iTransformer:Inverted Transformers Are Effective For Time Series Forecasting

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu
Shiyu Wang, Lintao Ma, Mingsheng Long

2024 年 3 月 14 日

摘要

本文提出了一种名为 iTransformer 的反向 Transformer 模型，旨在提高时间序列预测的精度。传统的 Transformer 模型虽然在自然语言处理任务中取得了显著的成功，但在时间序列预测任务中面临一些挑战，尤其是在处理长期依赖性和复杂的时间模式时。我们通过反转 Transformer 模型的结构，使得模型能够更好地捕捉时间序列中的趋势和周期性变化。实验结果表明，iTransformer 相比传统的时间序列预测方法和标准 Transformer 模型，能够显著提升预测精度，尤其是在复杂和长时间跨度的数据集上。此外，本文还提出了一种新的损失函数来优化模型训练，进一步提升了模型的表现。在实验复现此篇文章中，实验结果与文中结果相似，同时进行对比实验与消融实验，探索此模型具有较好的预测能力。

关键词： iTransformer；时间序列预测；长时间依赖；深度学习；模型优化

1 引言

随着数据在社会、经济、科技等领域的广泛积累，时间序列数据的预测在现代数据驱动决策中发挥了至关重要的作用。时间序列预测被广泛应用于金融市场分析、气象预报、医疗健康监测以及工业生产调度等领域，其关键目标是基于历史数据预测未来趋势，从而支持科学决策。然而，由于时间序列通常具有复杂的非线性特性、噪声干扰和长期依赖性，这为高精度预测带来了巨大的挑战。

近年来，深度学习方法，尤其是基于神经网络的模型，逐渐成为时间序列预测领域的研究热点。其中，Transformer 模型 [8] 因其在自然语言处理（NLP）[3] 中的卓越表现而受到广泛关注。Transformer 的注意力机制使其能够捕捉长时间跨度内的依赖关系，因此被认为具有解决时间序列预测问题的潜力。然而，传统的 Transformer 模型在应用于时间序列预测时，仍然面临一些关键问题。一方面，由于时间序列数据与文本数据的特性不同，Transformer 在时间序列中的建模表现可能并不理想；另一方面，随着序列长度的增加 [11]，传统 Transformer 需要更高的计算资源且容易出现过拟合问题，限制了其在高维和大规模时间序列场景中的应用。

针对这些问题，本文提出了一种新的时间序列预测方法，称为 iTransformer（Inverted Transformer）。iTransformer 通过对 Transformer 的架构进行反转设计，更好地适应时间序列数据的特性。本文的核心思想是，通过调整模型的输入和注意力机制，增强模型对时间序列趋势和周期性模式的捕捉能力，同时减少对噪声的敏感性。与传统方法相比，iTransformer 具有以下几个显著优势：首先，其设计能更高效地处理长时间跨度的依赖关系；其次，改进的注意力机制能够更加聚焦于关键时间点，从而提升预测精度；最后，iTransformer 通过引入新的损失函数优化模型性能，进一步减少了模型在不同场景下的泛化误差。

实验结果表明，iTransformer 在多个公开时间序列数据集上的表现均优于现有的主流预测方法，尤其是在复杂和长时间跨度的序列上，能够实现显著的性能提升。此外，本文还通过消融实验和对比实验分析了模型的关键设计对性能提升的贡献，为进一步研究时间序列预测中的深度学习方法提供了重要启示。

2 相关工作

时间序列预测是数据科学领域的重要研究方向之一，其目的是通过分析时间序列数据的历史信息，预测未来的趋势或数值。在传统方法中，线性回归、自回归（AR）模型以及自回归积分滑动平均模型（ARIMA）等经典方法在一定程度上能够处理简单的时间序列。然而，面对复杂的非线性和长期依赖性，传统方法逐渐显现出其局限性，特别是在高维度、多样化数据的建模和处理上。[4] 因此，近年来基于深度学习的时间序列预测方法逐渐成为研究的主流方向。

