arXiv:2601.00197v1 [cs.CE] 1 Jan 2026

StockBot 2.0：原始 LSTM 在股票价格预测方面优于基于 Transformer 的预测

Shaswat Mohanty[a,*]

[a]机械工程系，斯坦福大学，*CA 94305-4040*，美国

摘要

由于时间依赖关系复杂且往往存在潜在依赖关系、非线性动力学以及高度波动性，准确预测金融市场长期以来一直是一项挑战。在我们先前的循环神经网络框架基础上，我们提出了增强版 StockBot 架构，在统一的实验设置中系统评估现代基于注意力机制、卷积和循环的时间序列预测模型。尽管基于注意力机制和受 Transformer 启发的模型提供了更高的建模灵活性，但大量实证评估表明，在一组通用默认超参数下训练时，经过精心构建的原始 LSTM 始终能够实现更高的预测准确度，并作出更稳定的买卖决策。这些结果凸显了循环序列模型在金融时间序列预测中的鲁棒性和数据效率，尤其是在缺少大量超参数调优或单日间隔离散化后可用数据不足的情况下。此外，这些结果也强调了架构归纳偏置在数据有限的市场预测任务中的重要性。

关键词： 股票预测、StockBot、Transformer、时间序列预测、注意力机制

1. 引言

由于金融时间序列数据本身具有非线性、噪声和结构性波动等特点，预测股价变动仍是量化金融领域的一项核心挑战。传统统计模型和计量经济模型往往依赖线性假设，难以捕捉支配市场行为的动态依赖关系 [1, 2]。深度学习方法，尤其是循环神经网络（RNN）及其变体（如长短期记忆网络（LSTM）），在建模时间依赖关系和非线性相关性方面展现出更优的性能 [3–5]。 [6]通过消融研究展示了堆叠多个长短期记忆网络的优势，证明其在预测股票价格方面表现出更优的预测性能，尤其是在

*通讯作者电子邮箱地址： shaswatm@stanford.edu (Shaswat Mohanty)

能源行业。文献中已有大量此类研究，并展现出预测股价变动的潜力，但由于滞后效应 [7] 以及短期内波动率指数（VIX）的剧烈波动 [8]。

序列建模领域的后续进展引入了 Transformer [9]和时间卷积网络（TCN） [10], 等替代架构，使长程依赖关系能够得到更高效、更具表现力的表征。Transformer 通过自注意力机制灵活地建模不同时刻之间的上下文相关性，而时间卷积网络则利用扩张因果卷积，以更少的参数实现相近的感受野。这些架构在多种时间序列应用中展现出强大的泛化能力，促使研究者将其应用于金融预测 [11–13]。

在本研究中，我们扩展了此前提出的 StockBot 框架，将多种时间预测模型纳入统一的评估设置，包括循环模型、受 Transformer 启发的注意力模型、卷积模型以及注意力—循环混合架构。这一设计能够在多个市场行业中，对模型的表达能力、泛化行为和下游交易可解释性进行受控比较。通过在相同的训练流程和默认超参数设置下评估所有架构，我们的分析重点考察架构归纳偏置对预测性能和决策稳定性的影响，并通过简单的买入／卖出交易策略进行评估。

本文其余部分安排如下。第2 节介绍数据集构建、归一化过程和模型架构。第3 节展示不同模型系列的比较结果，并评估 StockBot 的交易表现。最后，第4 节总结主要发现，并概述未来工作的方向。

2. 数据与方法

2.1. 数据准备

数据集和预处理步骤遵循我们先前工作 [6]中确立的流程。历史股票价格数据取自 Yahoo! Finance [14], ，涵盖在纽约证券交易所和纳斯达克上市的股票。预测的主要变量为调整后收盘价。

滑动窗口方法用于构建监督学习序列。对于给定的`past_history` k 天，模型会预测随后 `forward_look`天的股价。虽然大多数实验使用 `forward_look`$= 1$，但该框架可推广至多步预测。数据涵盖 2010 - 2020 年，并按时间顺序划分为 80% 的训练集和 20% 的测试集。为防止数据泄漏，所有序列均根据训练集的统计数据进行 Z 分数标准化。此数据选择与预处理方式与 [6] 中先前的评估保持一致。

