D-Sparse: 减少对超参数 w 依赖的轻量化模型

刘轩麟 [1] **杨雨航** [2] **黄健** [2]

Abstract

本文介绍的 D-Sparse 模型是对于实现长期时间序列预测任务（LTSF）的一次探索，以 SparseTSF 模型的稀疏技术为基础，力求模型的轻量化，以保证模型可以在有限的计算资源下进行训练推理。同时应用了 DLinear 模型有效提取趋势性信息的方法，增强对趋势性信息的提取能力利用趋势性信息和周期性信息建模，旨在降低 SparseTSF 模型对于超参数 w 的依赖，并且提升了 SparseTSF 在长周期时间序列预测上的性能，尤其适合不明确数据周期或数据集周期不明显的情况。该模型在相对较少的参数规模下实现了与近期推出的模型相近的性能，在主流数据集中有着较好的表现，展现出了较强的竞争力。

由于本篇论文是在 SparseTSF (Lin et al., 2024) 模型基础上的改进，所以文中“原模型”指的是 SparseTSF。

1. 论文介绍

长期时间序列预测可以对工业、商业和日常生活的多种情境做出预测，从而为从业者或相关部门提供决策的建议，因此对不同种类的时间序列实现精准预测是研究者共同的目标。随着机器学习、深度学习的发展，长期时间序列预测取得了长足的进步，不少研究者基于 Transformer 架构搭建时间序列模型，取得了很好的预测效果。近来也有研究者致力于轻量化模型的研究，SparseTSF 实现了使用极少的参数实现长期时间序列预测，并且达到了和以往大参数量模型相仿甚至更优异的性能，这使得我们在有限计算资源上部署时间序列预测模型成为可能。但 SparseTSF 是基于“要预测的数据往往表现出恒定的、先验的周期性”这一假设的，这就很自然地引出周期超参数 w 在模型中的应用。但在实际情况中数据的周期性可能是未知的或者缺乏周期性，这种对超参数 w 的依赖就可能降低模型的性能，至少是缺乏通用性和普适性的。SparseTSF 论文中也提到了模型的另一不足，即长周期预测任务中模型性能有较大幅度下降，这与稀疏技术的特性有关。在这篇论文中，我们希望针对原模型依赖超参数 w 这一问题作出改进，在原模型“周期聚合”的基础上增强对趋势性信息的提取，借鉴 DLinear 模型的思路，采用趋势性和季节性信息分解的方式，使得在难以确定超参数 w 的情况下提取出趋势性信息，以此来平衡超参数 w 不准确可能造成的性能下降问题；同时，我们的模型相当于在 SparseTSF 模型的基础上增加了专门训练趋势性信息的参数，这使得我们的模型在长周期序列预测上的表现优于原模型。

先直观地展示我们的可视化结果（ETTh1，$H = 720$）：

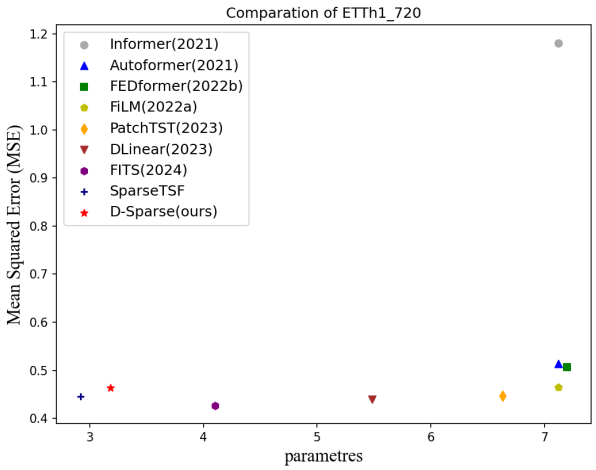


Figure 1: D-Sparse 与主流模型在预测误差和参数规模的比较

总而言之，我们的工作贡献如下：

- 保持了 SpaesrTSF 轻量化的特征。
- 在未确定 w 的情况下也能取得较好的预测效果，增强了模型的通用性。
- 在弱周期性的数据集中提升了 SparseTSF 模型的预测表现。
- 弥补了模型长周期预测的不足。

[1] 中国天津，南开大学，计算机学院 [2] 中国天津，南开大学，网络空间安全学院. Correspondence to: <>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

2. 相关工作

过往模型在时间序列预测上的应用 在过去的研究中，Transformer 架构广泛应用于长期时间序列预测中，比如 Informer (Zhou et al., 2021)、Autoformer (?)、FEDformer (Zhou et al., 2022b),Transformer 架构通过编码和注意力机制捕获序列中的信息，从而实现准确的预测，但该架构的缺陷也十分明显，比如参数量多、推理时占用内存大，对于计算资源的要求较高，尽管后续的模型对模型大小做出了改进，但没有根本性地改变这一问题。不过，经过改进的模型，如 PatchTST (Nie et al., 2023) 的预测效果非常好，使人印象深刻。

