

博金杯量化模型大赛算法报告

参赛队员：段志健¹，李世成²，吴童¹

指导老师：王峰²，邓小铁¹

¹ 北京大学 ² 武汉大学

`zjduan@pku.edu.cn`, `lishicheng@whu.edu.cn`,

`wutong1109@stu.pku.edu.cn`, `fengwang@whu.edu.cn`,

`xiaotie@pku.edu.cn`

Abstract

本文介绍了我们在博金杯量化模型大赛中采用的策略。本次比赛提供分钟级的实时股票数据，需要参赛选手训练模型预测股票涨跌并向系统发送买卖信号。我们通过训练基于 TCN 和 Transformer 的 TTR 神经网络模型，利用 TCN 提取股票的时序关系特征，并通过 Transformer 结合预先提取的分类信息，以进一步提升模型的预测效果。我们的模型预测结果为给定股票在下一分钟的涨跌百分比，和股票在未来十分钟的平均涨跌比。基于这些预测结果，我们综合考虑当前持仓股票成交价、下一分钟股票平均成交价预测值、未来十分钟股票平均成交价预测值三者的大小关系，设计了一套股票交易策略，旨在优化我们的收益。我们的策略最终在比赛中取得了第六名的成绩。

1 引言

随着互联网和大数据技术的迅猛发展，机器学习在金融市场的应用逐渐成为研究和实践的焦点 [11, 4]。这一趋势得益于机器学习算法能够从庞大的金融数据中提取模式、识别趋势，并作出相应的预测。这些方法不仅提供了更深入的市场理解，而且为投资者提供了强大的工具来制定更智能、基于数据的投资决策。

本文介绍了我们参加 2023 年 CCF 计算经济学大赛-博金量化模型挑战赛¹所采用的技术方案。本次比赛的题目为利用机器学习算法来预测股票日内走势，预测目标具体为股票在未来一段时间内的收益率或涨跌方向，最终发出股票的买卖信号最大化收益。比赛提供了分钟级别的历史行情数据与实时行情数据接口，参赛队伍需要基于历史数据训练模型，并在实时行情数据到达时进行预测并发出交易信号。参赛的股票范围为中证 500 指数的成分股，模型的预测频率为分钟级别。评价标准是参赛者通过模拟交易接口，利用模型实时产生的交易信号的累积收益。计算公式如下：

$$\text{Score} = \text{PNL} \times \text{WinRate} \times \ln(\text{trades}) \quad (\text{score})$$

其中，PNL 表示每笔交易的收益；WinRate 表示胜率，也就是盈利交易次数占比；Trades 为交易次数。多个股票多个交易日的 Score 会累加。

借鉴卷积神经网络和 Transformer [13] 等模型在金融时间序列预测方面的成功 [2, 12, 1, 15, 3, 14]，本研究旨在将这些先进技术融入我们设计的模型中，以更全面、准确地进行股票走势预测。通过将 TCN [7] 用于提取股票时序关系特征，结合 Transformer 技术和预先提取的分类信息，我们进一步提升了模型的预测性能。

与仅仅预测下一分钟股票平均成交价相比，我们的模型更进一步，额外预测了股票未来十分钟的平均涨跌比。这种短期和中期预测的结合赋予我们的模型独特的优势，使其能够更精准地捕捉股票即将经历的高低点。通过综合考虑当前持仓股票的平均成交价，我们成功实施了一套灵活而高效的策略，允许在高点卖出、低点买入，从而优化投资回报。

在博金杯量化模型大赛中，我们的策略最终取得了第六名的成绩，验证了我们模型的实用性和性能。通过本文的进一步探讨，我们将深入剖析我们的模型和策略，为未来量化金融研究提供新的思路和方法。

¹<https://ccf.pkucxpl.com/>

2 市场和数据描述

2.1 数据描述

本次比赛提供了丰富的股票行情历史数据，包含了中证 500 指数成分股在 2022 年 9 月 2 日至 2023 年 9 月 1 日的分钟级别价格信息。每支股票在每一天从上午九点三十分开盘至下午三点收盘，提供分钟级别股票数据，其中每条分钟级数据包括以下信息：股票代码、日期（年月日）、时间（精确到分钟）、开盘价、最高价、最低价、收盘价、vwap（即成交量加权平均价格）、成交金额、成交量。