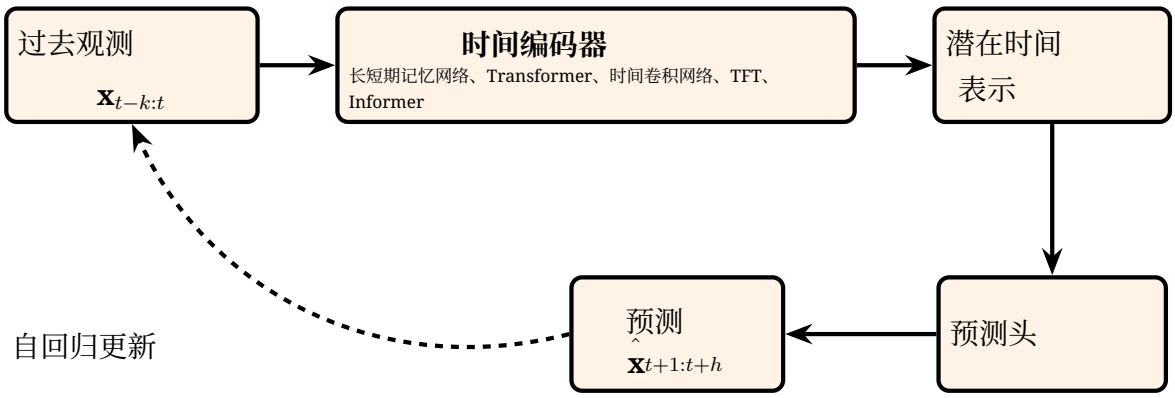

图1：本文所研究的所有模型共用的通用时间预测框架。过去观测被编码为潜在时间表示，再映射为未来预测。虚线回路表示可选的自回归预测，用于多步预测。

2.2. 模型架构

我们在统一框架下评估一系列时序深度学习模型，每个模型都实现为基类 `LSTM_Model` 的子类。这使不同类型的模型能够以一致的方式进行数据处理、训练和评估。在评估时，所有模型都可以执行两种预测：（i）自回归预测，即递归地将最新预测结果作为下一步的输入。这使我们能够评估模型在多个预测时域内的稳定性和漂移行为；（ii）教师强制预测，即在每一步都提供真实值，从而在不存在累积误差的情况下，为预测准确度提供一个上限。

2.2.1. 长短期记忆网络与注意力增强型长短期记忆网络

基准模型采用堆叠式长短期记忆网络（LSTM），通过门控循环单元捕获时间依赖关系。在时间步 t，长短期记忆网络层的前向传播由公式（1）–（6）给出：

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f), \tag{1}$$
$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i), \tag{2}$$
$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o), \tag{3}$$
$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c), \tag{4}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \tag{5}$$
$$h_t = o_t \odot \sigma_h(c_t), \tag{6}$$

其中，x_t 是长短期记忆网络单元的输入向量；f_t、i_t和 o_t 分别表示遗忘门、输入门和输出门的激活值；c_t 和 h_t 分别是细胞状态和隐藏状态。函数 $\sigma_g(\cdot)$和 $\sigma_c(\cdot) = \sigma_h(\cdot)$分别对应Sigmoid函数和双曲正切函数激活。矩阵 W、U和偏置 b在训练过程中学习得到。

注意力增强变体在长短期记忆网络输出上引入了可学习的注意力层，从而能够对相关时间步进行自适应加权。给定序列

由堆叠式 LSTM 层生成的隐藏状态 $\{h_1, h_2, \ldots, h_T\}$，应用 Bahdanau 风格注意力机制来计算上下文向量。查询向量定义为隐藏状态在时间维度上的平均值，

$$q = \frac{1}{T} \sum_{t=1}^{T} h_t. \tag{7}$$

每个时间步的注意力分数计算如下

$$e_t = v^\top \tanh(W_1 h_t + W_2 q)\,, \tag{8}$$

其中，W_1 和 W_2 是可学习的投影矩阵，v 是可学习的注意力向量。归一化后的注意力权重通过以下方式获得

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T} \exp(e_k)}. \tag{9}$$

由此得到的上下文向量是隐藏状态的加权和，

$$c = \sum_{t=1}^{T} \alpha_t h_t. \tag{10}$$

最后，通过全连接层将上下文向量映射到预测范围，

$$\hat{y} = W_o c + b_o. \tag{11}$$

2.2.2. 时间卷积网络（*TCN*）

时间卷积网络（TCN）使用因果一维卷积对序列依赖关系建模，确保时间 t 的预测仅依赖于来自时间步 $\leq t$的输入。给定输入序列 $\mathbf{X} = \{x_1, x_2, \ldots, x_T\}$ 其中 $x_t \in \mathbb{R}^{d_{\text{in}}}$，第一个因果卷积层计算

$$\mathbf{H}_t^{(1)} = \phi\left(\sum_{k=0}^{K-1} W_k^{(1)}\, x_{t-k} + b^{(1)} \right), \tag{12}$$

其中，K 是核大小，$\phi(\cdot)$表示 ReLU 激活函数，$W_k^{(1)}$ 和 $b^{(1)}$ 是可学习的卷积权重和偏置。第二个因果卷积层会进一步转换该序列：

$$\mathbf{H}_t^{(2)} = \phi\left(\sum_{k=0}^{K-1} W_k^{(2)}\, \mathbf{H}_{t-k}^{(1)} + b^{(2)} \right). \tag{13}$$