轻量化预测模型 DLinear (Zeng et al., 2023) 论文中讨论了 Transformer 和线性模型在时间序列预测上的应用，将时间序列分解为趋势和剩余序列，并分别使用两个单层线性网络对这两个序列进行建模以进行预测。这使得预测过程变得简单、高效，并且性能优于很多基于复杂的 Transformer 架构的模型。之后人们在轻量化的领域做出了更多的突破。在优化模型大小方面，FITS (Xu et al., 2024) 模型迈出了重要的一步，它通过复数频域线性变换和低通滤波技术实现 10k 参数量级的时间序列预测，性能也可以达到 SOTA。

SparseTSF 模型 SparseTSF (Lin et al., 2024) 模型在控制模型大小方面取得了突破性进展，通过交叉周期稀疏预测技术，把具有恒定周期的原始序列下采样为子序列进行预测，将原始时间序列预测任务简化为跨周期趋势预测任务，解耦了数据的周期性和趋势，使模型能够稳定地识别和提取周期性特征，同时专注于预测趋势变化，此外该模型极大地压缩了模型的参数大小，大大减少了对计算资源的需求，是一种非常轻量的 LTSF 模型，参数量规模在 1k 左右。令人兴奋的是，SparseTSF 模型使用如此之少的参数量就达成了足以媲美最先进预测模型的预测精度，这增强了人们探索轻量化模型的信心。

3. 实验方法

3.1. 已有的方法

长期时间序列预测 长期序列预测模型的基本思路就是通过获取的历史观测窗口信息，通过不同的变换，计算出预测窗口的信息，我们一般采用 L 表示历史观测窗口，H 表示预测窗口长度，C 表示通道数量 $\bar{x}_{t+1:t+H} = f(x_{t-L+1:t})$, 其中 $x_{t-L+1:t} \in \mathbb{R}^{L\times C}$, $\bar{x}_{t+1:t+H} \in \mathbb{R}^{H\times C}$。我们关注的预测方法（即 $f()$）大都是线性变换，比如 DLinear (Zeng et al., 2023) 和 NLinear (Zeng et al., 2023)。以 DLinear 为例，它将整个历史观测窗口长度 L 作为输入，预测窗口长度 H 作为输出，那么我们的线性层则是大小为 $L \times H$，面对长期时间序列预测任务，预测窗口 H 很大，这必然造成模型参数的大量增长，尽管线性模型在轻量化上做出了尝试，但显然有改进的空间。

下采样技术 为了实现更加轻量化的模型，SparseTSF (Lin et al., 2024) 提出了稀疏技术与线性层结合的方法。其中，降低模型复杂度最核心的步骤是下采样，我们规定周期超参数为 w，线性层的输入长度为 n，其中 $n = \lfloor \frac{L}{w} \rfloor$。这样做的道理在于，如果我们明确数据的周期，那么下采样操作就可以将周期和趋势性信息解耦，长度为 n 的序列中保存的就是各周期中某点的变化趋势。我们规定 m 为输出长度，$m = \lfloor \frac{H}{w} \rfloor$，这样一来就大大降低了线性层的参数规模，并且对数据的趋势性信息进行了预测，在长期时间序列预测任务上取得了很好的效果。（图 1 来自 SparseTSF (Lin et al., 2024)）

通道独立策略 通道独立策略（CI）在当下 LTSF 领域得到了广泛应用，以往在处理多变量数据时，我们可以应用多个线性层，得到多组参数，分别对应于不同的通道。而通道独立策略提供了更为简化的方法：关注数据集中的单变量时间序列来进行预测，为多个单变量时间序列共享一套参数。共享函数：$f : x^{(i)}_{t-L+1:t} \in \mathbb{R}^L \to \bar{x}^{(i)}_{t+1:t+H} \in \mathbb{R}^H$ 。这种做法大大降低了模型的规模，DLinear (Zeng et al., 2023) 和 SparseTSF (Lin et al., 2024) 均用到了通道独立策略，这些论文中的结果也表明，共享参数的做法也能取得很好的预测效果。

时间序列分解 在 Autoformer (Wu et al., 2021) 中作者基于滑动平均思想，将数据信息分成了趋势项和周期项。FEDformer (Zhou et al., 2022b) 进一步提出了更为复杂的分解时间序列的方法。DLinear 模型采用类似于 Autoformer 的分解时间序列方法，将时间序列分解为趋势性信息和季节性信息。实现原理如下：

趋势数据（趋势性信息）为 $X_t \in R^{L\times C}$

剩余数据（季节性信息）为 $X_s = X - X_t$

其中趋势性信息 X_t 是通过平均池化操作实现的。分解过程如图 1 所示（图片来源于网络）

3.2. D-Sparse

在真实的物理世界中，我们并非对一切事物都有先验的周期性判断。并且，许多事物的发展缺乏明确的周期特征。为了提高 SparseTSF 模型在数据周期未知和缺乏周期性数据上的通用性，我们提出了在跨周期稀疏预测的基础上分解时间序列，进一步增强对数据趋势性信息的提取，以减少模型对于超参数 w 的依赖。