2.2 数据预处理

我们对数据进行了预处理，包括以下几个步骤：

1. 数据清洗：去除无效数据，包括价格为 0 的数据和成交量为 0 的数据。
2. 特征增广：我们根据股票代码，使用 Wind 导出了每支股票的股票简称、Wind 一级行业、Wind 二级行业、公司属性四个额外的特征，并将其与原始数据合并。
3. 数据归一化：由于每只股票的初始价格不同，且相差较大，而在交易过程中我们往往更加关注价格的涨幅情况，即 $\rho_t = p_t / p_{t-1}$ 。其中 p_t p_{t-1} 表示当前时刻和上一时刻股票的价格。所以，为了避免股票初始价格对预测结果可能造成的影响，我们将每只股票原始数据中的开盘价、收盘价、最高价、最低价的价格数据全部都以股票的最新 vwap 进行归一化。此外由于原始数据中股票的成交金额和成交量数值较大，神经网络难以收敛。因此我们采用缩放因子 β_{money} 和 β_{volume} 将这两个特征归一化到一个合适的范围，可以减小这种数值计算的不稳定性，即 $money = money * \beta_{money}$, $volume = volume * \beta_{volume}$ 。

2.3 市场描述

本次比赛的预测启动日为 11 月 6 日开盘至 11 月 30 日封盘。整体来看，本比赛期间中证 500 指数呈现震荡下行的状态。开盘后的前两周呈现反弹行情，随后步入

调整，至 11 月底跌破关键支撑位，技术面转为空头。具体来看,11 月 6 日开盘价为 3708 点；之后指数反弹至 11 月 16 日的 3860 点左右，涨幅达近 4%；此后开始下跌，至 11 月 30 日收于 3546 点，期间跌去近 5%。从成分股情况看, 根据 Wind 统计, 本期间中证 500 成分股平均涨幅为 -1.38% ，显示了小幅调整特征。分行业看, 医药生物、有色金属等周期和资源板块处于相对强势, 而消费、餐饮旅游、教育类股份则表现疲软。市场风格轮动明显。整体来说, 本比赛期间中证 500 及成分股可谓是处于一个典型的震荡下行、积极性不足的结构性调整阶段，如何把握其中的交易机会和风险是参赛者面临的难点。

比赛期间，所有参赛队在每个交易日以空仓开盘，每天封盘时强制平仓，没有隔夜仓。每个交易时段只有一个单位头寸，上一个信号的反向信号会平掉上一个信号的仓位同时开本信号的仓位。因此，比赛的关键在于预测股票的涨跌，判断发出买卖信号的时机。

3 理论基础和文献综述

目前基于深度学习的股票价格趋势预测主要通过深度神经网络对价格序列进行时序预测，通常采用的模型有循环神经网络 (RNN) [10] 和长短时记忆网络 (LSTM) [9]、卷积神经网络 (CNN) [12]、时序卷积神经网络 (TCN) [7] 以及 Transformer [13] 等。我们接下来详细介绍本次比赛使用的 TCN 和 Transformer。

3.1 TCN

TCN (Temporal Convolutional Network) 是一种基于卷积神经网络 (CNN) 的时序模型，专门设计用于处理时序数据。TCN 在处理时序数据时展现出对长期依赖关系的较好捕捉能力，并且相较于传统的 RNN 和 LSTM，具有更好的并行性，因此训练速度较快。此外，TCN 还可以有效缓解梯度消失问题，使得模型更容易训练。这对于深层网络的建模是一个重要的优势，特别是在处理长序列时。TCN 的应用领域涵盖了语音识别、自然语言处理以及时间序列预测等多个领域。以下是 TCN 的一些关键特点和组成部分：

- 膨胀卷积层是 CNN 中的一种变体，也被称为扩张卷积或空洞卷积。与传统卷

积层相比，膨胀卷积引入了一个膨胀因子 (dilation factor)，该因子定义了卷积核内元素之间的间隔。这使得膨胀卷积能够在不增加参数的情况下有效地扩大感受野，从而更好地捕捉输入序列中的长距离依赖关系。膨胀卷积的数学表达式为：

$$y[t] = \sum_{i=1}^k w[i] \cdot x[t + (d \cdot (i - 1))] \quad (1)$$

其中， $y[t]$ 是卷积层的输出； $w[i]$ 是卷积核的权重； $x[t + (d \cdot (i - 1))]$ 是输入序列的对应位置； d 是膨胀因子。相比于普通卷积，膨胀卷积有以下特点：1) 感受野扩大：通过增加卷积核中元素的间隔，膨胀卷积能够捕捉更大范围的输入信息，有助于处理序列中的长期依赖关系。2) 参数共享：与传统卷积相比，膨胀卷积不引入额外的参数，因此具有相对较小的计算负担。3) 高效并行计算：膨胀卷积在计算时具有较好的并行性，因为它可以同时处理不同位置的输入。