为了获得固定维度的时间序列表示，沿时间维度应用全局平均池化：

$$\bar{\mathbf{h}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{H}_t^{(2)}. \tag{14}$$

最后，通过全连接层将池化表示映射为输出预测：

$$\hat{\mathbf{y}} = \bar{\mathbf{h}} W_o + b_o, \tag{15}$$

其中 $\hat{\mathbf{y}} \in \mathbb{R}^{\texttt{forward_look}}$ 表示未来 `forward_look` 个时间步的预测值。

2.2.3. Informer 变体

这些 Transformer 扩展降低了计算成本，并增强了时序表征。Informer 利用 probSparse 注意力进行长时域预测；Informer 风格模型以输入时间序列 $\mathbf{X} = \{x_1, x_2, \ldots, x_T\}$为输入，其中$x_t \in \mathbb{R}^{d_{\text{in}}}$ 和 T 表示历史窗口的长度。输入首先被嵌入到维度为 d 的潜在空间中：

$$\mathbf{H}^{(0)} = \mathbf{X} W_e + b_e, \tag{16}$$

其中 $W_e \in \mathbb{R}^{d_{\text{in}} \times d}$。嵌入后的序列由 L 个 Transformer 编码器块组成的堆栈进行处理。在编码器层 ℓ，多头自注意力的计算方式为

$$\mathbf{Q}_h^{(\ell)} = \mathbf{H}^{(\ell-1)} W_h^Q, \tag{17}$$

$$\mathbf{K}_h^{(\ell)} = \mathbf{H}^{(\ell-1)} W_h^K, \tag{18}$$

$$\mathbf{V}_h^{(\ell)} = \mathbf{H}^{(\ell-1)} W_h^V, \tag{19}$$

对于注意力头 h。缩放点积注意力定义为

$$\text{Attn}_h^{(\ell)} = \text{softmax}\left(\frac{\mathbf{Q}_h^{(\ell)} \mathbf{K}_h^{(\ell)\top}}{\sqrt{d_k}} \right) \mathbf{V}_h^{(\ell)}. \tag{20}$$

所有注意力头的输出经过拼接后，再进行线性投影：

$$\text{MHA}^{(\ell)} = \text{Concat}\big(\text{Attn}_1^{(\ell)}, \ldots, \text{Attn}_H^{(\ell)}\big) W^O. \tag{21}$$

应用残差连接和层归一化：

$$\tilde{\mathbf{H}}^{(\ell)} = \text{LayerNorm}\Big(\mathbf{H}^{(\ell-1)} + \text{MHA}^{(\ell)} \Big). \tag{22}$$

前馈子层定义为

$$\text{FFN}(\tilde{\mathbf{H}}^{(\ell)}) = \phi\Big(\tilde{\mathbf{H}}^{(\ell)} W_1 + b_1 \Big) W_2 + b_2, \tag{23}$$

$$\mathbf{H}^{(\ell)} = \text{LayerNorm}\Big(\tilde{\mathbf{H}}^{(\ell)} + \text{FFN}(\tilde{\mathbf{H}}^{(\ell)}) \Big), \tag{24}$$

其中， $\phi(\cdot)$ 表示 GELU 激活函数。经过最后一个编码器层后，使用全局平均池化聚合时间信息：

$$\bar{\mathbf{h}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{H}_t^{(L)}. \tag{25}$$

接下来 `forward_look` 个时间步的最终预测结果通过线性投影得到：

$$\hat{\mathbf{y}} = \bar{\mathbf{h}} W_o + b_o. \tag{26}$$

Autoformer 引入序列分解模块以进行趋势-季节性建模；FEDformer 则采用频域注意力，以提高长序列处理效率。

2.2.4. 时序融合 *Transformer*（*TFT*）

TFT 将循环编码器、多头注意力和门控机制相结合，以处理静态协变量和动态时序输入。它尤其适用于多变量预测，并能对多个特征组上的注意力进行解释。时序融合 Transformer（TFT）模型结合了循环编码、自注意力和门控残差连接，以建模短期时间动态和长程依赖。给定输入序列 $\mathbf{X} = \{x_1, x_2, \ldots, x_T\}$with $x_t \in \mathbb{R}^{d_{\text{in}}}$, 模型首先应用堆叠式长短期记忆网络编码器：