模型架构 L 表示历史观测窗口的长度，H 表示预测窗口的长度，w 表示周期超参数。x_T 表示趋势性信息，x_S 表示季节性信息我们对输入数据 x 进行分解：

$$x^{(i)}_{T/(t-L+1:t)} = \text{AvgPool}(x^{(i)}_{t-L+1:t}) \tag{1}$$

$$x^{(i)}_{S/(t-L+1:t)} = x^{(i)}_{t-L+1:t} - x^{(i)}_{T/(t-L+1:t)} \tag{2}$$

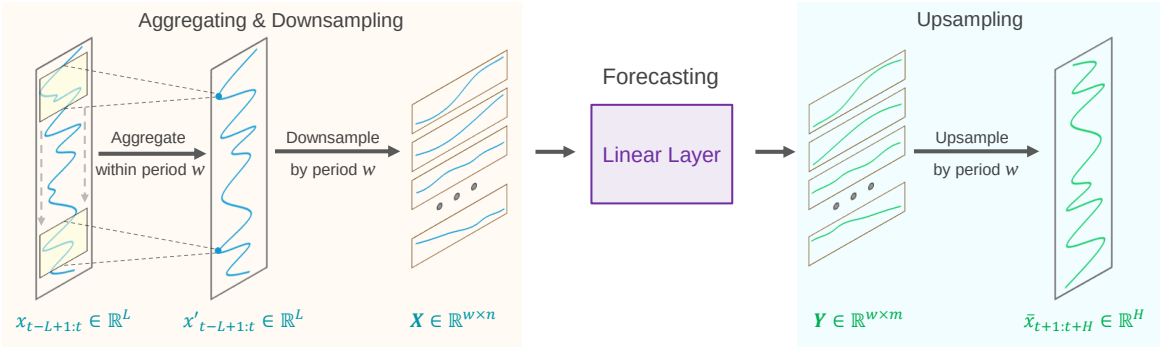


Figure 2: SparseTSF 预测过程

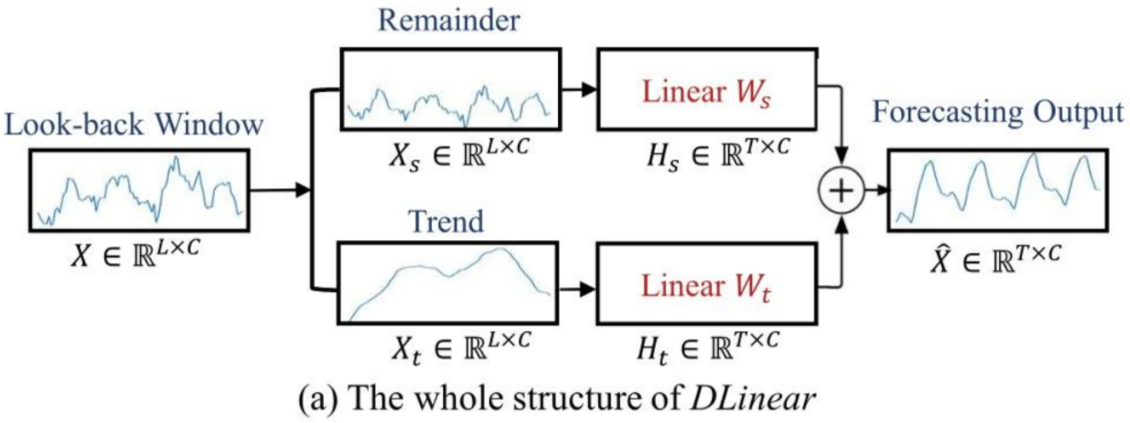


Figure 3: DLinear 时间序列分解示意图

平均池化操作采用经验值 25 大小的核进行池化，并在序列两端用首数据和尾数据填充，保证池化操作后张量大小不变。这样可以将某数据点周围的信息聚合，模拟趋势信息。

接着，我们对两部分信息分别应用跨周期稀疏预测技术。我们先对原始序列进行下采样，转换成长度为 $n = \lfloor \frac{L}{w} \rfloor$ 的下采样序列，这大大减少了输入长度。得到的序列是存储的是各周期相同位置的信息，反映了序列的趋势。在 SparseTSF 论文 (Lin et al., 2024) 中作者解释说：为了解决 (1) 信息丢失，因为每个周期只有一个数据点用于预测，而其他数据点被忽略 (2) 异常值影响的放大，因为下采样子序列中极值的存在会直接影响预测。在下采样之前采用了长度为 w 的一维卷积进行滑动聚合，一维卷积操作的本质是加权平均值，既聚合了周围数据的信息，改善了信息丢失的问题，又可以减轻异常值的影响。我们使用大小为 $2 \times \lfloor \frac{w}{2} \rfloor + 1$ 的卷积核进行卷积运算，对序列两端进行 0 填充，保证张量大小不变。

