

《Spectral Temporal Graph Neural Network for Multivariate Time-series Forecasting》

《基于谱时间图神经网络的时间序列预测》

摘要：

- 本文的背景：**多元时间序列预测在现实世界的许多应用中发挥着至关重要的作用。然而，这是一个充满挑战性的问题，因为我们需要同时考虑序列内的时间相关性和序列间的结构相关性。而尽管最近已经有许多研究试图捕捉这两种相关性，但大多数研究仍仅仅捕获了时域中的时间相关性，而将预定义的先验条件作为序列间的相关性。因此，本文旨在提出一种端到端的神经网络框架来进一步提高多元时间序列预测的准确性。
- 本文的贡献：**提出了一种新型的神经网络架构——谱时间图神经网络（*StemGNN*），它能够在频谱域上联合捕捉序列间的相关性和时间依赖性。
- 主要创新点：**通过离散傅里叶变换（DFT）和图傅里叶变换（GFT）将多元时间序列从时空域转到频域，并同时在频域上捕捉时空的依赖关系；使用数据驱动的方法来自动学习序列间的相关性以构建邻接矩阵，从而不再需要预先定义邻接矩阵。
- 实验结果：**研究者在十个实际数据集上进行了大量实验，而*StemGNN*在9个基准上取得了*sota*的结果，进而证明了*StemGNN*的有效性。

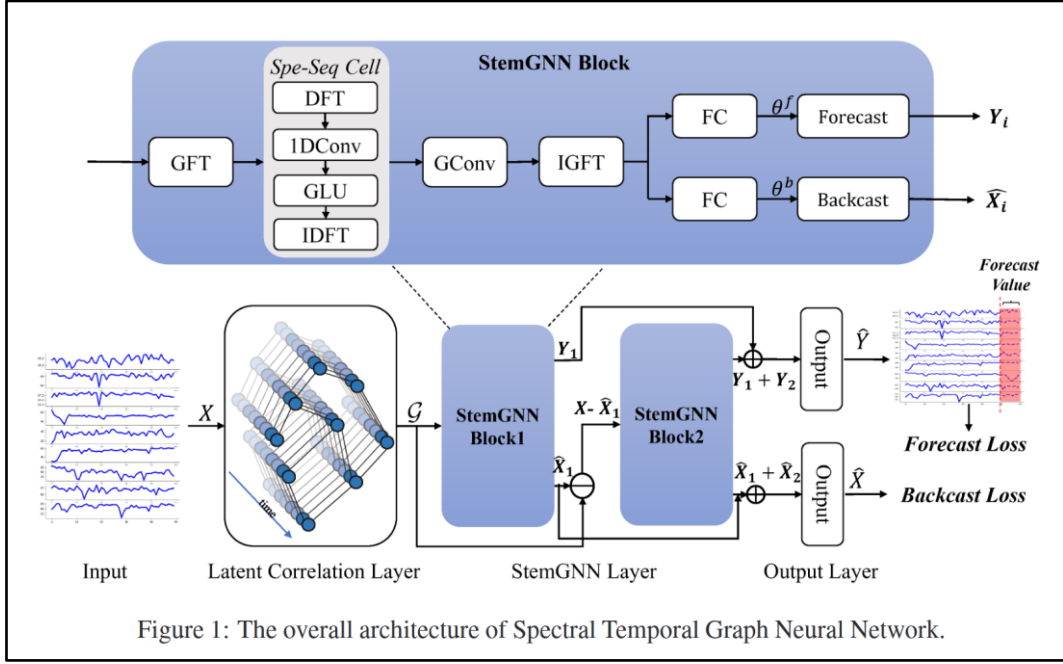
问题定义：

假设所有变量间存在隐藏的内在关联，那么可以将多元变量的观测值用拓扑结构 $\mathcal{G} = (X, W)$ 来表示。其中， $X = \{X_{it}\} \in \mathbb{R}^{N \times T}$ 表示 N 个变量在 T 个时间戳内的观测值； $W \in \mathbb{R}^{N \times N}$ 表示变量间关系的邻接矩阵， $w_{ij} > 0$ 表示存在连接节点 i 和 j 的边，而 w_{ij} 表示这条边的强度。因此，多元时间序列预测问题也可以表示为使用历史已知的 K 个观测值 X_{t-K}, \dots, X_{t-1} （ $X_t \in \mathbb{R}^N$ ）和拓扑结构 \mathcal{G} 来学习一个函数 F 以预测未来的 H 个值：

$$\hat{X}_t, \hat{X}_{t+1}, \dots, \hat{X}_{t+H-1} = F(X_{t-K}, \dots, X_{t-1}; \mathcal{G}; \Phi)$$