$$\mathbf{H}^{(1)} = \mathrm{LSTM}_1(\mathbf{X}), \tag{27}$$

$$\mathbf{H}^{(2)} = \mathrm{LSTM}_2(\mathbf{H}^{(1)}), \tag{28}$$

其中，$\mathbf{H}^{(2)} \in \mathbb{R}^{T \times d}$ and d 表示隐藏维度。

为捕捉长程时间依赖关系，将多头自注意力应用于长短期记忆网络输出。对于注意力头 h，查询、键和值矩阵定义如下：

$$\mathbf{Q}_h = \mathbf{H}^{(2)} W_h^Q, \tag{29}$$

$$\mathbf{K}_h = \mathbf{H}^{(2)} W_h^K, \tag{30}$$

$$\mathbf{V}_h = \mathbf{H}^{(2)} W_h^V. \tag{31}$$

缩放点积注意力的计算方式如下：

$$\mathrm{Attn}_h = \mathrm{softmax}\left(\frac{\mathbf{Q}_h \mathbf{K}_h^\top}{\sqrt{d_k}}\right) \mathbf{V}_h. \tag{32}$$

将所有注意力头的输出拼接并进行归一化：

$$\mathbf{A} = \mathrm{LayerNorm}(\mathrm{Concat}(\mathrm{Attn}_1, \ldots, \mathrm{Attn}_H)). \tag{33}$$

随后应用门控机制来控制来自注意力模块的信息流：

$$\mathbf{G} = \sigma(\mathbf{A} W_g + b_g), \tag{34}$$

$$\mathbf{Z} = \mathbf{G} \odot \mathbf{A}, \tag{35}$$

其中， $\sigma(\cdot)$表示Sigmoid激活函数， $\odot$ 表示逐元素乘法。门控注意力的输出通过残差连接与长短期记忆网络表示相结合：

$$\mathbf{H} = \mathbf{H}^{(2)} + \mathbf{Z}. \tag{36}$$

使用全局平均池化聚合时间信息：

$$\bar{\mathbf{h}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{H}_t. \tag{37}$$

最后，通过线性输出层获得接下来 `forward_look` 个时间步的预测结果：

$$\hat{\mathbf{y}} = \bar{\mathbf{h}} W_o + b_o. \tag{38}$$

2.3. 训练和超参数

每个模型均使用Adam优化器进行训练，初始学习率为10^{-3}。基于验证损失应用早停，以防止过拟合。

超参数	值	超参数	值
`past_history`	60	批量大小	64
训练轮数	500	Dropout	0.1
隐藏维度	64	前馈维度	128
Transformer 头	4	训练测试集划分	80%/20%

表 1：所有模型使用的默认超参数。

2.4. 股票预测机器人

在本文中，我们将继续评估预测模型的效能：按照前文 [6] 所述，通过评估一个机器人每天在收盘时执行买卖操作以最大化收益的表现来进行。该机器人根据我们对股票价格的预测做出决策。这些决策通过解析方式强制执行，而非单独学习得出。机器人通过以下两步流程做出决策：首先记录由 δ_i 定义的变化 $\delta_i = \operatorname{sign}(c_{i+1} - c_i)$，其中 c_i 表示第 i^{th} 天的股票价格。然后，我们通过跟踪 $\Delta_i = \delta_{i+1} - \delta_i$，观察 δ_i 的变化。$\Delta = -2$ 表示下跌的结束，此时我们买入；而 $\Delta_i = 2$ 表示下跌的开始，此时我们卖出。算法总结如下：

算法 1 : StockBot 决策算法

获取预测轨迹，$\mathbf{c}$ （ `forward_look` 天的股票预测；计算变化性质： $\delta_i = \operatorname{sign}(c_{i+1} - c_i)$；计算预测的曲率： $\Delta_i = \delta_{i+1} - \delta_i$；作出决策：决策 $= \begin{cases} \Delta_i = 2 \rightarrow & \text{卖出（表示局部极大值）} \\ \Delta_i = 0 \rightarrow & \text{卖出（表示无变化）} \\ \Delta_i = -2 \rightarrow & \text{卖出（表示局部极小值）}。 \end{cases}$

3. 结果

我们在统一的实验框架内评估所有预测架构，同时比较预测准确性和由 StockBot 评估的下游交易表现。所有模型均在相同的默认超参数下进行评估，不针对特定架构进行调优，以确定在缺乏大量超参数优化资源的情况下，哪些模型能够开箱即用地实现合理表现。

3.1. 单日提前预测

我们首先考虑提前预测一个交易日的任务，在此过程中，模型会递归运行以生成更长的轨迹。我们考察了两种评估模式：自回归预测，即将模型自身的预测结果反馈为输入；以及教师强制，即在推理过程中每一步都提供真实价格。

图 2 和 3 分别总结了股票代码 AAPL 和 MSFT 的结果。在每张图中，左上图展示自回归价格预测，右上图展示教师强制预测。左下图展示交易决策由自回归预测驱动时投资组合价值的演变，右下图则展示教师强制下相应的投资组合轨迹。