卷积操作后对序列进行下采样，得到的矩阵如下：

$$x^{(i)}_{T/(t-L+1:t)} \in R^{n \times w} \tag{3}$$

$$x^{(i)}_{S/(t-L+1:t)} \in R^{n \times w} \tag{4}$$

下采样的目的是为了降低参数规模，因为 DLinear (Zeng et al., 2023) 的做法是直接将 $n \times L$ 的矩阵送入线性层。下采样之后，无论是周期 w，还是周期数 n，都远远小于整个观测序列的长度 L。为了更好地捕获趋势性信息，我们对矩阵作转置操作，得到的矩阵如下：

$$x^{i}_{T/(t-L+1:t)} \in R^{w \times n} \tag{5}$$

$$x^{i}_{S/(t-L+1:t)} \in R^{w \times n} \tag{6}$$

这是为了对序列长度为 n 的序列进行预测，好处是实现了周期与趋势的解耦。

对处理得到的矩阵分别应用线性层，相当于季节性信息和周期性信息分别使用两组参数：

$$x^{i}_{S/(t+1:t+H)} = A_S \times x^{i}_{S/(t-L+1:t)} \in R^{w \times m} \tag{7}$$

$$x^{i}_{T/(t+1:t+H)} = A_T \times x^{i}_{T/(t-L+1:t)} \in R^{w \times m} \tag{8}$$

其中线性变换的矩阵 $A_S \in R^{n\times m}$、$A_T \in R^{n\times m}$.m 是线性层的输出长度 $m = \lfloor \frac{H}{w} \rfloor$。

最后将两矩阵进行转置，进行上采样重塑为长度为 H 的预测序列 $x^i_{T/(t+1:t+H)}$ 和 $x^i_{S/(t+1:t+H)}$。将季节序列和趋势序列相加，就得到了最终的预测序列：

$$x^i_{t+1:t+H} = x^i_{S/(t+1:t+H)} + x^i_{T/(t+1:t+H)} \tag{9}$$

我们的模型架构图如图 3。

数据归一化 在时间序列预测任务中，我们常常利用一种样本归一化的策略来提高模型性能。原模型 (Lin et al., 2024; Zeng et al., 2023) 中都用到了类似的处理，将原数据减去均值，得到结果后再加上均值送入模型实现如下：

$$x^{(i)}_{t-L+1:t} = x^{(i)}_{t-L+1:t} - \mathbb{E}_t(x^{(i)}_{t-L+1:t}), \tag{10}$$

$$\bar{x}^{(i)}_{t+1:t+H} = \bar{x}^{(i)}_{t+1:t+H} + \mathbb{E}_t(x^{(i)}_{t-L+1:t}). \tag{11}$$

我们对两部分信息都作归一化，取得了不错的效果。

损失函数 为了与 SparseTSF 模型和 DLinear 模型对齐，便于模型之间更加公平合理地比较，我们也采用了经典的均方误差（MSE）作为损失函数：

$$\mathcal{L} = \frac{1}{C}\sum_{i=1}^{C} \left\| y^{(i)}_{t+1:t+H} - \bar{x}^{(i)}_{t+1:t+H} \right\|_2^2. \tag{12}$$

其中 $\bar{x}^{(i)}_{t+1:t+H}$ 表示预测得到的结果，$y^{(i)}_{t+1:t+H}$ 表示标签序列。

3.3. 理论分析

我们将分析我们改进的模型在参数规模、不同 w 下的通用性、长周期的有效性等方面的原理机制，并适时地与 SparseTSF 模型、DLinear 模型作出比较。

3.3.1. 保持原模型的轻量特点

基于历史观测窗口 L、预测窗口 H、周期超参数 w,SparseTSF 输入线性层的序列长度为 $n = \lfloor \frac{L}{w} \rfloor$，输出的序列长度为 $m = \lfloor \frac{H}{w} \rfloor$; 再加上卷积的信息，需要参数规模 $\lfloor \frac{L}{w} \rfloor \times \lfloor \frac{H}{w} \rfloor + 2 \times \lfloor \frac{w}{2} \rfloor + 1 \ll L \times H$

D-Sparse 由于只是将原输入数据分成了两部分再分别应用线性层，所以参数量只是 SparseTSF 的两倍，并没有量级上的增长，这说明我们成功保持了原模型轻量化的特点。

同样用到序列分解技术 DLinear 模型直接以观测序列长度 L 作为输入、预测序列 H 作为输出，参数规模达到了 $L \times H$ 级别，D-Sparse 模型是远小于这个规模的。

3.3.2. 强化对趋势的预测

在原论文的论述中，通过下采样将时间序列信息解耦周期性信息和趋势性信息，用 P 表示周期，在应用线性层的维度则有：

$$P(t) = P(t + w). \tag{13}$$

这样一来，在预测公式的形式上就可以做出调整，

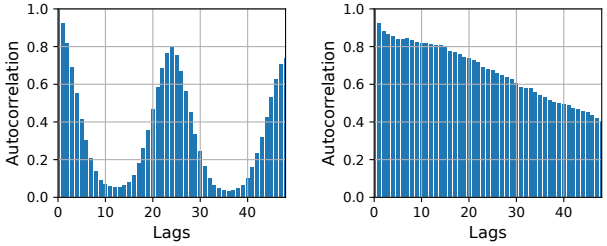
原公式:

$$x_{t+1:t+H} = f(x_{t-L+1:t}) \tag{14}$$

调整后的公式：

$$x'_{t+1:t+m} = f(x'_{t-n+1:t}) \tag{15}$$

由以上三个公式不难看出原模型在对序列趋势的预测方面做出了很大的努力。在设定的 w 完全符合序列周期时可以几乎完美地提取出趋势信息（图 3 来自于 SparseTSF 论文 (Lin et al., 2024)）。