- 因果卷积 (Causal Convolution) 是一种卷积操作，其目的是确保每个时间步的输出仅依赖于过去的观测值，而不受未来信息的影响。这在处理时序数据时尤为重要，因为在许多情况下，我们只能观察到过去的信息而无法获得未来信息。在因果卷积中，通常会使用零填充 (zero-padding) 来实现，确保卷积核在滑动过程中不会窥视到未来的时间步。数学上，因果卷积的表达式为：

$$y[t] = \sum_{i=1}^k w[i] \cdot x[t - (i - 1)] \quad (2)$$

其中， $y[t]$ 是卷积层的输出； $w[i]$ 是卷积核的权重； $x[t - (i - 1)]$ 是输入序列的对应位置。因果卷积确保卷积核只与当前时间步及之前的时间步进行卷积运算，因此不会使用未来的信息。这对于时序数据的建模非常关键，尤其在预测未来值时，防止将未来信息泄漏到模型中。因果卷积通常被用于处理时序数据，以更好地捕捉长期依赖关系，并确保模型的预测是基于过去信息而非未来信息。

- 残差连接 (Residual Connections) 是一种深度神经网络中的技术，旨在解决深层网络训练过程中的梯度消失问题。它的核心思想是引入跳跃连接，将输入直接与输出相加，形成残差块。这种结构允许梯度更容易地流经网络，特别适用于非常深层的神经网络。残差连接的数学表达式为：

$$output = ReLU(Conv(input) + input) \quad (3)$$

其中， $input$ 是残差块的输入； $Conv(input)$ 是卷积操作； $ReLU$ 是修正线性单元激活函数； $output$ 是残差块的输出。引入残差连接可以避免深层网络

中梯度消失或梯度爆炸的问题，更容易训练深层网络，尤其是在数十、数百层的深度模型中，有助于稳定训练。

- 堆叠层次结构。堆叠层次结构在深度学习模型中将多个相同或类似类型的层堆叠在一起，以逐渐提取更高级别的特征或表示。这种结构的使用旨在增加模型的表示能力，允许模型学习到数据中更复杂的抽象特征。TCN 可以由多个基本的 TCN 卷积层组成，这些卷积层可以堆叠在一起形成一个深层次的结构，逐渐提取更高级别的时序特征。一个简化两层的堆叠结构如下：

$$output = TCN(TCN(input)) \quad (4)$$

总体来说，TCN 以其膨胀卷积、残差连接和因果卷积的结合并采用层次堆叠的结构，形成了一个强大的时序建模框架，能够更有效地处理时间序列数据的长期依赖关系。

3.2 Transformer

Transformer 是一种基于注意力机制（attention mechanism）的深度学习模型，该模型主要应用于自然语言处理领域，特别是在机器翻译任务中取得了卓越成果 [13]。然而，由于其强大的序列建模能力和可扩展性，Transformer 也逐渐被用于其他领域，包括计算机视觉 [6]、语音识别 [5] 以及金融领域的时间序列分析 [15]。

Transformer 结构使用多头自注意力模块建模不同词项（token，比如机器翻译中句子的所有单词）之间的交互。不失一般性，我们记输入为：

$$E_{\text{input}} = (e_1, e_2, \dots, e_n)^T \in \mathbb{R}^{n \times d},$$

其中 n 是词项的数量， d 是每个特征向量 e_i 的维度。

记 d_h 为 transformer 的隐层维度， H 为 transformer 头部（即子空间）的数量。对于头部 $h \in [H]$ ，我们使用键值注意力机制来决定什么样的特征组合对于这个子空间而言是有意义的。特定地，对每个词项 $i \in [n]$ ，我们首先计算词项 i 和词项 j 在头部 h 中的相关性：

$$\alpha_{i,j}^{(h)} = \frac{\exp(\psi^{(h)}(e_i, e_j))}{\sum_{k=1}^n \exp(\psi^{(h)}(e_i, e_k))},$$

其中

$$\psi^{(h)}(e_i, e_j) = \langle W_{\text{query}}^{(h)} e_i, W_{\text{key}}^{(h)} e_j \rangle,$$

是一个注意力函数，定义了头部 h 下词项 i 和词项 j 的相似性。 $\langle \cdot, \cdot \rangle$ 是点积操作， $W_{\text{query}}^{(h)}, W_{\text{key}}^{(h)} \in \mathbb{R}^{d' \times d}$ 是把原嵌入空间 \mathbb{R}^d 映射到 $d' = \frac{d_h}{H}$ 维度空间 $\mathbb{R}^{d'}$ 的投影矩阵。

