

BEATS: NEURAL BASIS EXPANSION ANALYSIS FOR INTERPRETABLE TIME SERIES FORECASTING

Boris N. Oreshkin
Element AI
boris.oreshkin@gmail.com

Dmitri Carпов
Element AI
dmitri.carpov@elementai

Joaquin V. Chapados
Element AI
chapados@elementai.com

Yoshua Bengio
Mila
yoshua.bengio@mila.quebec

【深度学习 112】时间序列预测



张楚琦

清华大学 交叉信息院博士在读

43 人赞同了该文章

本周张天平学弟在组会上讲了两篇时间序列预测上的最新文章，其中一篇文章用到了 CV 领域非常有意思的一个工作。

原文传送门

N-BEATS (ICLR 2020) : [Oreshkin, Boris N., et al. "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting." arXiv preprint arXiv:1905.10437 \(2019\).](#)

Saliency detection (CVPR 2007) : [Hou, Xiaodi, and Liqing Zhang. "Saliency detection: A spectral residual approach." 2007 IEEE Conference on computer vision and pattern recognition. IEEE, 2007.](#)

Time series anomaly detection (KDD 2019) : [Ren, Hansheng, et al. "Time-Series Anomaly Detection Service at Microsoft." Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019.](#)

特色

第一篇 paper 是 ICLR 2020 的文章，讲了一个纯深度学习的模型，用来预测时间序列，在竞赛上有较好的结果。同时，具有一定的可解释性。比较有意思的是第二篇文章，通过几行代码就可以识别出来一个图片中最显著的关注点位置，这篇文章是 Xiaodi Hou 在本科的时候做出来的，现在引用已经超过了 3000。第三篇文章是 KDD 2019 的文章，主要用来做时间序列中的异常点检测，用到了第二篇文章的技术。

过程

1、N-BEATS

神经网络结构

神经网络的结构如下，输入的是过去一段时间的数据，预测未来一段时间的数据。模型由若干个 stack 组成，各个 stack 的结果加和起来得到最后的预测结果；每个 stack 又由若干个 block 组成，每个 block 会向前和向后预测，向前预测的结果会加和起来得到最后的结果，向后预测的结果用于和原始信号相减，然后给下一个 block 使用。我的理解是，通过这样的方法可以先预测比较明显的 pattern，减去之后再去预测另外的 pattern。

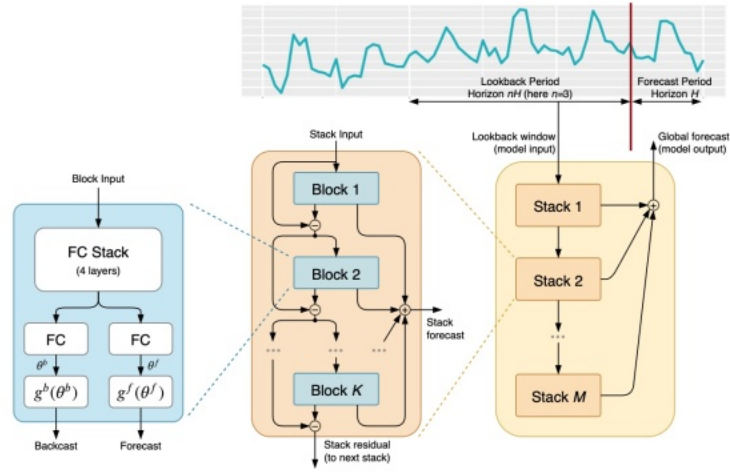


Figure 1: Proposed architecture. The basic building block is a multi-layer FC network with RELU nonlinearities. It predicts basis expansion coefficients both forward, θ^f , (forecast) and backward, θ^b , (backcast). Blocks are organized into stacks using doubly residual stacking principle. A stack may have layers with shared g^b and g^f . Forecasts are aggregated in hierarchical fashion. This enables building a very deep neural network with interpretable outputs.