(a) Original (b) Downsampled

Figure 5: 降采样前后自相关性对比

可以看出，原始序列具有明显的周期性，而下采样子序列仅保留趋势特征。

但由于我们之前提到的如果 w 不明确或者数据缺乏周期性，这种解耦周期和趋势的方法就会受到相应的影响：这种方法看似有效解耦了周期和趋势，但这种解耦的过程确实依赖于周期 w 的。

既然周期在模型中发挥了很强的影响作用，那么我们不妨提高趋势的影响程度，即对原始数据进行趋势和周期的解耦。这样我们的趋势序列可以相对独立地完成预测，提高了趋势信息在预测中对结果的影响，我们可以用如下公式来表达：

$$t'_{t+1:t+m} = f(t'_{t-n+1:t}) \tag{16}$$

$$r'_{t+1:t+m} = f(r'_{t-n+1:t}) \tag{17}$$

其中 t 表示趋势信息、r 表示剩余信息，这使得模型更显式地关注周期的变化。

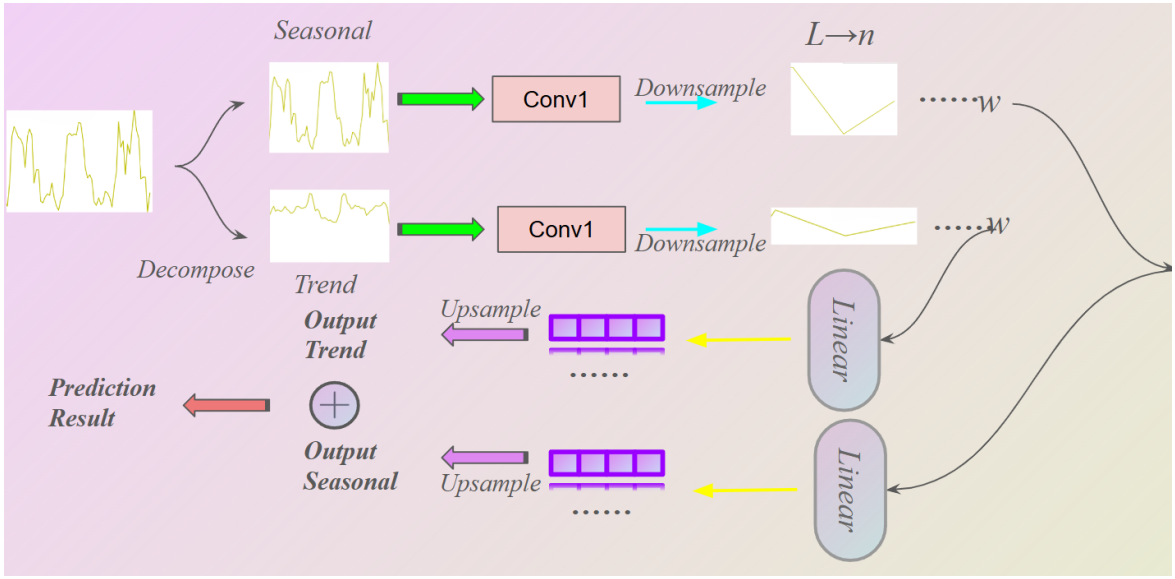


Figure 4: D-Sparse 模型示意图

3.3.3. 关于长周期序列预测的解释

原论文中提到，当下工作的缺陷之一是对超长周期的预测（主要指较大的超参数 w）。因为在涉及超长周期的情况下（例如，周期超过 100），稀疏技术会导致参数连接过于稀疏。因此，SparseTSF 在这样的场景中不能实现最佳的性能。我们对此的理解是输入全连接层的数据点之间在原序列中“相距太远”，也就是所谓的“过于稀疏”，难以很好地把握合理且正确的趋势。

对此，D-Sparse 模型中预先提取趋势性信息的方法就可以发挥作用。尽管应用线性层时仍然沿用 SparseTSF 的架构，但预先提取出的趋势性信息使用一套参数进行预测，在一定程度上弥补了数据点过于稀疏而难以反映趋势信息的不足，从而改善了长周期预测性能下降的问题。

总而言之，我们的 D-Sparse 模型沿用了原模型的下采样技术，保留了轻量化的特征。同时预先提取趋势信息，增强了模型的通用性和长周期预测能力。

4. 实验检验

我们将在这一部分展示 D-Sparse 模型在主流数据集上的实验结果，并对模型在缺乏周期性的数据集上的效果做出检验。此外，我们还将比较在不同超参数 w 下模型的表现，证明模型更强的通用性。我们也将对长周期数据集及相关超参数设置进行讨论分析，展示我们 D-Sparse 对原模型的改进

4.1. 实验设置

数据集 我们选用了五个主流数据集进行主要结果的展示，包括 ETTh1&ETTh2[1]、ETTm1&ETTm2[2]和 Weather[3]。原论文还呈现了 Electricity[4]和 Traffic[5]的结果，但受限于计算资源，我们没有对这两个数据集进行实验。数据集的基本信息呈现在表格中 1.