接下来，我们把所有相关特征结合，来更新词项 i 的表示。这一步是通过计算加权系数 $\alpha_{i,j}^{(h)}$ 来得到的：

$$\tilde{e}_i^{(h)} = \sum_{j=1}^n \alpha_{i,j}^{(h)} (W_{\text{value}}^{(h)} e_j) \in \mathbb{R}^{d'},$$

其中 $W_{\text{value}}^{(h)} \in \mathbb{R}^{d' \times d}$ 。因为 $\tilde{e}_i^{(h)} \in \mathbb{R}^{d'}$ 是词项 i 和所有相关词项的组合，所以它代表一个新的组合过的特征。

然后，我们收集从各种子空间得到的所有组合特征：

$$\tilde{e}_i = [\tilde{e}_i^{(1)}; \tilde{e}_i^{(2)}; \dots; \tilde{e}_i^{(H)}] \in \mathbb{R}^{Hd'} = \mathbb{R}^{d_h},$$

其中 \oplus 代表链接操作， H 是头部的总数量。

最后，一个针对词项的多层全连接网络 (MLP) 被用于每个词项 i ，我们得到了词项 i 的一个全新的表示。

$$e'_i = \text{MLP}(\tilde{e}_i) \in \mathbb{R}^{d_h},$$

最后的输出为

$$E_{\text{output}} = (e'_1, e'_2, \dots, e'_n)^T \in \mathbb{R}^{n \times d_h}.$$

注意到 transformer 中需要优化的参数为 $W_{\text{query}}^{(h)}, W_{\text{key}}^{(h)}, W_{\text{value}}^{(h)} \in \mathbb{R}^{d' \times d}$ (对任意的 $h \in [H]$) 和最后一个针对词项的多层全连接神经网络，所有的参数都和词项总数 n 无关。此外，transformer 是一个满足排列等价性的网络结构。

4 策略假设

深度神经网络在使用历史股票数据进行训练后，能够提供相对准确的下分钟股票平均成交价预测和未来十分钟内股票平均成交价预测。我们可以直接比较预测下分钟股票平均成交价与上次交易价格来判断买卖时机，也可以在此基础上通过判断下分钟股票平均成交价预测和未来十分钟内股票平均成交价的关系来获得股票走势的更多信息。在本次比赛中，我们的交易策略如下：

1. 如果预测下分钟股票平均成交价超过预测未来十分钟内股票平均成交价，差值达到某一阈值，则认为股票接下来有下跌趋势；倘若此时持仓，并且预测下分钟股票平均成交价高于买入价格（若有），我们认为股票达到空头，应该发送卖出信号。
2. 如果预测未来十分钟内股票平均成交价超过预测下分钟股票平均成交价，差值达到某一阈值，则认为股票接下来有上涨趋势；倘若此时未持仓，并且预测下分钟股票平均成交价低于上次卖出价格（若有），我们认为股票达到低点，即将反弹，应该发送买入信号。

5 模型建构

在本节主要介绍我们提出的混合网络模型（TTR, TCN-TRansformer）来对股票价格趋势进行预测。如图1所示，TTR 混合网络模型由两个主要组件组成分别是 TCN 和 Transformer，这两个组件在整个模型中协同工作，以提高预测性能。TCN 主要负责提取股票价格的时序特征信息，能够有效地捕捉不同时间尺度下的特征，通过卷积操作实现对时序数据的层级建模。除了时序特征信息外，为了进一步提高预测准确率，我们还引入了不同股票的分类信息，如同属于医疗产业的股票在新冠时期可能均有较大幅度地上涨。因此，我们使用 Transfomer 进一步提取了股票 Wind²的分类信息，包括 Wind 一级行业、Wind 二级行业以及公司属性等。最后，我们将 TCN 提取的时序特征和 transformer 提取到的股票分类信息拼接起来，最后通过一个全连接神经网络来输出最后的预测结果。

5.1 时序特征提取

如 3.1 节所示，TCN 是一种适用于处理时间序列数据的神经网络结构，其卷积层可以有效提取时序数据中的特征。这里使用 $X_t^i \in \mathbb{R}^d$ 表示股票 i 在时刻 t 具有 d 个不同的基本数据（包括开盘价、收盘价、最高价、最低价等），则 m 只股票的市场特征可以表示为 $X_t \in \mathbb{R}^{d \times m}$ 。由于考虑股票价格趋势预测不仅跟 t 时刻的股票价格有关，还需要考虑过一段时间时间内的历史价格。这里我们考虑时间窗口为 k 的情况，

