TimesMixer

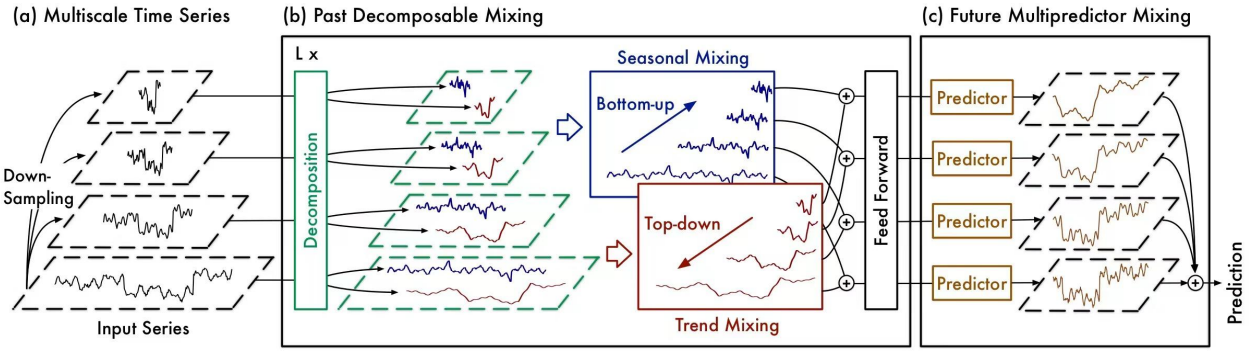
论文内容：

对于时间序列来说，未来的变化是由**多个尺度的共同变化决定**的，本论文通过将时间序列分解为多个尺度，然后通过 MLP 模型来进行时间序列的预测。

创新点：

1. 模型是多尺度混合，可以分解为多个尺度的变化。
2. 最后能多预测器混合多尺度信息，进行互补预测。

模型内容：

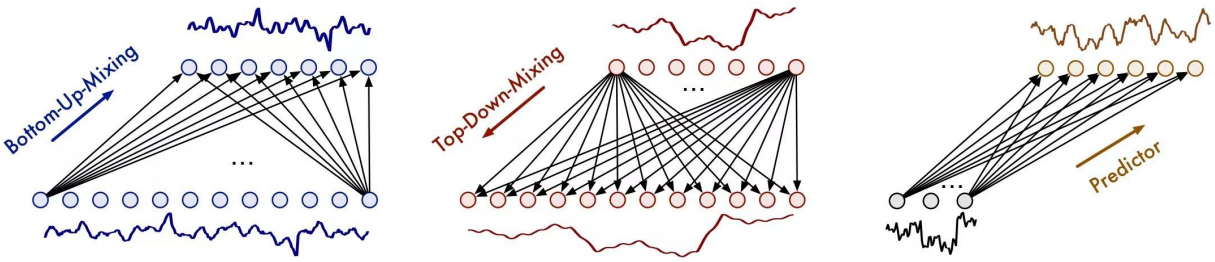


过程： PDM FMX

多尺度分解-- 季节趋势分解--多尺度季节和趋势分量分解 -- 多预测器混合。

季节融合和趋势融合的计算过程：

季节融合是自下而上的计算，趋势融合是自上而下的计算。



- $\mathcal{T}^l = \{\mathbf{t}_0^l, \cdots, \mathbf{t}_M^l\}$: Top-Down-Mixing
  - **自上而下合并信息**：残差的方式

$$\text{for } m: (M-1) \to 0 \text{ do:}$$
$$\mathbf{t}_m^l = \mathbf{t}_m^l + \text{Top-Down-Mixing}(\mathbf{t}_{m+1}^l),$$

$M=1$, $X_1 \in R^{48 \times d_{model}}$, $X_0 \in R^{96 \times d_{model}}$, $X_2 \in R^{24 \times d_{model}}$

$S_1 \in R^{48 \times d_{model}}$, $S_0 \in R^{96 \times d_{model}}$

$T_1 \in R^{48 \times d_{model}}$, $T_2 \in R^{24 \times d_{model}}$

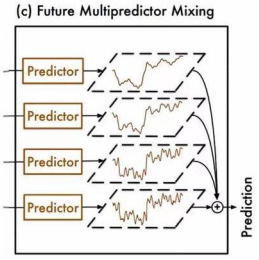
$T_{M=1}^l$: $T_1^l = T_1^l + \text{Mix}(T_2^l)$ { nn.Linear(24, 48), nn.GELU, nn.Linear(48, 48) }

➢ **多预测器混合：** $\widehat{\mathbf{x}}_m \in \mathbb{R}^{F \times C}$

$$\widehat{\mathbf{x}}_m = \text{Predictor}_m(\mathbf{x}_m^L),\ m \in \{0, \cdots, M\},\ \widehat{\mathbf{x}} = \sum_{m=0}^{M} \widehat{\mathbf{x}}_m,$$