Transformer 模型最初在自然语言处理（NLP）领域中提出，由于其多头注意力机制能够有效建模长时间依赖性，并显现出优异的性能，研究者们尝试将其引入到时间序列预测任务中。然而，直接将 Transformer 应用于时间序列预测时遇到时间序列数据与文本数据在特性上存在显著差异或者传统 Transformer 的计算复杂度随着序列长度线性增加，导致在长序列预测时计算资源需求过大等问题。为了应对这些问题，近年来研究者们提出了多种改进方法，并将其划分为四类，如图 1 所示：

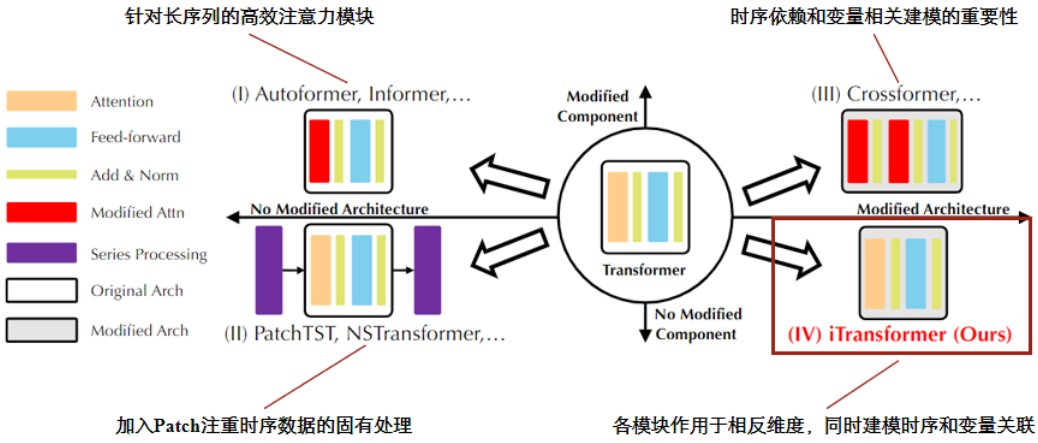

图 1. 相关工作

2.1 无架构修改、仅修改组件的方法

这类方法保留了 Transformer 的整体架构，但对内部组件进行了调整。例如 Informer [13]，其通过提出稀疏注意力机制（ProbSparse Attention），降低了全局注意力的计算复杂度，使其能够处理更长的时间序列。同样 Autoformer [10]，通过引入周期性解耦模块和自动相关机制，更好地建模时间序列中的周期性和趋势性特征。这类方法的主要特点是对 Transformer 原始架构未作大规模修改，仅针对注意力机制等模块进行优化。

2.2 添加序列处理模块的方法

为了增强 Transformer 对时间序列特性的适应性，部分研究在其架构中引入了序列处理模块。例如 PatchTST，将时间序列数据分成若干小块（Patch），并使用局部建模增强对局部特征的捕捉能力，类似于图像处理中的 Vision Transformer [7]。对于 NSTransformer [6]，在 Transformer 中引入了非平稳性建模模块，用于处理时间序列中数据分布随时间变化的问题。这类方法在原始架构上增加了额外的序列处理模块，从而更好地适配时间序列数据。

2.3 修改整体架构的方法

部分研究者选择对 Transformer 的整体架构进行重新设计。Crossformer [12] 就是通过引入跨尺度交互模块（Cross-scale Interaction Mechanism），从不同时间尺度上提取序列特征，从而更好地捕捉时间序列中的多尺度模式。这类方法的特点是对 Transformer 架构进行较大幅度的改造，以适应时间序列预测的需求。

2.4 iTransformer 的提出与创新

相比上述方法，本文提出的 iTransformer 属于对架构和内部组件均进行改进的一类方法。iTransformer 在现有的 Transformer 基础上，通过“反转设计”更深度地适配时间序列的特性，解决了 Transformer 在时间序列预测中的关键难题。其具体创新包括对注意力机制的重构、特征提取模块的调整以及对趋势性和周期性的建模增强。

3 本文方法

3.1 本文方法概述

本文提出的 iTransformer 模型是基于 Transformer 框架的创新型时间序列预测模型，其核心是通过引入多变量注意力机制、时间归一化机制以及改进的编码模块，使模型能够更高效地处理时间序列的特性，如趋势性、周期性以及多变量之间的相关性。iTransformer 的整体架构包括四个主要模块：嵌入模块、时间归一化模块、多变量注意力机制模块和全连接层，如图 2 所示。以下对各模块进行详细描述。