²<https://www.wind.com.cn/mobile/WFT/zh.html>

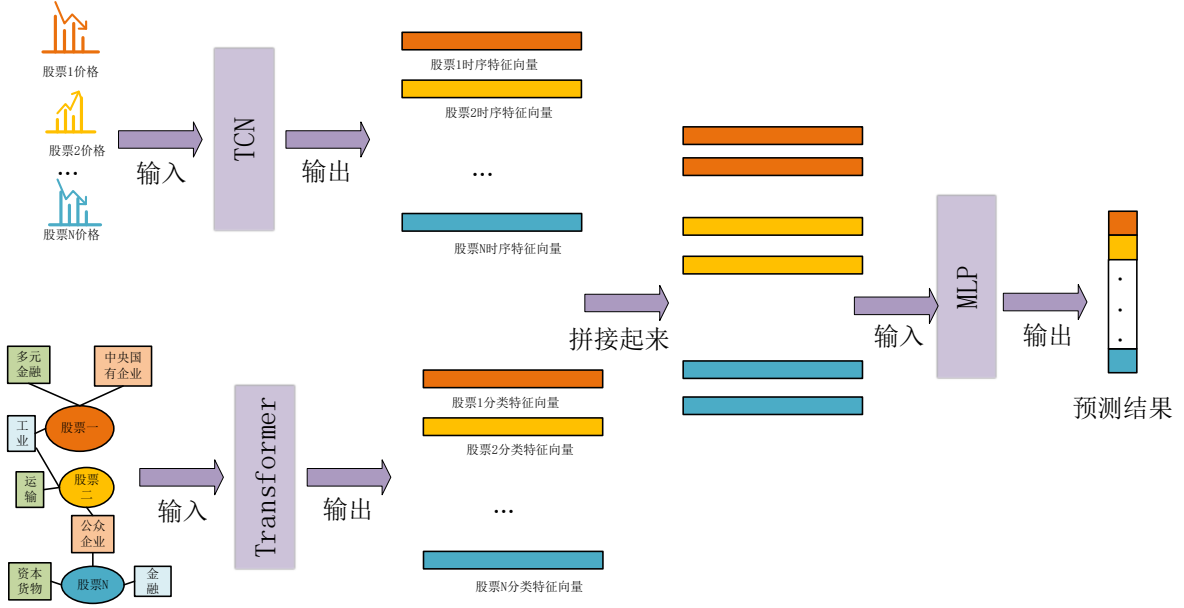


Figure 1: TTR 结构示意图

用 $\mathcal{X}_t = \{X_{t-k}, \dots, X_{t-1}\} \in \mathbb{R}^{d \times m \times k}$ 表示所有股票最近 k 个时间段的价格情况，将其作为 TCN 的输入。

$$H_t^{tcn} = TCN(\mathcal{X}_t) \quad (5)$$

其中 $H_t^{tcn} \in \mathbb{R}^{m \times d_h}$ 表示 TCN 在 t 时刻提取到的时序特征, d_h 是定义的向量维度。需要指出的是，将多只股票的时序数据作为网络的多个通道输入。每个通道都可以使用相同的 TCN 模块处理，这样每个 TCN 模块就能够同时提取每只股票的时序特征。虽然使用相同的网络结构和网络参数提取多只股票的数据可以更好地学习通用的时序特征，提高模型的泛化能力，但是这种共享的结构也忽略了不同股票之间的相互关系以及独特特性。

5.2 分类特征提取

为了解决 TCN 结构不能对股票之间的相互关系以及独特性进行建模，我们使用 transformer 基于股票的分类信息对每只股票的特性进行个性化特征提取。首先我们从 wind 数据库中下次本次实验所涉及股票的 Wind 一级行业、Wind 二级行业、公司属性等特征。我们使用 Transformer 对 Wind 一级行业、Wind 二级行业、公司

属性特征进行特征提取。

$$H_t^{transformer} = Transformer(\mathcal{E}) \quad (6)$$

其中, \mathcal{E} 表示所有股票的分类特征 $H_t^{transformer} \in \mathbb{R}^{m \times 3 \times d_h}$ 表示经过 transformer 提取后的每只股票分类特征。经过上述两个特征提取后, 我们可以将其拼接起来的到股票完整的特征信息, $H_t = [H_t^{tcn}, H_t^{transformer}] \in \mathbb{R}^{m \times 4 \times d_h}$ 。最后, 使用一个全连接层根据这个既包含时序特征又包含分类特征的向量来计算最后的预测结果。

$$out_t = MLP(H_t) \quad (7)$$

其中 $out_t \in \mathbb{R}^{m \times 2}$ 为 t 时刻模型的预测结果, 需要预测下分钟股票平均成交价以及预测未来十分钟内股票平均成交价。