预测 F=192

$\sum$
- m=3 : nn.Linear(12, 192), $d_{model} \to C$
- m=2 : nn.Linear(24, 192), $d_{model} \to C$
- m=1 : nn.Linear(48, 192), $d_{model} \to C$
- m=0 : nn.Linear(96, 192), $d_{model} \to C$



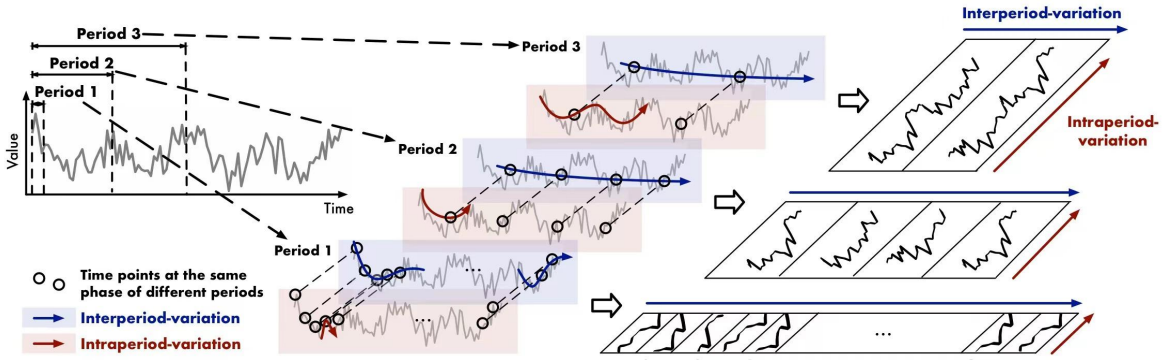
Timesnet

针对问题：

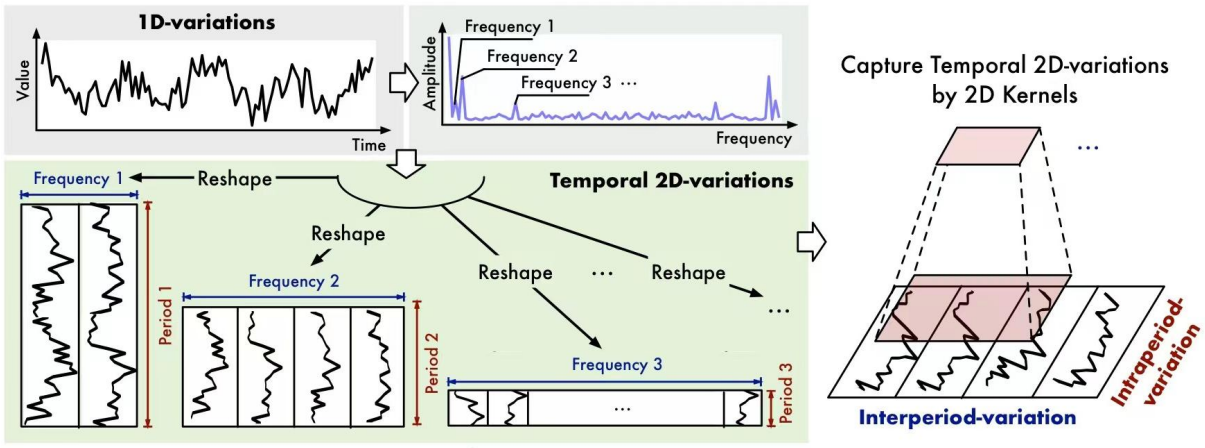
时间序列的多周期性会包含**周期内的变化**和**周期间的变化**，而这两种变化之间会导致相互重叠，相互影响。