每个 block 内的具体的计算过程如下：

$$\begin{aligned} \mathbf{h}_{\ell,1} &= \text{FC}_{\ell,1}(\mathbf{x}_{\ell}), \quad \mathbf{h}_{\ell,2} = \text{FC}_{\ell,2}(\mathbf{h}_{\ell,1}), \quad \mathbf{h}_{\ell,3} = \text{FC}_{\ell,3}(\mathbf{h}_{\ell,2}), \quad \mathbf{h}_{\ell,4} = \text{FC}_{\ell,4}(\mathbf{h}_{\ell,3}). \\ \theta_{\ell}^b &= \text{LINEAR}_{\ell}^b(\mathbf{h}_{\ell,4}), \quad \theta_{\ell}^f = \text{LINEAR}_{\ell}^f(\mathbf{h}_{\ell,4}). \end{aligned} \quad (1)$$

接下来，forecast 和 backcast 的结果是预测出来的系数 θ ，再结合一个基底 \mathbf{v} 就可以得到最后的结果。

$$\hat{\mathbf{y}}_{\ell} = \sum_{i=1}^{\dim(\theta_{\ell}^f)} \theta_{\ell,i}^f \mathbf{v}_i^f, \quad \hat{\mathbf{x}}_{\ell} = \sum_{i=1}^{\dim(\theta_{\ell}^b)} \theta_{\ell,i}^b \mathbf{v}_i^b.$$

可解释性

基底 v 可以学习，也可以规定成相应的反映比如趋势或者周期性的基底，这样就具有一定的可解释性。（什么叫可解释性？感觉这种所谓的可解释性比较弱啊）

Trend model. A typical characteristic of trend is that most of the time it is a monotonic function, or at least a slowly varying function. In order to mimic this behaviour we propose to constrain $g_{s,\ell}^b$ and $g_{s,\ell}^f$ to be a polynomial of small degree p , a function slowly varying across forecast window:

$$\hat{\mathbf{y}}_{s,\ell} = \sum_{i=0}^p \theta_{s,\ell,i}^f t^i. \quad (2)$$

Here time vector $\mathbf{t} = [0, 1, 2, \dots, H-2, H-1]^T / H$ is defined on a discrete grid running from 0 to $(H-1)/H$, forecasting H steps ahead. Alternatively, the trend forecast in matrix form will then be:

$$\hat{\mathbf{y}}_{s,\ell}^{tr} = \mathbf{T} \theta_{s,\ell}^f,$$

where $\theta_{s,\ell}^f$ are polynomial coefficients predicted by a FC network of layer ℓ of stack s described by equations (1); and $\mathbf{T} = [\mathbf{1}, \mathbf{t}, \dots, \mathbf{t}^p]$ is the matrix of powers of \mathbf{t} . If p is low, e.g. $p=3$, it forecasts $\hat{\mathbf{y}}_{s,\ell}^{tr}$ to mimic trend.

Seasonality model. Typical characteristic of seasonality is that it is a regular, cyclical, recurring fluctuation. Therefore, to model seasonality, we propose to constrain $g_{s,\ell}^b$ and $g_{s,\ell}^f$ to belong to the class of periodic functions, i.e. $y_t = y_{t-\Delta}$, where Δ is a seasonality period. A natural choice for the basis to model periodic function is the Fourier series:

$$\hat{\mathbf{y}}_{s,\ell} = \sum_{i=0}^{\lfloor H/2-1 \rfloor} \theta_{s,\ell,i}^f \cos(2\pi i t) + \theta_{s,\ell,i+\lfloor H/2 \rfloor}^f \sin(2\pi i t), \quad (3)$$