其中， Φ 代表函数 F 中待训练的参数。

StemGNN模型架构:



*StemGNN*的整体架构如图 1 所示。

首先，作为输入的多元时间序列 $X \in \mathbb{R}^{N \times T}$ 会被馈送到潜在相关层（Latent Correlation Layer），该结构可以从数据中自动学习推断出序列之间的图结构及其相关的权重矩阵 $W \in \mathbb{R}^{N \times N}$ 。

接着，图 $G = (X, W)$ 会作为由两个 *StemGNN Block* 组成的 *StemGNN* 层的输入。而 *StemGNN Block* 的设计目的是在频谱域中联合建模多元时间序列内的结构和时间相关性（如图 1 的顶部图表所示），它包含着一系列精心设计的运算符。

首先，图傅立叶变换 (*GFT*) 算子会将图 G 转换为频谱矩阵表示，其中每个节点的单变量时间序列会变得线性无关。然后，离散傅立叶变换 (*DFT*) 算子会将每个单变量时间序列分量进一步转换到频域中。而在频域中，相关表示会被送入一维卷积和 *GLU* 子层以捕获特征模式，然后再通过逆离散傅立叶变换 (*IDFT*) 再转换回到时域。最后，我们再将频谱矩阵表示上应用图卷积并执行逆图傅立叶变换 (*IGFT*)。

而对于 *StemGNN Block* 最后的输出结果，则是由两个部分所组成的输出层。其一是 *Forecast* 块，用于产生预测结果 $Y_i \in \mathbb{R}^{N \times H}$ ；其二是 *Backcast* 块，用于产生原始输入数据 X 的还原结果 \hat{X}_i 。而我们还可以看到在两个 *StemGNN Block* 之间，*StemGNN Block2* 接受的输入实际上是 $X - \hat{X}_1$ ，这样的设计类似于残差连接的效果。

最后，对于 $StemGNN$ 的输出层，也有两个输出结果，分别是 $\hat{Y} = Y_1 + Y_2 \in \mathbb{R}^{N \times H}$ 以及 $\hat{X} = \hat{X}_1 + \hat{X}_2$ 。而为了更好地做出预测，本文对这两个输出也设计了新的 $Loss$ 函数：

$$\mathcal{L}(\hat{X}, X; \Delta_\theta) = \sum_{t=0}^T \|\hat{X}_t - X_t\|_2^2 + \sum_{t=K}^T \sum_{i=i}^K \|B_{t-i}(X) - X_{t-i}\|_2^2$$

其中， B 表示生成反向输出的整个网络； Δ_θ 表示网络中的所有参数。

相关说明：上式的前一部分是预测 $Loss$ ，即 $Y - \hat{Y}$ 的损失值，后一部分是回溯 $Loss$ ，即 $X - \hat{X}$ 的损失值。而之所以形式不同，是因为该损失函数 \mathcal{L} 代表的是整个训练过程中的整体损失函数。

潜在相关层（Latent Correlation Layer）：

首先，将 $X \in \mathbb{R}^{N \times T}$ 输入到一层 GRU 中提取序列的潜在时间关联。然后，对 GRU 的最后一个隐藏状态 R 使用 $self - attention$ 的方式计算其邻接矩阵。（之所以使用最后一个隐藏状态是因为它拥有之前所有历史状态的信息。）具体公式如下：

$$Q = RW^Q, K = RW^K, W = softmax\left(\frac{QK^T}{\sqrt{d}}\right)$$

其中， Q 和 K 分别表示 $query$ 和 key 的表示，它们可以通过线性变换计算得出； W^Q 和 W^K 是注意力机制中可学习的参数； d 是 Q 和 K 的隐藏维度的大小；而输出矩阵 $W \in \mathbb{R}^{N \times N}$ 被用作图 \mathcal{G} 的邻接权重矩阵。