Table 1: 数据集汇总

Datasets	ETTh1 & ETTh2	ETTm1&Ettm2	Traffic	Electricity	Exchange-Rate	Weather	ILI
Channels	7	7	862	321	8	21	7
Granularity	1hour	5min	1hour	1hour	1day	10min	1week
Timesteps	17,420	69680	17,544	26304	7588	52696	966

Baselines 我们和近年来有代表性的模型进行比较，包括 Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), Pyraformer (Liu et al., 2022), FEDformer (Zhou et al., 2022b), Film (Zhou et al., 2022a), TimesNet (Wu et al., 2023), and PatchTST (Nie et al., 2023)。此外，我们还与轻量化模型 DLinear (Zeng et al., 2023) 、FITS (Xu et al., 2024) 和 SparseTSF (Lin et al., 2024) 比较。我们自己实验得到的结果将用 * 标注，在主要实验结果中，SparseTSF 和 D-Sparse 的结果是我们自己实验得到的，其余结果均来自于原论文 (Lin et al., 2024)。

实验环境 我们的实验均是在华为 MindSpore2.4.0 框架下进行的，设备均为 CPU（Intel i7-13700H）。在此

[1]https://github.com/zhouhaoyi/ETDataset
[2]https://github.com/zhouhaoyi/ETDataset
[3]https://www.bgc-jena.mpg.de/wetter
[4]https://archive.ics.uci.edu/ml/datasets
[5]https://pems.dot.ca.gov/

前的实验中我们发现，Pytorch 框架和 MindSpore 框架的模型表现相差并不大。

4.2. 主要结果

表格 9 展示了 D-Sparse 和当下最先进的时间序列预测模型的比较结果。这个结果是根据数据的周期性设定最佳超参数 w 运行出的最佳结果。在对原模型做出改进后，模型的最佳效果略有下降，但在部分数据上的表现优于原模型，更重要的是，D-Sparse 以较少的参数维持在了第一梯队的模型表现。

4.3. 缺乏周期性的数据集

原模型的预测需要提供超参数 w, 这说明在已知数据集周期的情况下才能获得模型的最佳效果，如果数据本身本身并没有明显周期（比如 exchange_rate）, 那么超参数 w 的设置将无从下手。实验表明，原模型在面对此类数据时的表现有待提高。下面是在超参数 w 设为 24，预测长度分别为 96、144、336、720 的模型表现对比：

Table 3: exchange_rate:$w = 24$ 模型对比

model	Parameter w(24)		
	336	192	96
Sparse TSF	0.396	0.213	0.104
D-Sparse	**0.382**	**0.209**	**0.093**

改为 w=30, 预测长度分别为 60、120、240、360 时的模型表现比较：

Table 4: exchange_rate:$w = 30$ 模型对比

model	Parameter w(30)			
	360	240	120	60
Sparse TSF	0.472	0.274	0.134	0.077
D-Sparse	**0.432**	**0.261**	**0.120**	**0.067**

我们可以看出, 在周期性不明显的数据集中, D-Sparse 模型的表现优于原模型。因为如果没有合适的 w 的话，原模型很难展现出最佳效果。由于我们的模型增强了对趋势性的预测，所以在弱周期数据集上的表现有所提升。

4.4. 对超参数 w 的依赖性

假设数据集有明显周期并且是先验已知的，那么原模型的预测效果是非常好的。但在真实物理世界中，有些周期并非先验已知, 我们只能设置一些经验周期。在这部分实验中，我们将对 ETTh1 设置不同的超参数 w, 来模拟周期并非先验已知的情况。

我们在预报水平为 96 的情况下，给模型设置 w=4，12，24，48，对 ETTh1 进行预测。结果如下：

Table 5: ETTh1: 不同 w 下的模型对比

	48	24	12	4
Sparse TSF	**0.362**	**0.364**	0.381	0.379
D-Sparse	0.371	0.375	**0.377**	**0.375**

ETTh1 的周期是 24，所以在 $w < 24$ 时，由于不足一个周期，原模型的表现下降，不如 D-Sparse 模型；而 $w = 24, 48$ 时，由于恰好是是整周期，原模型的预测能力提升了，达到了较好的水平。

结合上一部分实验对 exchange_rate 数据集结果的分析，我们发现，改进过后的 D-Sparse 模型对超参数 w 的依赖性的确降低了，这也提高了模型的通用性。

4.5. 长周期预测

原论文中提到超长周期时间序列预测是原模型的短板。ETTm1、ETTm2 和 Weather 都有很长的周期。ETTm1&2 的周期是 96，而 Weather 的周期达到了 144。原论文进行实验验证，发现 $w = 4$ 时，得到了比较好的预测结果。但面对有先验已知周期的数据，进行反复的实验是消耗成本的一件事，我们依次看到 ETTm1、ETTm2、Weather 在 $w = 1\ 4\ 24\ 48\ 72\ 144$ 时两模型的对比结果（预报水平为 720）：