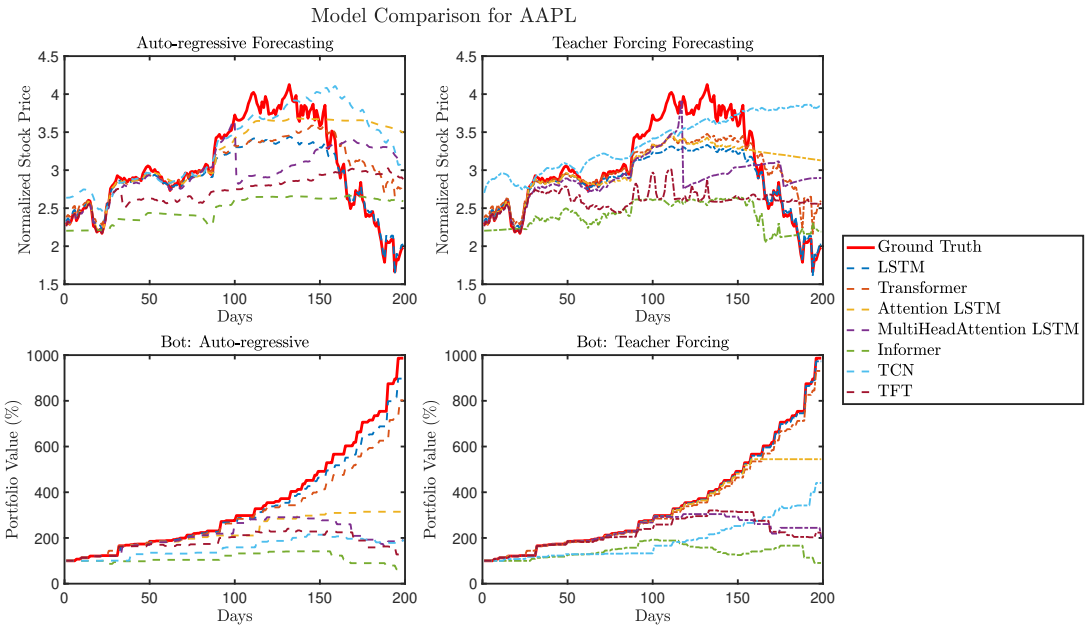

图 2：AAPL 的单日提前预测结果。左上：自回归价格预测。右上：教师强制价格预测。左下：由自回归预测驱动的投资组合价值。右下：由教师强制预测驱动的投资组合价值。

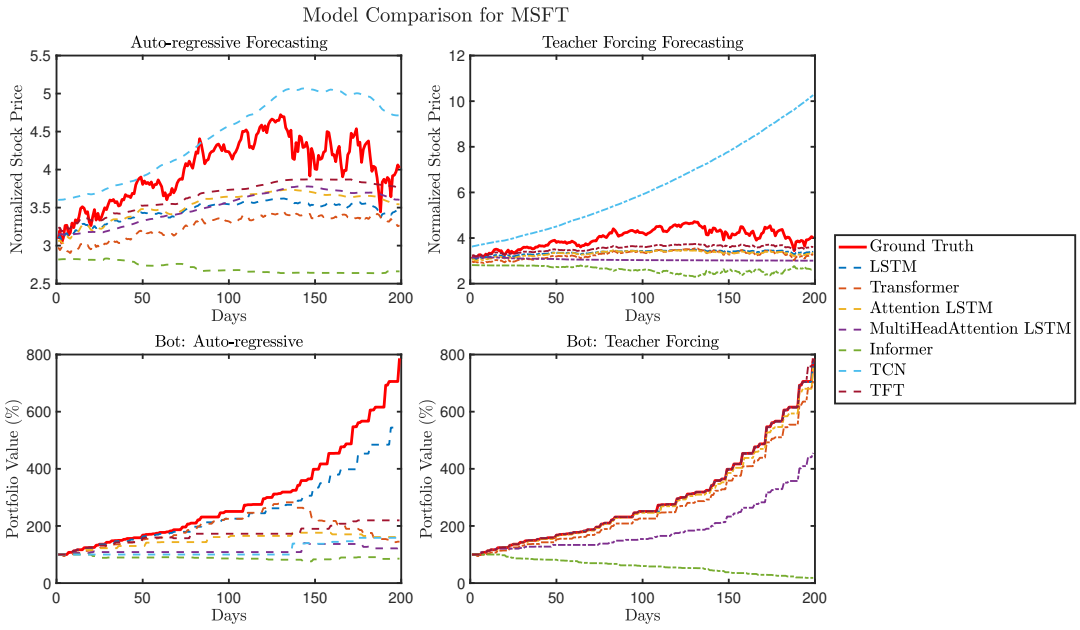

图 3：MSFT 的单日提前预测结果。左上：自回归价格预测。右上：教师强制价格预测。左下：由自回归预测驱动的投资组合价值。右下：由教师强制预测驱动的投资组合价值。