$StemGNN$ 块（ $StemGNN$ Block）：

$StemGNN$ 将数据的时空域转换到频域并使用谱序列卷积 $Spe - Seq$ 来捕捉时间序列的周期性和自相关性，使用谱图卷积来提取空间的相关性。

$$Z_j = \mathcal{GF}^{-1} \left(\sum_i g_{\theta_{ij}}(\Lambda_i) \mathcal{S}(\mathcal{GF}(X_i)) \right)$$

其中， \mathcal{GF} 、 \mathcal{GF}^{-1} 和 \mathcal{S} 分别表示 GFT 、 $IGFT$ 和 $Spe - Seq$ Cell， θ_{ij} 是第 i 个输入通道和第 j 个输出通道对应的图卷积核， Λ_i 是归一化拉普拉斯矩阵的特征值矩阵， GFT 中使用的特征向量数相当于未降维的多变量维数(N)。然后，我们将每个输出通道 Z_j 连接起来，得到最终结果 Z 。

说明：关于图傅里叶变换和逆变换（ GFT 、 $IGFT$ ）的细节请参考<https://zhuanlan.zhihu.com/p/147687999>。而在 $StemGNN$ 模型的代码实现中，实际上是使用到了切比雪夫多项式。因此，从个人知识点掌握的角度简单来说， GFT 就是利用了切比雪夫多项式的 GCN 模型。【但 GCN 实际上就是 $ChebNet$ 的公式简化】

谱序单元 Spectral Sequential Cell ($Spe - Seq Cell$):

谱时间卷积 $Spe - Seq$ 通过将每个时间序列分解到频域然后再来学习频域上的特征表示。首先，它对输入数据进行 DFT ，将其从时域转到频域，然后再对实数部分 \hat{X}_u^r 和虚数部分 \hat{X}_u^i 分别进行 $1D - CNN$ 和 GLU :

$$M^*(\hat{X}_u^*) = GLU(\theta_\tau^*(\hat{X}_u^*), \sigma^*(\hat{X}_u^*)) = \theta_\tau^*(\hat{X}_u^*) \odot \sigma^*(\hat{X}_u^*), * \in \{r, i\}$$

其中， θ_τ^* 是实验中大小为3的卷积核； \odot 是 Hadamard 运算符；而非线性正弦门 σ^* 决定了当前输入中有多少信息与序列模式密切相关。

然后，将上述结果的实数输出 $M^r(\hat{X}_u^r)$ 与虚数输出 $M^i(\hat{X}_u^i)$ 相加得到一个复数结果 $M^r(\hat{X}_u^r) + M^i(\hat{X}_u^i)$ ，最后再使用 $IDFT$ 将数据从频域转回到时域。

说明：关于离散傅里叶变换和逆变换（ DFT 、 $IDFT$ ）的细节请参考<https://zhuanlan.zhihu.com/p/19763358> 和 <https://zhuanlan.zhihu.com/p/635646102>。而在 $StemGNN$ 模型的代码实现中，实际上是使用到了 FFT 和 $IFFT$ ，即快速傅里叶变换和逆变换，而关于 FFT 和 $IFFT$ 相关的具体细节说明请参考：

<https://www.bilibili.com/video/BV1za411F76U>

和 <https://www.bilibili.com/video/BV1mc411o7Bg>

简而言之，这是人类历史上最为精妙、且用途最为广泛的算法之一。

实验及结果:

1) Datasets:

本文一共使用了 10 个数据集，如下表所示:

Table 1: Summary of Datasets										
	METR-LA	PEMS-BAY	PEMS07	PEMS03	PEMS04	PEMS08	Solar	Electricity	ECG5000	COVID-19
# of nodes	207	325	228	358	307	170	137	321	140	25
# of timesteps	34,272	52,116	12,672	26,209	16,992	17,856	52,560	26,304	5,000	110
Granularity	5min	5min	5min	5min	5min	5min	10min	1hour	-	1day
Start time	9/1/2018	1/1/2018	7/1/2016	5/1/2012	7/1/2017	3/1/2012	1/1/2006	1/1/2012	-	1/22/2020