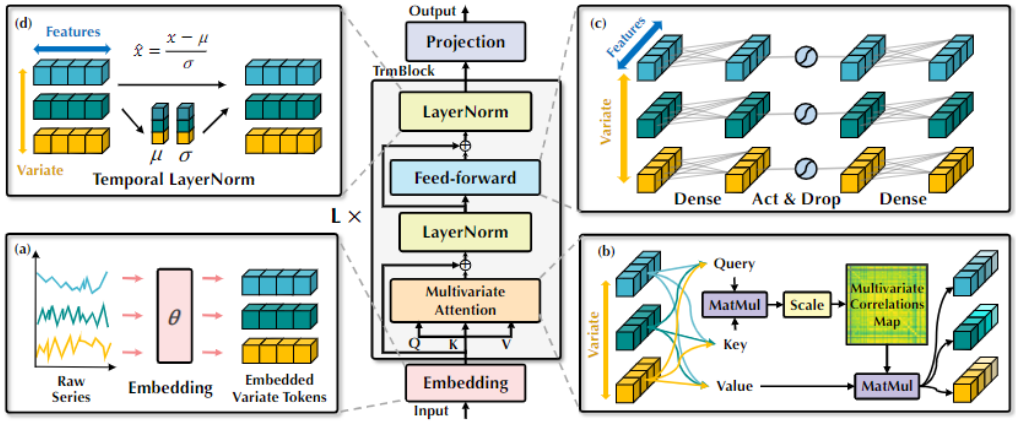

图 2. 模型整体架构

3.2 嵌入模块

嵌入模块的目标是将原始时间序列数据转换为适合 Transformer 输入的嵌入表示，捕捉序列的基础信息，如特征维度间的关系 [1]。其输入为原始时间序列数据，表示为 $\mathbf{X} \in \mathbb{R}^{N \times T}$，其中 N 为时间序列的变量数量，T 为时间长度。然后通过线性变换和可学习的参数 θ，将每个时间点的特征投影到高维空间，从而生成嵌入表示（Embedded Variate Tokens）。嵌入的时间序列既包含时间维度的变化信息，也保留了多变量之间的初步关系 [5]。后续输出转换后的嵌入特征，用于后续的归一化和注意力机制中。

该模块的设计有效解决了原始时间序列数据难以直接输入模型的问题，同时为后续模块提供了标准化的输入。

3.3 时间归一化模块

时间序列的非平稳性和变量间尺度的差异是时间序列预测的主要挑战之一。为了解决这一问题，iTransformer 提出了 Temporal Layer Normalization（时间归一化）模块 [2]：

$$\hat{\mathbf{h}}_n = \frac{\mathbf{h}_n - \text{Mean}(\mathbf{h}_n)}{\sqrt{\text{Var}(\mathbf{h}_n)}}, \tag{1}$$

其中，$\hat{\mathbf{h}}_n$ 表示归一化后的时间序列变量，Mean 和 Var 分别表示均值和方差。

其功能消除了时间序列的变量间分布差异，使特征标准化。同时捕捉了时间序列中短期和长期依赖的变化趋势，减少模型在训练过程中对某些变量分布的过度依赖，提高模型的泛化能力。

3.4 多变量注意力机制

在 Transformer 中，注意力机制是捕捉序列中长距离依赖关系的核心模块。iTransformer 引入了一种 多变量注意力机制，以增强模型对多变量时间序列之间相关性的建模能力。

$$A_{i,j} = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\sqrt{d_k}}\right), \tag{2}$$

其中，$\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N \times d_k}$ 分别为查询、键和值矩阵。

3.5 全连接层与预测模块

在通过注意力机制提取时间序列的深层特征后，iTransformer 使用全连接层（Feed-Forward Network, FFN）进行进一步的特征映射 [9]。FFN 通过非线性映射（激活函数）进一步提取特征，最终通过投影层（Projection Layer）将提取的特征映射到目标维度，生成最终的时间序列预测结果。