对于这两只股票，所有模型都能捕捉短期趋势，或者至少在教师强制下捕捉方向性趋势；然而，在自回归评估中，各模型之间出现了显著差异。原始 LSTM 始终表现出最稳定的长周期行为，相较于基于注意力机制和卷积架构的模型，其误差累积更慢、漂移更小。这些稳定性优势直接转化为更平滑的投资组合轨迹和更少的虚假交易。对于 StockBot 而言，预测在数值上准确并非先决条件，因为它只取决于股票价格的方向性变动。

更具表现力的架构，例如 Transformer、TFT 和 Informer 模型，通常能取得更低的训练损失，但在递归部署时，对累积预测误差更加敏感。若不进行超参数调优，这种敏感性会导致交易表现下降，尽管短期预测准确度相当。

模型	AAPL		MSFT	
	自动 RMSE	TF RMSE	自动 RMSE	TF RMSE
LSTM	0.2556	0.3061	0.6129	0.7040
Transformer	0.3713	0.3094	0.7952	0.7737
注意力长短期记忆网络	0.5885	0.4732	0.5314	0.7422
多头注意力长短期记忆网络	0.6137	0.5164	0.5502	1.0478
Informer	0.7670	0.7749	1.3772	1.4819
TCN	0.5805	0.7188	0.5295	2.8410
TFT	0.6587	0.6955	0.4254	0.5254

表 2：AAPL 和 MSFT 在不同预测模式下的测试集均方根误差对比。各模型均报告自回归（Auto.）和教师强制（TF）的结果。

表 2 定量汇总了两种预测模式下的测试均方根误差（RMSE）值，证实长短期记忆网络在自回归和教师强制部署下的误差最低或接近最低。

3.2. 提前十天预测

接下来，我们评估一种更具挑战性的情形：模型需要同时预测未来十个交易日的价格向量。这种设定减少了递归反馈，但对模型捕捉中期时间结构的能力提出了更高要求。

图 4 和 5 分别展示了股票代码 AAPL 和 MSFT 对应的结果。与之前一样，左上和右上图分别展示自回归和教师强制价格预测，底部图则报告由这两种预测策略驱动的投资组合增长情况。