5.3 损失函数

通过上述构建的模型, 我们根据股票一段时间内的价格数据 \mathcal{X}_t 和分类信息 \mathcal{E} 我们通过 TTR 可以得到目前的预测结果 out_t , 从历史数据中可以计算得到真实值 y_t 。MSE (均方误差) 损失函数是用于评估回归问题中模型预测与实际观测值之间差异的一种常用损失函数。它计算预测值与真实值之间差异的平方, 然后取这些平方的平均值。计算公式如下:

$$MSE = \frac{1}{n} \sum_{t=1}^n (out_t - y_t)^2 \quad (8)$$

6 策略细节

我们的策略主要在 `quote_impl.py` 文件的 `onTick()` 成员函数中实现。

首先, 我们维护了三个关键值为股票代码的字典, 分别用于存放每只股票的上次交易行为和价格。在每天开市时, 我们对上述字典数据初始化, 同时清空 `self.kline_data` 中缓存的股票数据。仅当 `self.kline_data` 中数据达到 30 条 (模型的输入规模, 可调整大小) 时, 我们加载模型, 提取**开盘价, 收盘价, 最高价, 最低价, 加权成交价, 成交金额, 成交量**这 7 个核心特征并完成归一化、预处理, 然后预测下分钟平均交易价和未来十分钟平均交易价。

6.1 短视策略

基于上述数据，我们最终采用如下策略判断发送交易信号的时机：

1. 如果上次交易为买入；预测下分钟股票平均成交价卖出上次买入价格，发送卖出信号。
2. 如果上次交易为卖出；预测下分钟股票平均成交价低于上次卖出价格，发送买入信号。

我们将此策略命名为短视策略，因为此策略在预估收益非负的第一个时间点就进行了买卖操作。

6.2 远见策略

由于比赛中评分公式 $score = pnl * win_rate * \log(trades)$ ，我们需要对买入和卖出的时机进行进一步判断，从而提高单次买卖获利的大小。我们对短视策略作如下改进：

1. 如果上次交易为买入；预测下分钟股票平均成交价超过预测未来十分钟内股票平均成交价，差值达到阈值；并且预测下分钟股票平均成交价高于上次买入价格（若有），发送卖出信号。
2. 如果上次交易为卖出，预测未来十分钟内股票平均成交价超过预测下分钟股票平均成交价，差值达到阈值，并且预测下分钟股票平均成交价低于上次卖出价格（若有），发送买入信号。

我们将此策略命名为远见策略，因为此策略会一直等待到预估的最高点/最低点出现后才会进行卖/买操作。

7 参数优化和模型选择

7.1 参数优化

方法的参数优化过程可以分为神经网络参数优化和超参数优化。需要先确定一组超参数进行网络参数优化，根据结果在验证集上进行超参数优化。

1. 对于 TTR 的网络参数而言，计算损失函数关于参数的梯度，并以负梯度方向更新参数，逐渐降低损失值。这里采用的优化器为 AdamW [8]。AdamW 是对 Adam 优化算法的一种改进，主要通过权重衰减（Weight Decay）的方式来解决 Adam 在某些情况下可能导致权重更新过于激进的问题，具有更快的收敛速度。
2. 对于训练模型所需要的超参数，包括学习率、网络层数、向量维度等，因此需要选择一组合适的超参数训练网络模型并构建策略。因此根据官方提供的数据历史数据我们对数据进行了按照 8 : 2 划分为训练集、验证集。对于一些基本的网络模型参数如 TCN 和 Transformer 的基本参数我们参考原文中最佳的超参数进行设定。而对于本次实验中特定的参数如学习率、dropout 的值设定时，我们采用网格搜索的方式来调整不同的超参数，在验证集上找到最佳的组合以提高模型性能。

7.2 避免过拟合

在本文中的策略主要从两个方面来解决过拟合的问题，1) 从模型层面，使用 dropout 来解决神经网络预测过拟合问题。2) 从策略角度，通过设置交易阈值来缓解预测不准的过拟合问题。

1. **Dropout:** Dropout 是一种在神经网络中用于防止过拟合的正则化技术。过拟合是指模型在训练集上表现很好，但在未见过的数据上表现较差的现象。过拟合通常是由于模型过于复杂，学习了训练集中的噪声或样本特定的特征。Dropout 降低了每个神经元的影响，使得模型更容易适应训练集的真实特征，而不是过多关注噪声。这有助于减少过拟合风险。Dropout 强制网络在训练过程中随机地舍弃一些神经元的输出，使得模型不会过分依赖于某些特定神经元。