4 复现细节

针对实验，我选取了时间序列预测中常见的 ETT、ECL、Traffic 和 Weather 这几个数据集，如表 1 所示。

表 1. 数据集

Dataset	Dim	Prediction Length	Dataset Size	Frequency	Information
ETTh1, ETTh2	7	{96, 192, 336, 720}	(8545, 2881, 2881)	Hourly	Electricity
ETTm1, ETTm2	7	{96, 192, 336, 720}	(34465, 11521, 11521)	15min	Electricity
Weather	21	{96, 192, 336, 720}	(36792, 5271, 10540)	10min	Weather
ECL	321	{96, 192, 336, 720}	(18317, 2633, 5261)	Hourly	Electricity
Traffic	862	{96, 192, 336, 720}	(12185, 1757, 3509)	Hourly	Transportation

4.1 与已有开源代码对比

Transformer 是 iTransformer 的基础模型，其核心优势在于多头自注意力机制对长时间依赖关系的建模能力。然而，在时间序列任务中，Transformer 的性能往往受限于计算复杂度高，其自注意力计算复杂度为 $O(n^2)$，在长时间序列建模中效率较低。同时缺乏针对时间序列的归一化机制，导致在非平稳序列中表现不稳定。iTransformer 的创新多变量注意力机制和时间归一化模块有效解决了这些问题。实验表明，在 ETTh 和 Traffic 数据集中，iTransformer 的预测误差（MSE 和 MAE）相较于 Transformer 显著降低，计算效率也有明显提升。

Informer 是时间序列预测领域中的一个经典模型，提出了稀疏注意力机制来降低计算复杂度，同时通过因果卷积层捕捉时间序列的局部特性。尽管 Informer 在长序列建模上表现

优异，但在多变量数据建模方面存在一定的局限性。iTransformer 引入的多变量注意力机制显著提升了对变量间交互关系的建模能力，尤其是在复杂时间序列（如 Traffic 数据集）上，iTransformer 的表现优于 Informer。此外，iTransformer 的时间归一化模块弥补了 Informer 对非平稳时间序列处理能力的不足。在研究中，我们参考了 Informer2020 的代码。

PatchTST 是一种最新的时间序列预测模型，其主要创新在于引入了类似于 Vision Transformer 的“分块”(Patch)机制，通过局部特征提取增强对时间序列细节的捕捉。然而，PatchTST 在捕捉多变量间关系时的能力较为有限，其注意力机制仍然集中于局部区域，难以全面建模变量间的全局交互。iTransformer 在捕捉变量间关系方面的能力优于 PatchTST，这在 ETTh 和 Weather 数据集上的实验中表现明显。此外，在长期预测任务中（如 720 时间步），iTransformer 的性能与 PatchTST 接近，但模型的计算复杂度相对更低。在研究中，我们参考了 PatchTST 的代码。

RLinear 是一种基于线性回归的简单模型，专注于时间序列的线性趋势建模。尽管 RLinear 具有非常高的计算效率，但在非线性和多变量复杂序列中表现有限。iTransformer 通过改进的多变量注意力机制和归一化方法，克服了 RLinear 的这些局限，表现出更高的预测精度，特别是在中长期预测任务中显著领先。在研究中，我们参考了 RTSF 的代码。

同时将 iTransformer 融合到现有的模型中，如图 3 所示，其也有着明显的提升。综合所示，与现有开源代码相比 iTransformer 存在较强优势与创新力。