The seasonality forecast will then have the matrix form as follows:

$$\hat{\mathbf{y}}_{s,\ell}^{seas} = \mathbf{S} \theta_{s,\ell}^f,$$

where $\theta_{s,\ell}^f$ are Fourier coefficients predicted by a FC network of layer ℓ of stack s described by equations (1); and $\mathbf{S} = [\mathbf{1}, \cos(2\pi \mathbf{t}), \dots, \cos(2\pi \lfloor H/2-1 \rfloor \mathbf{t}), \sin(2\pi \mathbf{t}), \dots, \sin(2\pi \lfloor H/2-1 \rfloor \mathbf{t})]$ is the matrix of sinusoidal waveforms. The forecast $\hat{\mathbf{y}}_{s,\ell}^{seas}$ is then a periodic function mimicking typical seasonal patterns.

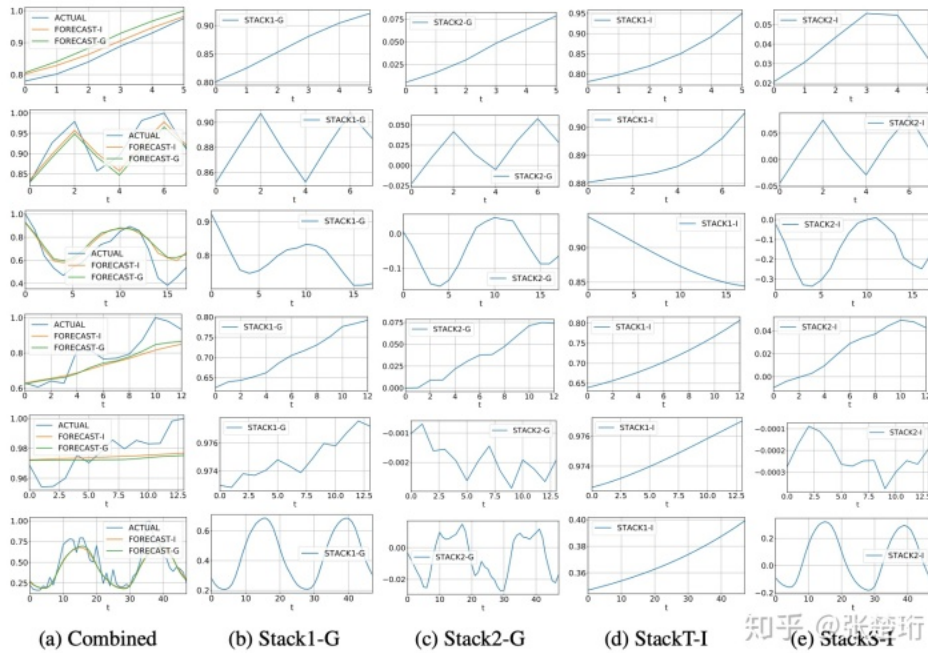
ps. 最后还需要做 ensembling, 确实我们在实验中也发现对于时间序列的预测, 特别是金融里面, ensembling 的效果提升是比较大的。Ensemble 中不同的模型使用不同的 loss function、horizon 和样本 (bagging)。不同的 loss function 包括:

$$\text{sMAPE} = \frac{200}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{|y_{T+i}| + |\hat{y}_{T+i}|}, \quad \text{MAPE} = \frac{100}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{|y_{T+i}|},$$

$$\text{MASE} = \frac{1}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{\frac{1}{T+H-m} \sum_{j=m+1}^{T+H} |y_j - y_{j-m}|}, \quad \text{OWA} = \frac{1}{2} \left[\frac{\text{sMAPE}}{\text{sMAPE}_{\text{Naive2}}} + \frac{\text{MASE}}{\text{MASE}_{\text{Naive2}}} \right].$$

实验结果

这里主要看一下这个模型不同 stack 的作用是什么，其中标记为 G 的模型基底是可以学习的，而标记为 I 的模型是指定的基底，这样具有一定的可解释性。可以看到 I 模型中可以现实地要求学习出来各个时间序列的趋势和周期。