这有助于减少神经元之间的复杂关系，防止网络过分记忆训练集的噪声或特定样本的特征，从而提高泛化能力。

2. **交易阈值**：如前文第四节所述，只有预测下分钟股票平均成交价和预测未来十分钟内股票平均成交价大于一个阈值时才会发出交易信号，这样可以避免由于模型在预测的过程中受噪声的影响从而出现较小的误差导致出现与预期不符的交易，提高策略的稳健性。由于训练时市场和实际交易时的市场环境可能差别较大，导致预测结果出现过拟合的结果。通过调整阈值，可以使策略更好地适应不同的市场条件。例如，当市场波动性较高时，可以适当提高阈值，减少过多的交易。相反，当市场趋于平静时，可以降低阈值以更及时地响应变化。

8 性能评估

在第6节中，我们介绍了如下的两种交易策略：

短视策略 只比较预测下分钟平均交易价格和上次交易价格做出预测。

远见策略 比较预测下分钟平均交易价格和上次交易价格做出预测，同时考虑下分钟预测价格与未来 10 分钟内平均预测价格的差值大小。

在本次比赛中，为了最大化 *score*，我们选择远见策略；然而，由于真实市场交易中收益与交易次数成线性关系，短视策略同样具有应用价值。下面我们将对这两种策略进行性能评估与对比。

8.1 总收益

对于每个交易日，我们在回测中通过计算所有交易的收益率之和来刻画当日的总收益率。在回测中，我们测试得到上述两种量化策略在 2023.11.20~2023.11.30 内 9 个交易日中的每日总收益率（包括强制平仓交易），如Table 1和Table 2所示：

实验结果表明：

- 短视策略的单次 pnl 较低，交易频率较高，因而受收盘平仓的随机性影响很小，

Table 1: 短视策略在给定交易日期下的 Score 和总收益表现。

交易日期	score	总收益
2023.11.20	23.811	6.038
2023.11.21	16.436	2.715
2023.11.22	21.741	5.614
2023.11.23	25.863	6.774
2023.11.24	18.498	4.836
2023.11.27	21.654	5.716
2023.11.28	21.918	5.791
2023.11.29	20.599	5.505
2023.11.30	24.732	6.163

score 一般较低，但方差较小。同时总收益率更高，且具有很高稳定性。适用于真实市场交易。

- 远见策略的单次 pnl 较高，交易次数较少，因而受收盘平仓的随机性影响很大。score 一般较高，但方差很大，有时甚至为负值。总收益率较低，且不稳定。适用于本次比赛。

8.2 夏普比率

夏普比例是量化策略评估中的常用指标之一，在权衡交易策略的收益与风险方面有很大的价值。夏普比率以无风险收益为基准计算额外收益与风险的比值。夏普比率小于零的说明策略的平均收益率为负；夏普比率的绝对值大小则表明该策略风险与收益的关系。夏普比率小于 1 意味着风险小于收益。一般而言，一个优秀的量化策略的夏普率高于 1。

在回测中，我们测试得到上述量化策略在 2023.11.20~2023.11.30 内 9 个交易日中的每次交易（包括强制平仓交易）的日收益率 R_p ，并根据公式

$$SharpeRatio = \frac{E(R_p) - R_f}{\sigma_p}$$

Table 2: 远见策略在给定交易日期下的 Score 和总收益表现。

交易日期	score	总收益
2023.11.20	126.586	0.459
2023.11.21	-25.307	0.500
2023.11.22	-45.537	-0.646
2023.11.23	73.677	0.802
2023.11.24	-30.289	0.153
2023.11.27	24.940	0.892
2023.11.28	25.212	-0.406
2023.11.29	-25.639	0.320
2023.11.30	-6.890	0.802

Table 3: 短视策略与原见策略的收益率、夏普比率实验结果。

策略	收益率均值	收益率标准差	夏普比率
短视策略	5.524	0.905	6.101
远见策略	0.248	0.547	0.454

计算得到在这 9 个交易日内的夏普比率。实验结果如Table 3所示，该结果表面：

- 短视策略的总收益率，收益率均值和夏普比率均优于远见策略。
- 短视策略的夏普比率高达超达 6.1，表明其收益大于风险，这显示了短视策略在实际交易中具有带来稳定收益的能力。

9 分析和讨论

由上述实验结果可知，模型在预测准确率方面具有较好的效果，能够在快速变化的市场中较为准确地预测出模型下一步的走势。但由于只依靠预测下提供相对准确的下分钟股票平均成交价预测和未来十分钟内股票平均成交价进行判断卖出和买入，较

为简单。可以从回测结果看出，在某些情况下实验结果仍不如人意。此外，策略构建部分对于比赛中每日结束后的强制平仓并没有采取有效的处理手段，这是导致策略在特殊市场环境中收益较差的另一种原因。尽管在实际市场运行中，该操作并不是强制的。