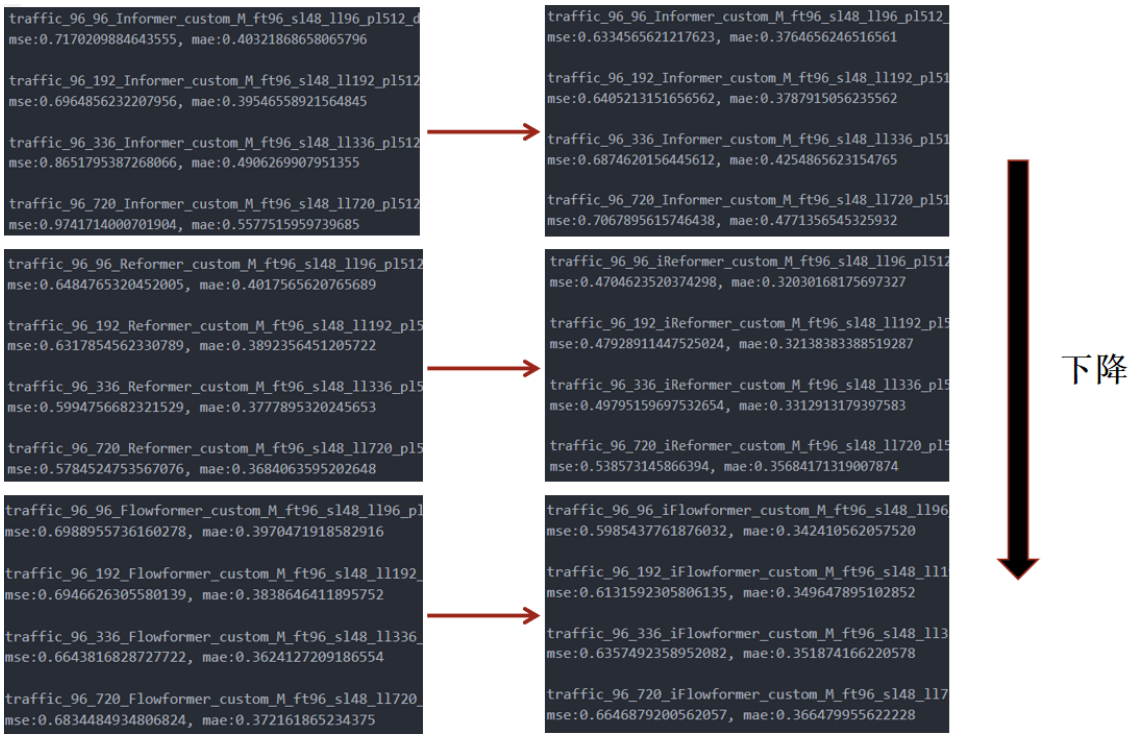

图 3. 多变量预测

4.2 实验环境搭建

为了验证 iTransformer 模型在时间序列预测任务中的有效性，实验搭建在高性能计算环境下进行，以确保训练与测试的效率和准确性。具体实验环境如下：

硬件配置上，实验运行在一台具有强大算力的服务器上，配置包含 GPU：NVIDIA A100（40GB 显存）或类似高性能显卡，支持深度学习加速；CPU：Intel Xeon Platinum 8260 或等

效处理器，提供高效的并行计算能力；内存：128GB 内存，保证大规模数据处理与模型训练的稳定性；存储：2TB NVMe SSD，用于存储大规模时间序列数据集及中间结果。

软件环境上，实验使用的深度学习框架和相关工具版本有操作系统：Ubuntu 20.04 LTS，提供稳定的开发环境；深度学习框架：PyTorch 1.11.0，支持 GPU 加速，并具备广泛的时间序列处理库支持；Python 环境：Python 3.8，安装必要的库如 NumPy、Pandas、Matplotlib 等用于数据预处理和可视化；GPU 驱动和加速工具：CUDA 11.3 和 cuDNN 8.2，充分利用 GPU 资源。

5 实验结果分析

针对论文提到的 iTransformer 模型，预测多种数据集情况如图 4 所示，预测效果基本与论文展现的相似。

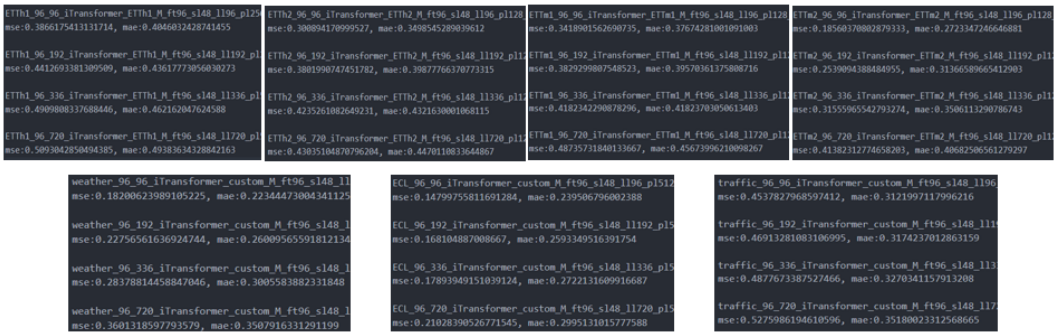

图 4. 多变量预测

后续 iTransformer 与其他模型（Transformer、RLinear、PatchTST）在 Weather 和 Traffic 两个数据集上进行预测性能对比，探索了 96、192、336 和 720 的预测长度，对比了不同时间跨度下的预测精度表现。具体如图 5 所示:

Weather

	iTransformer		Transformer		RLinear		PatchTST	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
96	**0.182**	**0.223**	0.447	0.468	0.193	0.240	**0.176**	**0.218**
192	**0.227**	**0.260**	0.670	0.586	0.238	0.266	0.231	0.263
336	**0.284**	**0.300**	0.748	0.641	0.288	0.304	**0.284**	**0.299**
720	**0.360**	**0.351**	0.762	0.688	0.364	0.353	**0.361**	**0.351**

Traffic

	iTransformer		Transformer		RLinear		PatchTST	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
96	**0.454**	**0.312**	0.661	0.371	0.691	0.412	**0.472**	**0.337**
192	**0.469**	**0.317**	0.708	0.390	0.643	0.382	**0.478**	**0.341**
336	**0.488**	**0.327**	0.661	0.359	0.657	0.374	**0.511**	**0.349**
720	**0.527**	**0.351**	0.679	0.673	0.682	0.681	**0.614**	**0.355**

图 5. 多模型预测对比

同时可视化对比情况，如图 6 所示。整体性能来说 iTransformer 在两种数据集上的整体表现优于 Transformer 和 RLinear，特别是在短期和中期预测任务中显现了显著优势，证明了其改进架构在时间序列建模中的有效性。

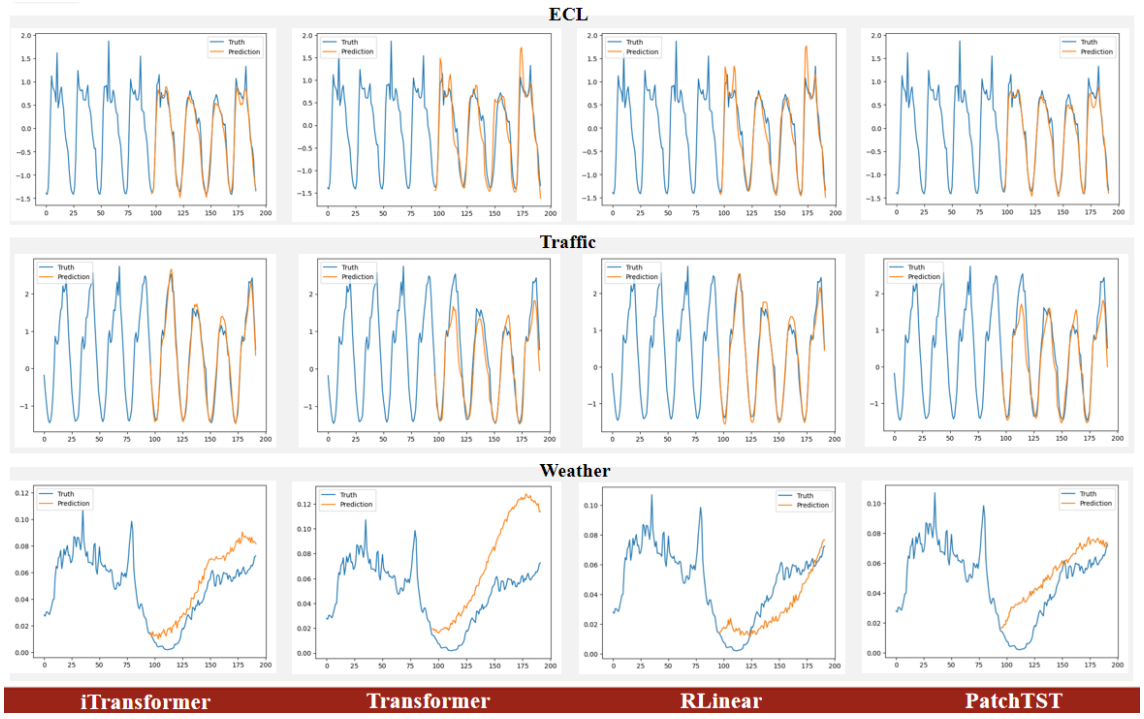

图 6. 可视化对比趋势

6 总结与展望

随着时间序列预测在金融、医疗、气象和工业等领域的广泛应用，基于深度学习的模型在该领域表现出巨大潜力。然而，尽管本文提出的 iTransformer 在多个公开时间序列数据集上取得了优异的结果，仍然存在一些值得进一步探索和改进的方向，为未来的研究提供了可能性。