Table 6: ETTm1: 不同 w 下模型对比

model	Parameter w (ETTm1)					
	144	72	48	24	4	1
Sparse TSF	0.454	0.450	**0.420**	**0.418**	**0.413**	0.422
D-Sparse	**0.430**	**0.447**	0.426	0.425	0.414	**0.417**

Table 7: ETTm2: 不同 w 下模型对比

model	Parameter w (ETTm2)					
	144	72	48	24	4	1
Sparse TSF	0.379	0.376	**0.360**	**0.354**	0.361	0.351
D-Sparse	**0.360**	**0.365**	0.361	0.365	**0.360**	**0.350**

Table 8: Weather: 不同 w 下模型对比

model	Parameter w (weather)					
	144	72	48	24	4	1
Sparse TSF	0.339	0.3332	0.328	**0.3236**	0.3235	0.325
D-Sparse	**0.334**	**0.3325**	**0.326**	0.3237	**0.3229**	**0.320**

我们可以发现，D-Sparse 模型在这些 w 下误差变化

Table 2: D-Sparse 模型和主流时间序列预测模型的 MSE 结果比较，并展示了 D-Sparse 在各组数据上与表现最好的模型对比

Dataset	ETTh1				ETTh2				ETTm1				ETTm2			
Horizon	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720
Informer(2021)	0.865	1.008	1.107	1.181	3.755	5.602	4.721	3.647	0.672	0.795	1.212	1.166	0.365	0.533	1.363	3.379
Autoformer(2021)	0.449	0.500	0.521	0.514	0.358	0.456	0.482	0.515	0.505	0.553	0.621	0.671	0.255	0.281	0.339	0.433
Pyraformer	0.664	0.790	0.891	0.963	0.645	0.788	0.907	0.963	0.543	0.557	0.754	0.908	0.435	0.730	1.201	3.625
FEDformer	0.376	0.420	0.459	0.506	0.346	0.429	0.496	0.463	0.379	0.426	0.445	0.543	0.203	0.269	0.325	0.421
TimesNet	0.384	0.436	0.491	0.521	0.340	0.402	0.452	0.462	0.338	0.374	0.410	0.478	0.187	0.249	0.321	0.408
PatchTST	0.370	0.413	0.422	0.447	**0.274**	0.341	**0.329**	0.379	**0.292**	0.330	**0.365**	0.419	0.163	0.219	0.276	0.368
DLinear	0.374	**0.405**	0.429	0.440	0.338	0.381	0.400	0.436	0.299	**0.335**	0.369	0.425	0.167	0.224	0.281	0.397
FITS	0.375	0.408	0.429	**0.427**	**0.274**	0.333	0.340	**0.374**	0.303	0.337	0.366	**0.415**	**0.162**	**0.216**	**0.268**	**0.348**
SparseTSF	**0.363**	0.438	**0.406**	0.446	0.283	**0.332**	0.354	0.396	0.305	0.354	0.424	0.424	0.178	0.226	0.277	0.356
D-Sparse(ours)	0.376	0.412	0.440	0.463	0.281	0.338	0.354	0.396	0.315	0.337	0.367	**0.412**	0.178	0.220	0.278	0.359
Imp.	-0.013	-0.007	-0.034	-0.044	-0.007	-0.006	-0.025	-0.022	-0.023	-0.002	-0.002	+0.003	-0.016	-0.004	-0.010	-0.011

比较小，而原模型误差波动较大，这也印证了我们关于超参数 w 依赖性的论述。如果 w 贴近实验得到的最佳超参数（这里是 w 较小的情况），原模型展现了优异的性能，而随着 w 增大并且贴近数据集真实周期时，D-Sparse 模型的效果实现了对 Sparse-TSF 的反超，尤其时 Weather 数据集中 $w = 144$ 的情况下，w 等于真实周期，我们的模型展现出了预测效果的提升。以上的实验结果说明我们改善了长周期序列预测问题，减少了寻找最佳超参数 w 的成本。

5. 讨论

5.1. 提取趋势信息的区别

分解时间序列和下采样技术都提到了“趋势信息”的概念，我们需要对此加以区分，以便更好地理解我们的模型。

下采样技术提及的“趋势”指的是周期和周期之间的趋势，针对的是每个周期相同位置的数据的点。将这些点输入线性层，能反映跨周期的趋势。

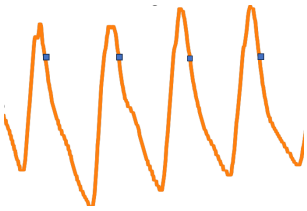


Figure 6: 下采样趋势点

分解时间序列得到的趋势信息指的是我们所理解一般意义上的“趋势”，即在整条时间序列上选取一段作池化操作，针对的是时间序列上的一段段子序列。

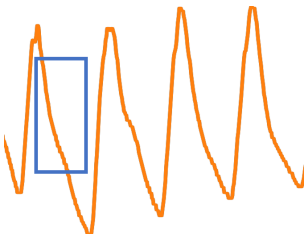


Figure 7: 分解时间序列的趋势信息

5.2. 有待改进的方面

模型最佳性能 ：尽管我们较为有效地改善了原模型长周期序列预测不足，但应用较大的超参数 w, 尽管更贴合数据集的周期，但效果不如原模型通过实验找到的最佳性能；尽管我们减少了模型对 w 的依赖，但也牺牲了模型的最佳性能（大多数情况下 D-Sparse 的最佳性能略差于 SparseTSF）。