2) Results:

*StemGNN*与单变量时序预测模型*TCN*、*NBeats*；多变量时序预测模型*DCRNN*、*ST-GCN*等进行了比较。而在下表中我们可以看到*StemGNN*除了在*PEMS08*数据集上得到的*MAE*外均取得了最好的结果，同时多变量时序模型在多变量任务上也优于单变量模型。

	MAE	RMSE	MAPE(%)	MAE	RMSE	MAPE(%)	MAE	RMSE	MAPE(%)
	METR-LA [15]			PEMS-BAY [4]			PEMS07 [4]		
FC-LSTM [32]	3.44	6.3	9.6	2.05	4.19	4.8	3.57	6.2	8.6
SFM [39]	3.21	6.2	8.7	2.03	4.09	4.4	2.75	4.32	6.6
N-BEATS [24]	3.15	6.12	7.5	1.75	4.03	4.1	3.41	5.52	7.65
DCRNN [20]	2.77	5.38	7.3	1.38	2.95	2.9	2.25	4.04	5.30
LSTNet [17]	3.03	5.91	7.67	1.86	3.91	3.1	2.34	4.26	5.41
ST-GCN [38]	2.88	5.74	7.6	1.36	2.96	2.9	2.25	4.04	5.26
TCN [3]	2.74	5.68	6.54	1.45	3.01	3.03	3.25	5.51	6.7
DeepState [26]	2.72	5.24	6.8	1.88	3.04	2.8	3.95	6.49	7.9
GraphWaveNet [35]	2.69	5.15	6.9	1.3	2.74	2.7	-	-	-
DeepGLO [30]	2.91	5.48	6.75	1.39	2.91	3.01	3.01	5.25	6.2
StemGNN (ours)	2.56	5.06	6.46	1.23	2.48	2.63	2.14	4.01	5.01
	PEMS03 [4]			PEMS04 [4]			PEMS08 [4]		
FC-LSTM [32]	21.33	35.11	23.33	27.14	41.59	18.2	22.2	34.06	14.2
SFM [39]	17.67	30.01	18.33	24.36	37.10	17.2	16.01	27.41	10.4
N-BEATS [24]	18.45	31.23	18.35	25.56	39.9	17.18	19.48	28.32	13.5
DCRNN [20]	18.18	30.31	18.91	24.7	38.12	17.12	17.86	27.83	11.45
LSTNet [17]	19.07	29.67	17.73	24.04	37.38	17.01	20.26	31.96	11.3
ST-GCN [38]	17.49	30.12	17.15	22.70	35.50	14.59	18.02	27.83	11.4
TCN [3]	18.23	25.04	19.44	26.31	36.11	15.62	15.93	25.69	16.5
DeepState [26]	15.59	20.21	18.69	26.5	33.0	15.4	19.34	27.18	16
GraphWaveNet [35]	19.85	32.94	19.31	26.85	39.7	17.29	19.13	28.16	12.68
DeepGLO [30]	17.25	23.25	19.27	25.45	35.9	12.2	15.12	25.22	13.2
StemGNN (ours)	14.32	21.64	16.24	20.24	32.15	10.03	15.83	24.93	9.26
	Solar [17]			Electricity [2]			ECG [5]		
FC-LSTM [32]	0.13	0.19	27.01	0.62	0.2	24.39	0.32	0.54	31.0
SFM [39]	0.05	0.09	13.4	0.08	0.13	17.3	0.17	0.58	11.9
N-BEATS [24]	0.09	0.15	23.53	-	-	-	0.08	0.16	12.428
LSTNet [17]	0.07	0.19	19.13	0.06	0.07	14.97	0.08	0.12	12.74
TCN [3]	0.06	0.06	21.1	0.072	0.51	16.44	0.1	0.3	19.03
DeepState [26]	0.06	0.25	19.4	0.065	0.67	15.13	0.09	0.76	19.21
GraphWaveNet [35]	0.05	0.09	18.12	0.071	0.53	16.49	0.19	0.86	19.67
DeepGLO [30]	0.09	0.14	21.6	0.08	0.14	15.02	0.09	0.15	12.45
StemGNN (ours)	0.03	0.07	11.55	0.04	0.06	14.77	0.05	0.07	10.58