2、Saliency detection

显著性检查 (saliency detection) 的目标是，给定一张图片，输出这个图片中拿一个部分比较有意思。这个任务比较符合人的视觉思维，把一张图片展示给人看的时候，人通常会注意力集中在图

片的特定的一些点上。这里就是想找出图片中的哪些地方比较具有吸引力。

这篇文章从信息论的角度，认为一张图片的信息可以被分为两个部分，即先验信息和这个图片特定的信息。

$$H(\text{Image}) = H(\text{Innovation}) + H(\text{Prior Knowledge}),$$

对于图像来说，给定一个频率 f ，可以计算得到图像在该频率下的频谱强度 $L(f)$ ，把许多图片的频谱强度函数做平均可以得到 $A(f)$ 。由于图片具有缩放不变性，因此对于所有图片的平均来讲，这个频谱满足一定的关系

$$E\{A(f)\} \propto 1/f. \quad (1)$$

但是对于特定的图片来说，只是大致趋势上差不多，但是会有一些细节上的特别之处。

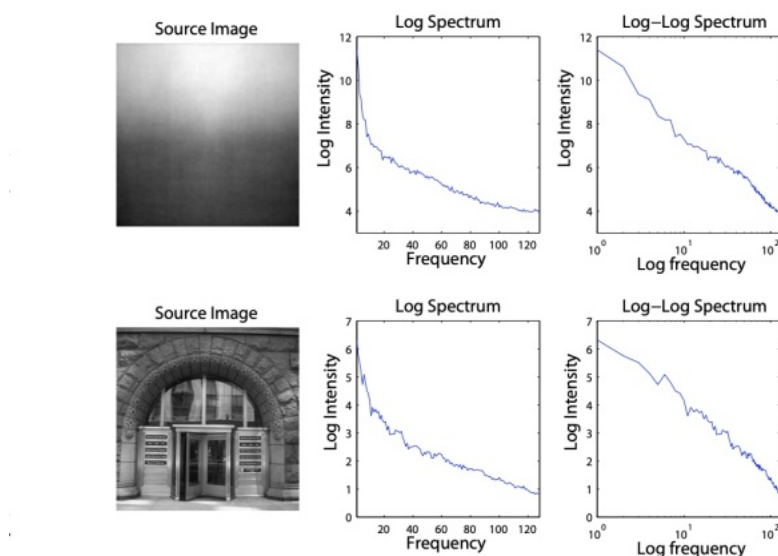


Figure 1. Examples of log spectrum and log-log spectrum. The first image is the average of 2277 natural images. 知乎 @张楚珩

第一行的图片是多张图片的平均，因此 Log Spectrum 是比较平滑的，符合上述规律；而对于特定的图片，Log Spectrum 具有一些起伏。

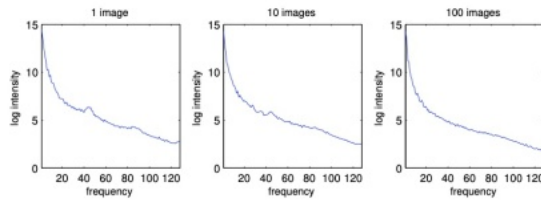


Figure 3. Curves of averaged spectra over 1, 10 and 100 images.

知乎 @张楚珩

类似地，对于单张图片，Log Spectrum 上有一些突起；而平均之后就是一条比较平滑的曲线。

把这些特别之处定位出来，再利用傅里叶逆变换就可以锁定图片中比较有意思的部分。文章提出计算 spectral residual $\mathcal{R}(f)$ ，它是单张图片的频谱函数 $\mathcal{L}(f)$ 和所有图片的平均 $\mathcal{A}(f)$ 的差

$$\mathcal{R}(f) = \mathcal{L}(f) - \mathcal{A}(f). \quad (4)$$

它也可以看做是已知图片先验之后，再见到这张图片所提供的信息

$$H(\mathcal{R}(f)) = H(\mathcal{L}(f)|\mathcal{A}(f)), \quad (2)$$

另外文章还提出平均谱 $\mathcal{A}(f)$ 也不需要根据所有图片取平均，而是直接对给定的这一张图片做平滑即可。

$$\mathcal{A}(f) = h_n(f) * \mathcal{L}(f), \quad (3)$$

where $h_n(f)$ is an $n \times n$ matrix defined by:

$$h_n(f) = \frac{1}{n^2} \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix}$$