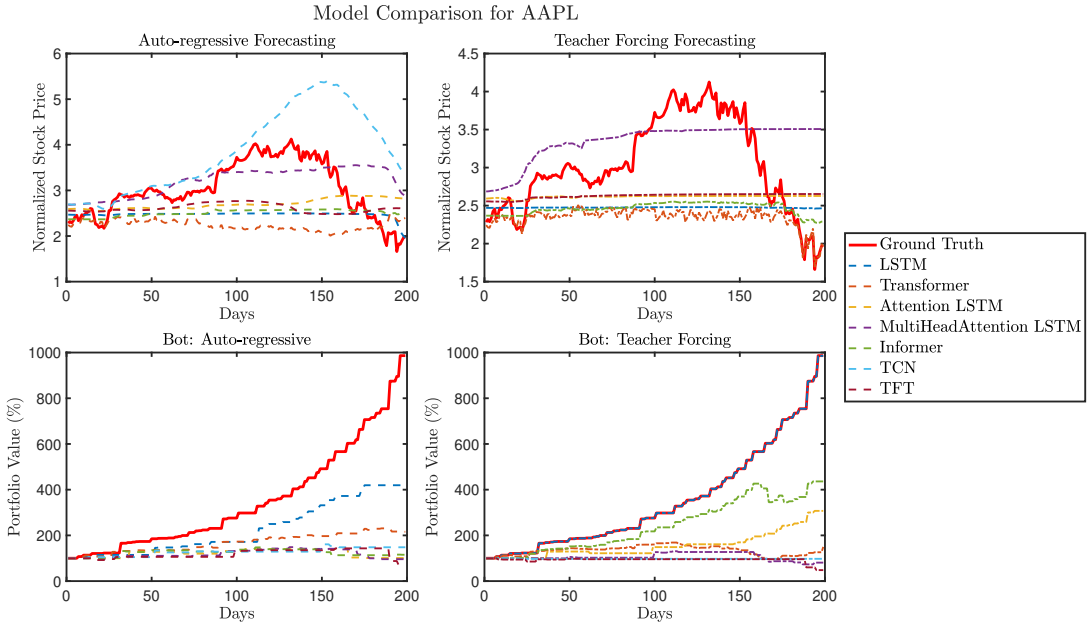

图 4：AAPL 提前十天预测结果。左上：自回归价格预测。右上：教师强制价格预测。左下：由自回归预测驱动的投资组合价值。右下：由教师强制预测驱动的投资组合价值。

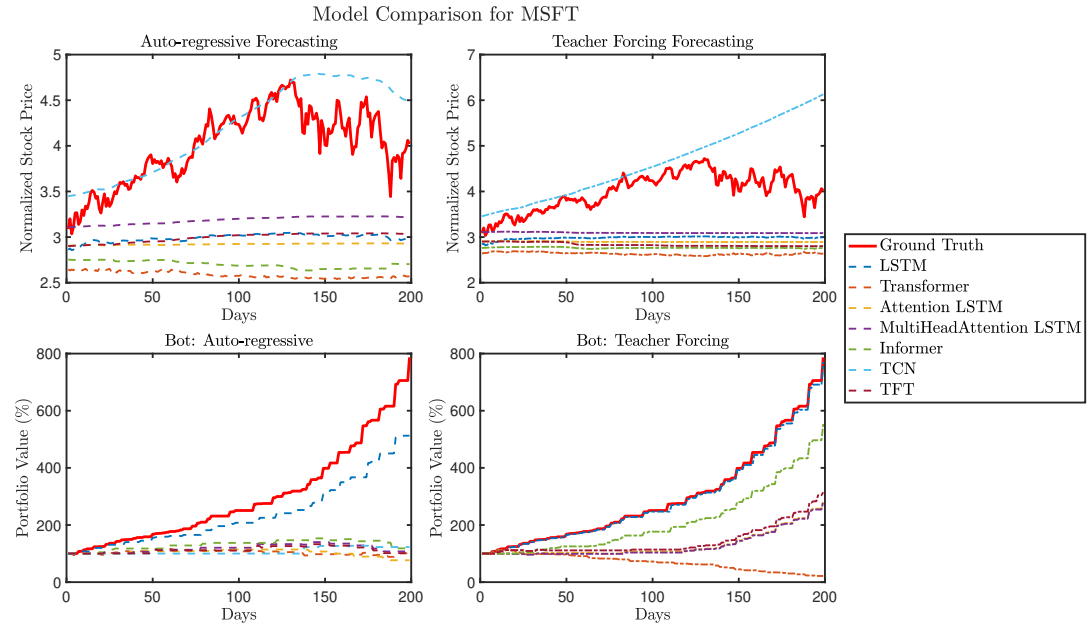

图 5：MSFT 提前十天预测结果。左上：自回归价格预测。右上：教师强制价格预测。左下：由自回归预测驱动的投资组合价值。右下：由教师强制预测驱动的投资组合价值。

模型	AAPL		MSFT	
	自动均方根误差	TF 均方根误差	自动均方根误差	TF 均方根误差
LSTM	0.8033	0.8190	1.0722	1.0845
Transformer	1.0405	0.8749	1.4791	1.4319
注意力长短期记忆网络	0.6997	0.7276	1.1506	1.1778
多头注意力长短期记忆网络	0.5624	0.6294	0.8965	0.9958
Informer	0.7676	0.7812	1.3725	1.3022
TCN	1.1186	150.3449	0.3543	0.8802
TFT	0.7160	0.7167	1.0699	1.2448

表3：AAPL 和 MSFT 在不同预测模式下的测试集均方根误差比较。报告了各模型的自回归（Auto.）和教师强制（TF）结果。

与提前一天预测相比，所有模型在更长预测期限下的不确定性都会增加。基于 Transformer 和注意力增强架构的模型在教师强制下生成的多日轨迹更加平滑，但使用默认超参数进行训练后，其自回归部署仍容易受到漂移影响。相比之下，长短期记忆网络保持了有竞争力的准确性，并展现出稳健的交易表现，其投资组合增长曲线与更复杂的模型相当或更优，如表3 所示；该表指出，在自回归和教师预测模式部署下，长短期记忆网络模型的均方根误差更低。

这些结果揭示了一项重要的实践启示：尽管基于注意力机制的架构具有更强的建模灵活性，但强大的循环归纳偏置和数据效率使原始 LSTM 尤其适用于数据有限场景下的金融时间序列预测。当评估侧重于递归式部署和下游决策时，架构简洁性和稳定性可能胜过纯粹的表达能力。

4. 结论

在本研究中，我们在统一的实验与决策框架下，对用于股价预测的循环式、基于注意力机制、卷积式及混合时序架构进行了系统评估。与根据大规模自然语言和序列建模基准得出的普遍预期相反，我们的结果表明，在采用一组通用默认超参数进行训练、且金融数据有限的情况下，精心构建的原始 LSTM 始终优于更复杂的注意力驱动模型。