创新点：

1. 通过模块化的方法 将一位转换成二维。
2. 模型可以自适应的发现多周期，从二维张量中捕获时间变化。



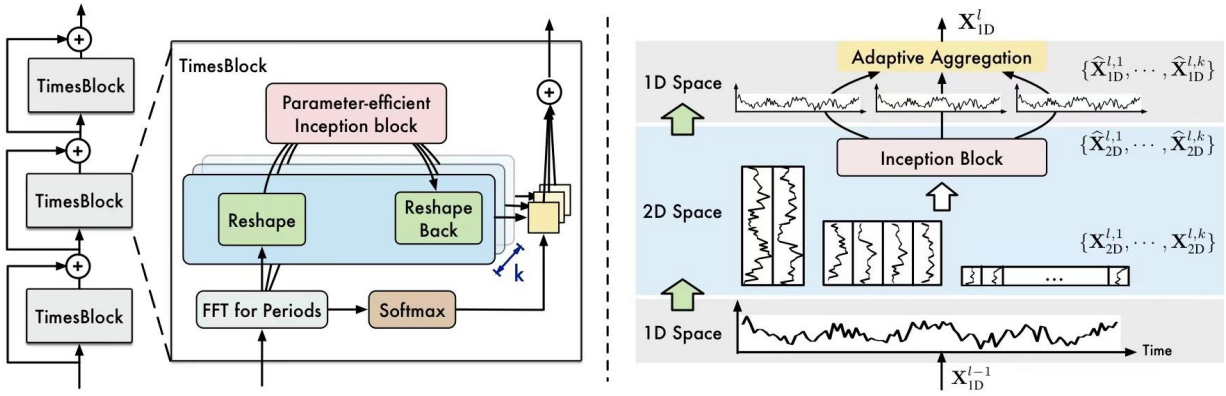
通过 FFT 变化，根据 FFT 之后中最显著的频率振幅来确定其周期。获得不同的 period。



模型内容：

获得 2D 的数据之后，然后就可以用常规的卷积操作来提取特征。

然后根据 FFT 的振幅的大小进行加权计算，完成最后的预测。



TimeXer

论文内容：

对于时间序列的预测，如果仅仅对其内部变量而分析，通常不足以保证准确的预测，所以要对内部变量和外部变量之间差异和依赖性进行挖掘。

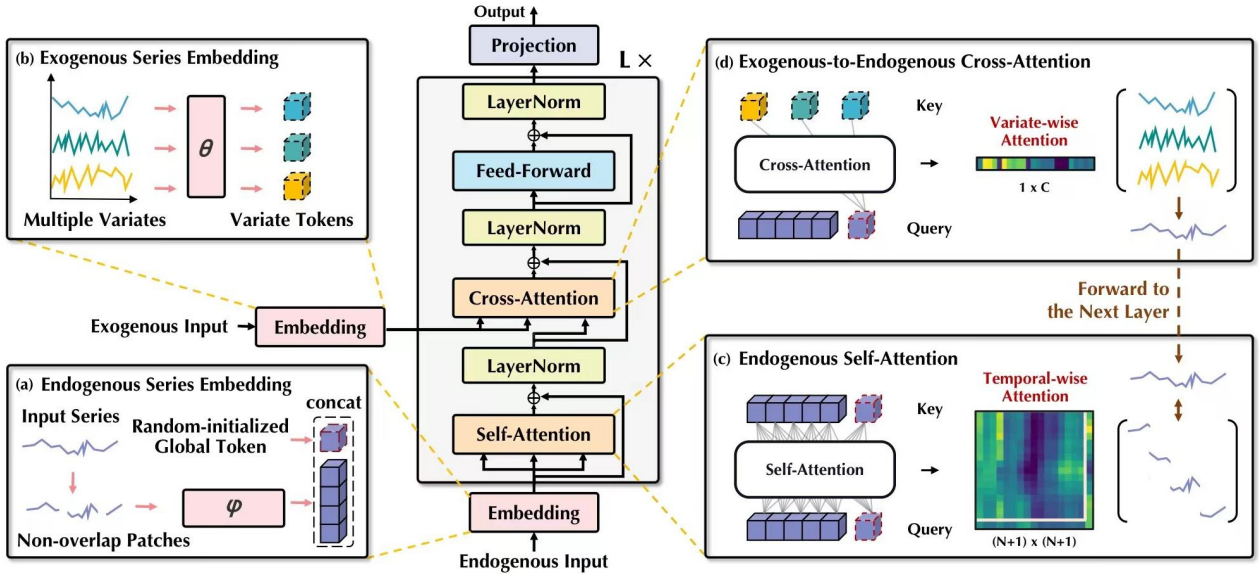
同样的外部因素对内部变量也可能产生连续和时滞的影响。

针对于 patchTST 只能捕获时间依赖性，不能捕获多变量，而 itransformer 无法捕获不同子序列之间的时间变化。

创新点：

通过设计嵌入层，是得能够处理外部变量，并且为内部变量和外部变量设计了不同的嵌套策略。

模型内容：



模型最重要的就是两个 Embedding，self-attention 和 cross-attention 其余和 transformer 一样。

内部序列嵌入：patch 嵌入和原始变量嵌入都用于内部变量，分别获得多个时间 tokens 和一个变量 token。 （最后做自注意力机制计算为（N+1）*（N+1））。

外部序列嵌入：每个外部变量通过变量嵌入作为变量 token 嵌入。

内部自注意力：将自注意力应用于内生时间 token 以捕获补依赖关系。

内外交叉注意力了：采用交叉注意力来对内生和外生变量的序列级依赖性进行建模。（Q 为内部变量，K,V 为内部变量）

外部变量的融合核心：外部 embedding 与内部变量做交叉注意力（cross-attention）