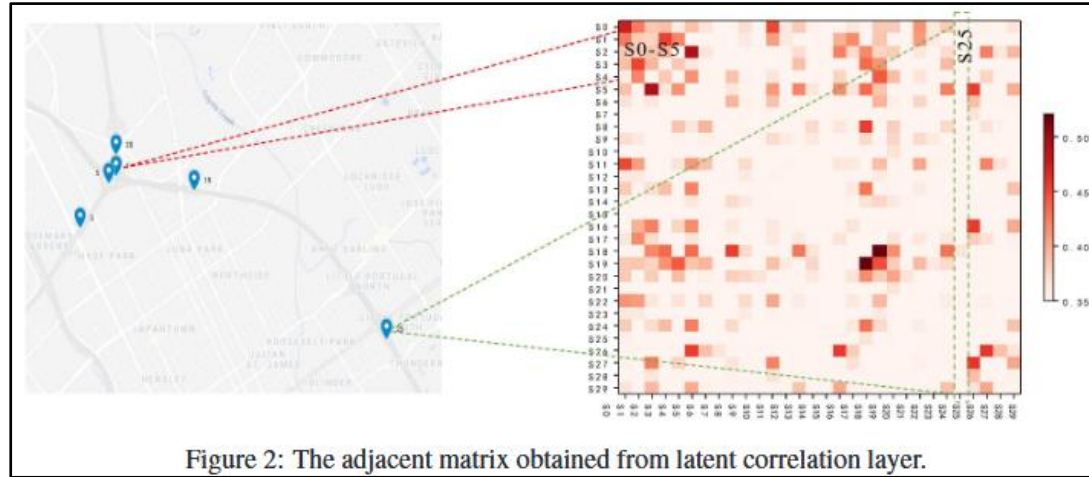
3) Ablation Study:

本文的消融实验验证了*StemGNN*的各个部分的有效性，可以看到没有时间提取*Spe-Seq*后的效果最差。

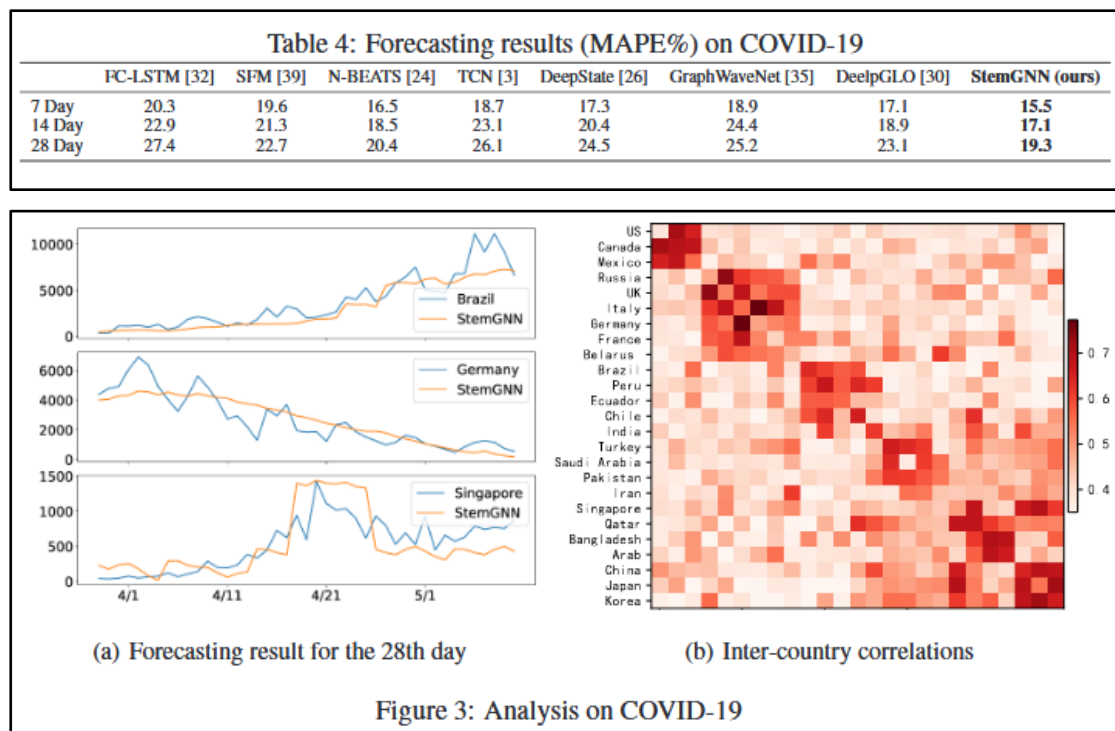
	StemGNN	w/o LC	w/o Spe-Seq Cell	w/o DFT	w/o GFT	w/o Residual	w/o Backcasting
MAE	2.144	2.158	2.612	2.299	2.237	2.256	2.203
RMSE	4.010	4.017	4.692	4.170	4.068	4.155	4.077
MAPE(%)	5.010	5.113	6.180	5.336	5.222	5.230	5.130