知乎 @张楚珩

得到 saliency map 的过程总结如下。对于一张图片，做傅里叶变换，得到相应的振幅和相位。把振幅减去该图片平均之后的振幅，然后加上相位做傅里叶逆变换。为了看起来更舒服，对于最后的 saliency map 还用 $g(x)$ 做了一定的平滑。

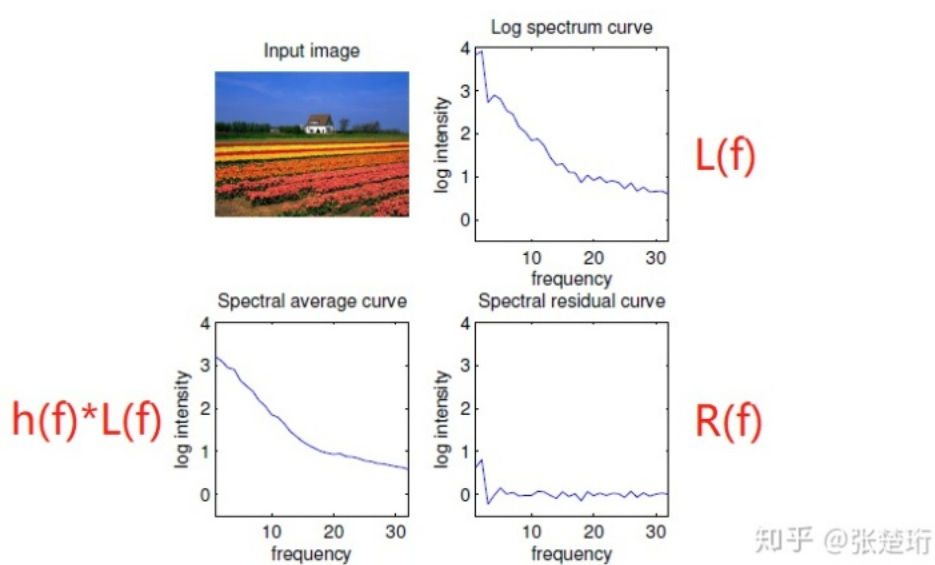
$$\mathcal{A}(f) = \Re(\mathfrak{F}[\mathcal{I}(x)]), \quad (5)$$

$$\mathcal{P}(f) = \Im(\mathfrak{F}[\mathcal{I}(x)]), \quad (6)$$

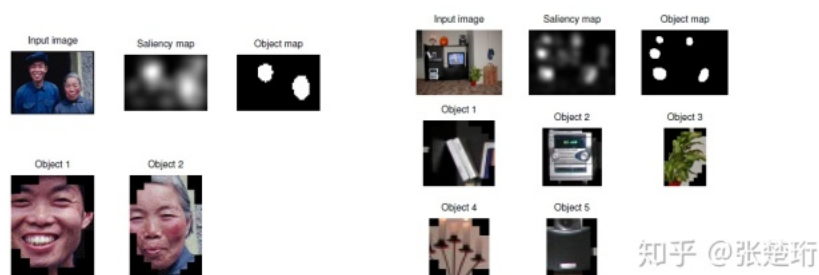
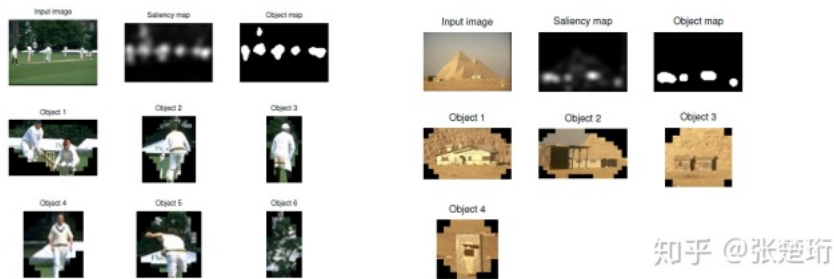
$$\mathcal{L}(f) = \log(\mathcal{A}(f)), \quad (7)$$