由前述可知，在不同市场环境下，可以通过设置不同的交易阈值达到快速适应，稳定收益的目的。然而，在本次比赛中由于时间较短、经验较少，尚未完全发挥其能力。所以本次实验构建的策略在不同风格的市场下仍有波动，未来可以对市场分析调整市场阈值来缩小投资策略的收益波动，实现持续稳定的收益。

10 结论

本文介绍了我们在博金杯量化模型大赛中采用的策略，该策略由深度学习的网络模型和基于模型预测结果的策略构建组成。在深度学习模型方面，我们提出了 TTR 模型，该模型提取了多维度的股票特征信息：1) 使用 TCN 提取股票的时序特征信息；2) 使用 Transformer 来提取股票所述类别的特征信息。通过提取到的混合特征信息，我们使用神经网络来预测股票在下一分钟的涨跌百分比，和股票在未来十分钟的平均涨跌。基于这些预测结果，我们综合考虑当前持仓股票成交价、下一分钟股票平均成交价预测值、未来十分钟股票平均成交价预测值三者的大小关系，设计了一套股票交易策略，旨在优化我们的收益。我们的策略最终在比赛中取得了第六名的成绩。本次实验结果体现了提出的策略仍然存在不足，可以从一下几个方面进行改进：1) 结合大模型来提取多模态的信息。尽管我们在赛事方提供的基本价格数据基础上挖掘了分类信息，在实际应用中可以进一步使用更加丰富市场信息包括新闻信息、财报信息等。结合最近展现出强大能力的大模型，可以将神经模型的预测结果进一步提高，从而提高策略的盈利能力。2) 提升策略的稳健性。尽管通过控制交易阈值可以实现风险控制，快速适应不同风格的市场。但如何判断市场风格并且设置最优的阈值下一步仍然值得探索。

References

- [1] Anastasia Borovykh, Sander Bohte, and Cornelis W Oosterlee. Dilated convolutional neural networks for time series forecasting. Journal of Computational Finance, Forthcoming, 2018.
- [2] Jou-Fan Chen, Wei-Lun Chen, Chun-Ping Huang, Szu-Hao Huang, and An-Pin Chen. Financial time-series data analysis using deep convolutional neural networks. In 2016 7th International conference on cloud computing and big data (CCBD), pages 87–92. IEEE, 2016.
- [3] Qianggang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Jian Guo. Hierarchical multi-scale gaussian transformer for stock movement prediction. In IJCAI, pages 4640–4646, 2020.
- [4] Matthew F Dixon, Igor Halperin, and Paul Bilokon. Machine learning in finance, volume 1170. Springer, 2020.
- [5] Linhao Dong, Shuang Xu, and Bo Xu. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP), pages 5884–5888. IEEE, 2018.
- [6] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. IEEE transactions on pattern analysis and machine intelligence, 45(1):87–110, 2022.
- [7] Colin Lea, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks: A unified approach to action segmentation. In Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14, pages 47–54. Springer, 2016.
- [8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

- [9] Kriti Pawar, Raj Srujan Jalem, and Vivek Tiwari. Stock market price prediction using lstm rnn. In Emerging Trends in Expert Applications and Security: Proceedings of ICETEAS 2018, pages 493–503. Springer, 2019.
- [10] Akhter Mohiuddin Rather, Arun Agarwal, and VN Sastry. Recurrent neural network and a hybrid model for prediction of stock returns. Expert Systems with Applications, 42(6):3234–3241, 2015.
- [11] Francesco Rundo, Francesca Trenta, Agatino Luigi di Stallo, and Sebastiano Battiato. Machine learning for quantitative finance applications: A survey. Applied Sciences, 9(24):5574, 2019.
- [12] Sreelekshmy Selvin, R Vinayakumar, EA Gopalakrishnan, Vijay Krishna Menon, and KP Soman. Stock price prediction using lstm, rnn and cnn-sliding window model. In 2017 international conference on advances in computing, communications and informatics (icacci), pages 1643–1647. IEEE, 2017.
- [13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.
- [14] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In Proceedings of the AAAI conference on artificial intelligence, volume 37, pages 11121–11128, 2023.
- [15] Qiuyue Zhang, Chao Qin, Yunfeng Zhang, Fangxun Bao, Caiming Zhang, and Peide Liu. Transformer-based attention network for stock movement prediction. Expert Systems with Applications, 202:117239, 2022.

附录

我们在本次比赛中用到的所有代码见 <https://github.com/Rachmanino/2023CCF-Quant>