4) Analysis:

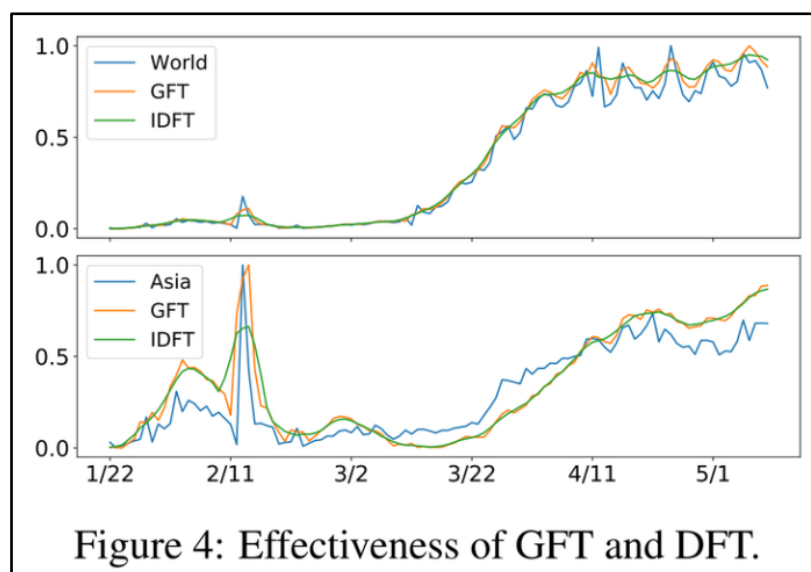
首先，本文在交通预测任务上对学习到的邻接矩阵进行了可视化分析，结果如下图所示：



然后，本文对 COVID-19 数据集进行了实验并可视化了邻接矩阵，结果如下图所示：



最后，本文还可可视化了 GFT 和 DFT 后的时间序列，可以看到频域上的序列更加平滑。



结论：

本文提出了一种新型的深度学习模型，即频谱时序图神经网络（*StemGNN*），通过在频谱域对多元时间序列进行联合建模，并有效利用了序列间相关性和时序依赖性的优势。此外，实验表明 $StemGNN$ 在各种多元时间序列预测应用中，始终优于现有的方法。

对本文的感悟：

时间序列分析是机器学习中的一个重要研究领域，而多元时间序列预测则是该领域中最常见的任务之一。总的来说， $StemGNN$ 是将离散傅里叶变换与图神经网络相结合的首次尝试。此外， $StemGNN$ 还在端到端框架中采用了潜在相关层以自动学习多元信号之间的关系，使得 $StemGNN$ 成为了一种更加通用的方法，可广泛应用于交通流监控、医疗数据监测、自然灾害预测和经济等领域。但是，在 $StemGNN$ 模型的实际开源代码中，有些代码实现并没有体现出如论文中所描述的结构和作用，并且一些代码也似乎存在着诸多明显的问题。然而，作者也并没有给出进一步的详细解释或者修改，这值得引起我们的注意。