小数据集 :national_illness 数据集是时间序列预测中常用的小数据集，并且周期性较弱，在预报水平为 24、36、48、60 的条件下，各模型的表现如下：

Table 9: national_illness 数据集上的表现对比

Dataset	ILL			
Horizon	60	48	36	24
Informer(2021)	5.264	4.763	4.755	5.764
Autoformer(2021)	2.770	2.669	3.103	3.483
Pyraformer	7.762	7.551	7.394	1.420
FEDformer	2.857	2.622	2.679	3.228
LogTrans	5.278	4.800	4.799	4.480
PatchTST	**1.470**	**1.553**	**1.579**	**1.319**
DLinear	2.368	2.130	1.963	2.215
SparseTSF	2.167	2.223	2.283	2.326
D-Sparse(ours)	2.141	2.187	2.265	2.286
Imp.	-0.671	-0.634	-0.686	-0.967

各模型在该数据集上的表现都不尽如人意，PatchTST 模型的表现是最为亮眼的。我们的模型表现和 SparseTSF 差不多，在该模型下表现尚可，略差于将观测窗口应用于全连接层的线性模型，但好于大部分 transformer 架构的复杂模型。无论如何，这些模型的表现都很难让人信服。

5.3. 未来的工作

提升模型表现 在大部分数据集上，D-Sparse 模型的最佳表现略差于最好的模型表现，在一部分情况下不如 SparseTSF 的表现，尽管调整参数的过程中我们见证了 D-Sparse 的稳定性，但如何寻找其最佳表现，或者说怎样能提升模型的最佳表现是我们需要进一步研究的课题。

处理小数据集预测问题 我们看到对于弱周期小数据集，各模型都很难训练出较为准确的参数应用到预测任务上，而现实世界中又有大量的数据集合是弱周期且样本量小的。如何提升模型在这类数据集上的性能，是我们应该关注的问题。

6. 总结

本篇论文中我们介绍了分解时间序列和跨周期稀疏预测技术相结合的 D-Sparse 模型。

它继承了 SparseTSF 模型的轻量化特征，沿用了其中的下采样方法减小参数规模，保证模型可以在有限的计算资源下实现预测。D-Sparse 模型还借鉴了 DLinear 模型中分解时间序列的方法，将序列分解为季节性信息和趋势性信息，可以显式地对趋势进行预测，通过提高“趋势”在预测任务中的占比减少原模型对于超参数 w 的依赖，并优化了模型在长周期预测、弱周期时间序列预测方面的表现。与近年来推出的先进模型相比，，D-Sparse 也基本可以保持相近的水准。在今后的工作中，我们希望找到既能提高模型通用性，又能够提升模型最佳效果的方法，推动时间序列预测模型进一步向轻量化精确预测迈进。

影响

我们的工作减少了 SparseTSF 对超参数 w 的依赖，提高了模型的灵活性和通用性。使得轻量模型在时间序列预测任务方面的应用更为广泛。

References

Challu, C., Olivares, K. G., Oreshkin, B. N., Ramirez, F. G., Canseco, M. M., and Dubrawski, A. Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, X., Li, Y., Tan, J., Wu, B., and Li, F. Oneshotstl: One-shot seasonal-trend decomposition for online time series anomaly detection and forecasting. *arXiv preprint arXiv:2304.01506*, 2023.

Kim, T., Kim, J., Tae, Y., Park, C., Choi, J.-H., and Choo, J. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lin, S., Lin, W., Wu, W., Chen, H., and Yang, J. Sparsetsf: Modeling long-term time series forecasting with 1k parameters. *arXiv preprint arXiv:2405.00946*, 2024.

Liu, S., Yu, H., Liao, C., Li, J., Lin, W., Liu, A. X., and Dustdar, S. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*, 2022.

Madsen, H. *Time series analysis*. CRC Press, 2007.

Nie, Y., H. Nguyen, N., Sinthong, P., and Kalagnanam, J. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Qiu, X., Hu, J., Zhou, L., Wu, X., Du, J., Zhang, B., Guo, C., Zhou, A., Jensen, C. S., Sheng, Z., et al. Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods. *arXiv preprint arXiv:2403.20150*, 2024.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wu, H., Xu, J., Wang, J., and Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., and Long, M. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Xu, Z., Zeng, A., and Xu, Q. Fits: Modeling time series with $10k$ parameters. In *The Twelfth International Conference on Learning Representations*, 2024.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Zhou, T., Ma, Z., Wen, Q., Sun, L., Yao, T., Yin, W., Jin, R., et al. Film: Frequency improved legendre memory model for long-term time series forecasting. *Advances in Neural Information Processing Systems*, 35:12677–12690, 2022a.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., and Jin, R. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pp. 27268–27286. PMLR, 2022b.