *SSRN eLibrary*.

[7] Cont, R., Stoikov, S. and Talreja, R. (2010). A stochastic model for order book dynamics. *Operations Research* **58**, 549–563.

[8] W. Huang, Y. Nakamori, and S.-Y. Wang, ‘‘Forecasting stock market movement direction with support vector machine,’’ Comput. Oper. Res., vol. 32, no. 10, pp. 2513–2522, Oct. 2005.

[9] K. J. Kim, ‘‘Financial time series forecasting using support vector machines,’’ Neurocomputing, vol. 55, nos. 1–2, pp. 307–319, Sep. 2003.

[10] L. J. Cao and F. E. H. Tay, ‘‘Support vector machine with adaptive parameters in financial time series forecasting,’’ IEEE Trans. Neural Netw., vol. 14, no. 6, pp. 1506–1518, Nov. 2003.

[11] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, “Temporal bag-of-features learning for predicting mid price movements using high frequency limit order book data,” IEEE Trans. Emerg. Topics Comput. Intell., to be published.

[12] A. Ntakaris, M. Magris, J. Kanniainen, M. Gabbouj, and A. Iosifidis, “Benchmark dataset for mid-price prediction of limit order book data with machine learning methods,” J. Forecasting, vol. 37, no. 8, 852–866, 2018.

[13] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, “Forecasting stock prices from the limit order book using convolutional neural networks,” in Proc. IEEE 19th Conf. Business Inform., 2017, vol. 1, pp. 7–12.

[14] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[15] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” IEEE Trans. Neural Netw., vol. 5, no. 2, pp. 157–166, 1994.

[16] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in Proc. Advances Neural Inf. Process. Syst., 2014, pp. 3104–3112.

[17] J. Sirignano and R. Cont, “Universal features of price formation in financial markets: perspectives from deep learning,” arXiv preprint arXiv:1803.06917.

[18] C. Carrie, “The new electronic trading regime of dark books, mashups and algorithmic trading,” Trading, vol. 2006, no. 1, pp. 14–20, 2006.

[19] O'Hara, M., & Ye, M. (2011). Is market fragmentation harming market quality? Journal of Financial Economics, 100(3), 459–474.

[20] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324.

[21] A. Lipton, U. Pesavento, and M. G. Sotiropoulos, “Trade arrival dynamics and quote imbalance in a limit order book,” 2013, arXiv preprint arXiv:1312.0514.

[22] Tarun Chordia and Avanidhar Subrahmanyam. Order imbalance and individual stock returns: Theory and evidence. Journal of Financial Economics, 72:485–518, 2004.

[23] Han-Ching Huang, Yong-Chern Su, and Yi-Chun Liu. The performance of imbalance-based trading strategy on tender order announcement day. Investment Management and Financial Innovations, 11(2):38–46, 2014.

[24]

# 哈尔滨工业大学本科毕业设计（论文）原创性声明

本人郑重声明：在哈尔滨工业大学攻读学士学位期间，所提交的毕业设计（论文）《时间序列大数据压缩与查询》，是本人在导师指导下独立进行研究工作所取得的成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明，其它未注明部分不包含他人已发表或撰写过的研究成果，不存在购买、由他人代写、剽窃和伪造数据等作假行为。

本人愿为此声明承担法律责任。

作者签名： 日期： 2020 年 6 月 4 日



# 致 谢

冬来春往，寒暑交替。赴哈尔滨求学的第四年，随着毕业论文的行文至此，大学生活也将画上句号。

感谢父母对我从小的辛勤培养，为了让我接受良好的教育不断奔波，我能进入哈工大并顺利完成学业离不开你们的支持。

感谢我的老师高宏教授，在我刚刚笨拙地进入科研的大门时，是您耐心地做我的领路人，从选题到定稿的每一个阶段都少不了您的悉心指导。

感谢原声吉他社的小伙伴们，在我为论文进展缓慢而烦闷时，是你们陪我放松心情，整理好心态重新面对挑战。

最后，感谢所有成长路上给予过我帮助与温暖的亲人、朋友们，我将不负你们的期望，心怀感恩继续前行！