在自回归和教师强制评估两种模式下，单步（提前一天）预测对所有模型类别而言都是最可靠的设置，能够带来更低的预测误差和更稳定的下游交易行为。尽管多步预测仍然可行，但对于基于 Transformer 和卷积的架构而言，误差累积以及对超参数选择的敏感性日益明显，尤其是在缺乏特定任务调优的情况下。这些发现凸显了

架构归纳偏置和优化稳定性在金融时间序列建模中的重要性；在这一领域，数据可用性和信噪比与大规模序列学习领域存在根本差异。

嵌入 StockBot 交易框架后，由长短期记忆网络驱动的预测相较于基于注意力机制的替代模型，能够带来更稳定的投资组合增长和更低的波动性。尽管 Transformer 类模型具有更强的表征灵活性和可解释性，但在数据受限且超参数配置统一的情况下，其优势并未稳定显现。这表明，仅凭模型复杂度并不足以保证其在实际金融预测应用中取得更优表现。

总体而言，我们的结果再次证实，循环序列模型对于市场预测和决策制定仍具有重要价值，尤其是在优先考虑鲁棒性、数据效率和部署简便性时。对于 StockBot 的具体应用，更审慎的探索方向是训练预测分类器而非回归器，因为买入／卖出决策完全基于方向变化。未来工作将重点关注有针对性的超参数优化、循环与注意力相结合的混合架构、通过模型集成进行不确定性量化，以及状态感知训练策略，以更好地评估在何种条件下，表达能力更强的时间模型能够较经典循环架构带来切实益处。

数据访问

本研究使用的所有数据集均可通过 Yahoo Finance 公开获取 [14] 和 Google Trends [15]。如有需要，可向作者索取有关计算过程和代码实现的更多信息。

致谢

作者感谢同事和导师提出的深入讨论与反馈，这些意见提升了本文的清晰度。

参考文献

[1] John J Murphy.《金融市场技术分析：交易方法与应用综合指南》. Penguin, 1999年。[2] Chih-Fong Tsai 和 Yu-Chieh Hsiao. 结合多种股票预测特征选择方法：并集、交集与多重交集方法。*Decision Support Systems*，50(1):258–269，2010年。[3] Luca Di Persio、Oleksandr Honchar 等。用于股票价格预测的人工神经网络架构：比较与应用。*International Journal of Circuits, Systems and Signal Processing*，10:403–413，2016年。

[4] Thomas Fischer 和 Christopher Krauss。使用长短期记忆网络进行金融市场预测的深度学习。《欧洲运筹学杂志》，270(2):654–669，2018年。[5] Sreelekshmy Selvin、R Vinayakumar、EA Gopalakrishnan、Vijay Krishna Menon 和 KP Soman。使用长短期记忆网络、循环神经网络和卷积神经网络滑动窗口模型进行股票价格预测。载于2017年国际计算、通信与信息学进展会议（ICACCI），第1643–1647页。IEEE，2017年。[6] Shaswat Mohanty、Anirudh Vijay 和 Nandagopan Gopakumar。StockBot：使用长短期记忆网络预测股票价格。arXiv预印本 arXiv:2207.06605，2022年。[7] Rufeng Chen。基于长短期记忆神经网络的股票预测挑战。载于经济管理与绿色发展国际会议，第490–499页。Springer，2023年。[8] CJ Finnegan、James F McCann 和 Salissou Moutari。少即是多：运用动态深度神经网络进行人工智能决策，实现短期股票指数预测。arXiv预印本 arXiv:2408.11740，2024年。[9]Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。注意力机制就是你所需要的一切。《神经信息处理系统进展》第30卷，2017年。[10] Colin Lea、Michael D Flynn、Rene Vidal、Austin Reiter 和 Gregory D Hager。用于动作分割与检测的时间卷积网络。载于IEEE计算机视觉与模式识别会议论文集，第156–165页，2017年。[11] Yujie Liu、Hongbin Dong、Xingmei Wang 和 Shuang Han。基于时间卷积网络的时间序列预测。载于2019年IEEE/ACIS第十八届计算机与信息科学国际会议（ICIS），第300–305页。IEEE，2019年。[12] Qiuyue Zhang、Chao Qin、Yunfeng Zhang、Fangxun Bao、Caiming Zhang 和 Peide Liu。用于股票走势预测的基于Transformer的注意力网络。《专家系统与应用》，202:117239，2022年。[13] Chaojie Wang、Yuanyuan Chen、Shuqi Zhang 和 Qiuhui Zhang。使用深度Transformer模型进行股票市场指数预测。《专家系统与应用》，208:118128，2022年。[14] Yahoo! Finance。 https://finance.yahoo.com/.[15]Google Trends Anchor Bank。https://github.com/epfl-dlab/GoogleTrendsAnchorBank。