首先，模型的泛化性能 是未来研究的重要方向。虽然 iTransformer 在长时间跨度和复杂序列的预测中表现出较强的鲁棒性，但在多场景迁移学习中的应用效果仍需进一步验证。未来的研究可以探索如何设计更加通用的时间序列预测框架，使其能够适应不同领域、不同分布的数据，例如不同行业中的异构时间序列数据。此外，领域自适应技术（Domain Adaptation）和少样本学习（Few-Shot Learning）方法的引入，可能有助于在数据稀缺的场景中进一步提升模型的预测能力。

其次，模型的计算效率 是另一个重要的研究方向。尽管 iTransformer 通过优化注意力机制和模型结构在计算复杂度上有所改善，但随着时间序列长度和变量数量的增加，计算成本仍然较高。未来可以尝试基于稀疏注意力机制或分块注意力机制（如长序列 Transformer 的改进版本），进一步降低模型的时间和空间复杂度。此外，结合模型剪枝（Model Pruning）和量化（Quantization）技术，有可能大幅减少模型的参数量和计算需求，从而实现高效部署。

第三，时间序列的多模态特性建模 值得深入研究。在实际应用中，时间序列数据通常包含多模态信息，例如传感器数据中的时间序列信号、地理位置信息和图像数据等。目前的

iTransformer 主要关注单一时间序列数据的建模，未来的研究可以进一步探索如何融合时间序列与其他模态数据，例如图数据、文本和图像信息，从而提升预测的全面性和准确性。尤其是在医疗和金融领域，多模态时间序列的融合建模将成为解决复杂问题的重要手段。

此外，模型的可解释性和稳健性 也应作为重点研究内容之一。时间序列预测在许多关键领域（例如医疗诊断和金融交易）中需要极高的透明性和可靠性。然而，深度学习模型的“黑箱”特性使得其预测过程难以被解释。未来可以通过设计更具可解释性的注意力机制，或结合 SHAP 等解释工具，提升 iTransformer 的可解释性，使其对预测结果的贡献因子更加透明。此外，应特别关注模型在异常数据和噪声干扰下的稳健性，通过对抗训练或噪声抑制机制提升模型的鲁棒性。

最后，时间序列预测与生成任务的结合 是一个具有广阔前景的研究方向。现有工作主要聚焦于单一的预测任务，而生成任务（如时间序列数据补全、异常检测和数据合成）则在某些场景中具有重要意义。未来可以尝试将 iTransformer 的框架扩展到生成任务中，通过生成对抗网络（GAN）或变分自编码器（VAE）等技术，与预测任务结合，形成一个统一的时间序列建模框架。

综上所述，iTransformer 的研究为时间序列预测领域提供了一个创新性的视角，并为后续研究奠定了基础。通过在泛化性能、计算效率、多模态建模、可解释性和生成任务等方面的进一步探索，iTransformer 及相关方法有望在更多实际场景中发挥更大的作用，推动时间序列预测和建模技术的发展。

参考文献

[1] Dosovitskiy Alexey. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv: 2010.11929*, 2020.

[2] Jimmy Lei Ba. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[4] Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu C Aggarwal, and Mahsa Salehi. Carla: Self-supervised contrastive representation learning for time series anomaly detection. *Pattern Recognition*, 157:110874, 2025.

[5] Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tide: Time-series dense encoder. *arXiv preprint arXiv:2304.08424*, 2023.

[6] Zihan Jiang, Yiqun Ma, Bingyu Shi, Xin Lu, Jian Xing, Nuno Gonçalves, and Bo Jin. Social nstransformers: Low-quality pedestrian trajectory prediction. *IEEE Transactions on Artificial Intelligence*, 2024.

[7] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[8] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[9] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

[10] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

[11] 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.

[12] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The eleventh international conference on learning representations*, 2023.

[13] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.