$$\mathcal{R}(f) = \mathcal{L}(f) - h_n(f) * \mathcal{L}(f), \quad (8)$$

$$\mathcal{S}(x) = g(x) * \mathfrak{F}^{-1} \left[\exp(\mathcal{R}(f) + \mathcal{P}(f)) \right]^2 \quad \text{知乎 @张楚珩}$$



文章后面还讲了如何把 saliency map 转化为 object map。

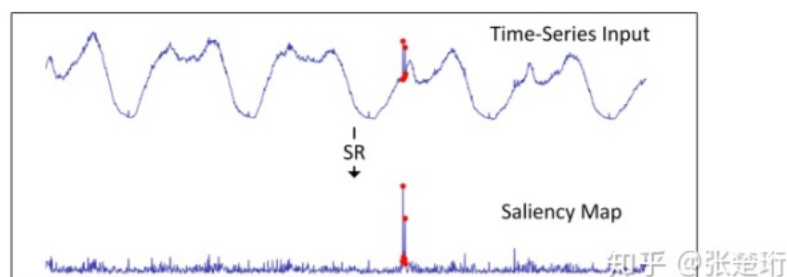


3、Time series anomaly detection

这篇文章号称是第一个把 spectral residual 应用到时间序列异常检测。现实中时间序列异常检测遇到如下困难

- Lack of labels: 数据很多，但是异常点的标签很少。
- Generalization: 需要监测的时间序列数据类型很多，希望模型能适用各种时间序列数据。
- Efficiency: 由于是在时间序列上做异常监测，因此需要实时给出反馈，而不能用特别复杂的模型。

文章的做法就是在时间序列上计算 saliency map。



$$A(f) = \text{Amplitude}(\mathcal{F}(\mathbf{x})) \quad (1)$$

$$P(f) = \text{Phrase}(\mathcal{F}(\mathbf{x})) \quad (2)$$

$$L(f) = \log(A(f)) \quad (3)$$

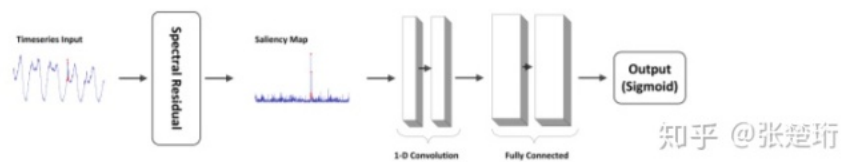
$$AL(f) = h_q(f) \cdot L(f) \quad (4)$$

$$R(f) = L(f) - AL(f) \quad (5)$$

$$S(\mathbf{x}) = \|\mathcal{F}^{-1}(\exp(R(f) + iP(f)))\|$$

知乎 @张楚珩

不过最后把哪个点判断为异常点呢？这里文章训练一个 CNN 来做判断。训练数据是人造的数据，并且人为加入异常点，CNN 的输入就是 saliency map 这个时间序列，输出就是在异常点进入的时候给出相应的信号。



编辑于 2020-03-16

机器学习

时间序列分析

深度学习 (Deep Learning)

赞同 43



1 条评论

分享

喜欢

收藏



文章被以下专栏收录



强化学习前沿
读呀读paper

进入专栏