Deep Learning for Time Series

Session 3: Attention-based architectures

Romain Tavenard

Intro

Self-attention vs RNNs vs ConvNets

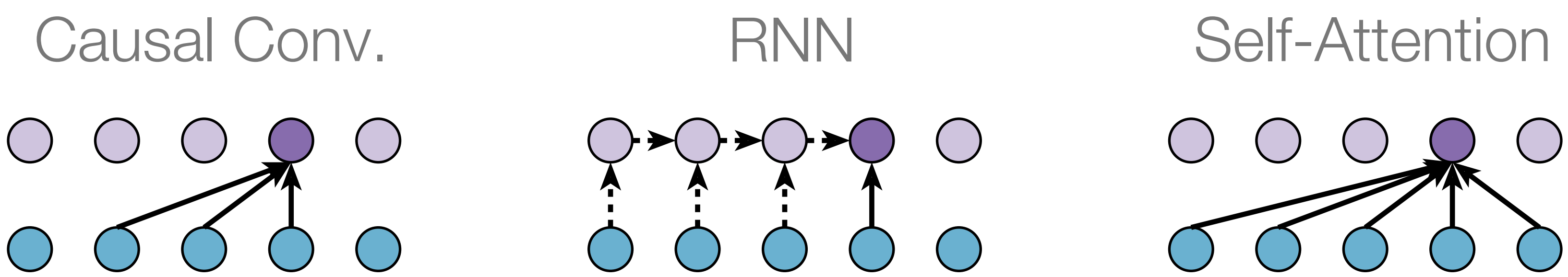
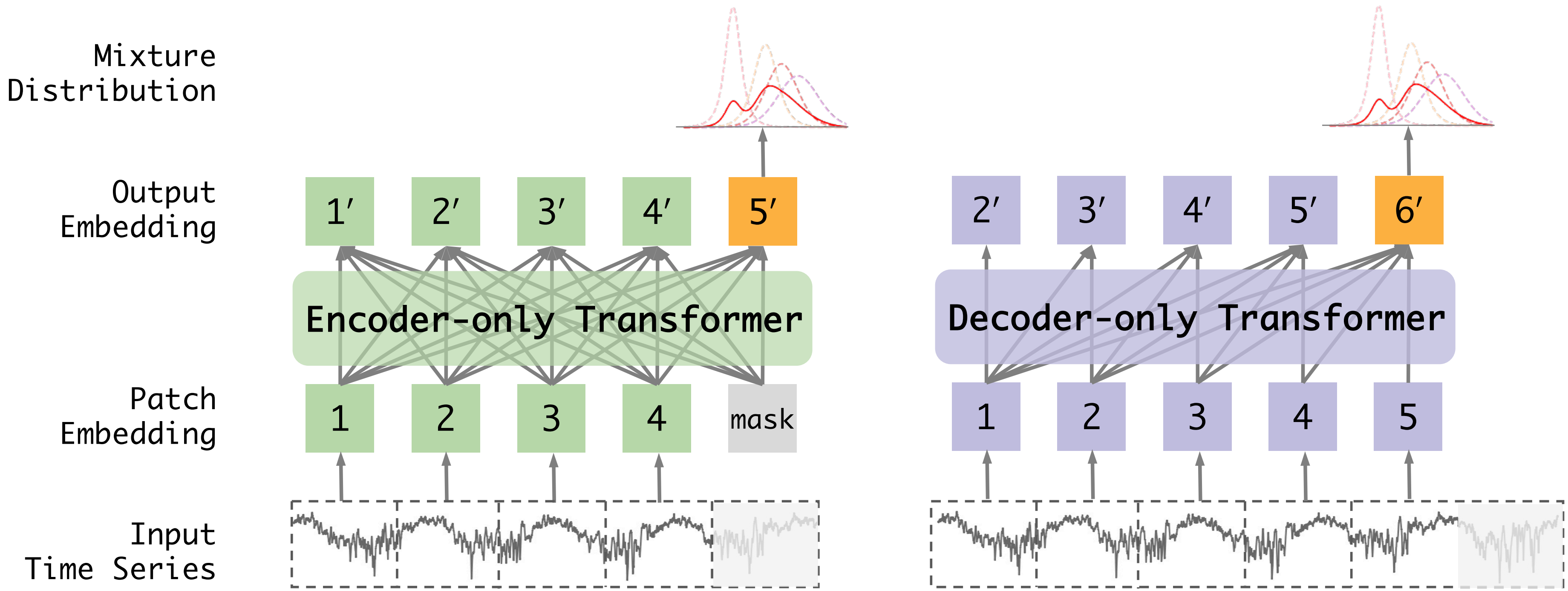


Illustration of self-attention

Basic transformer architecture

Encoder-only vs Decoder-only for forecasting



Source: "Towards Neural Scaling Laws for Time Series Foundation Models" (ICLR'25)

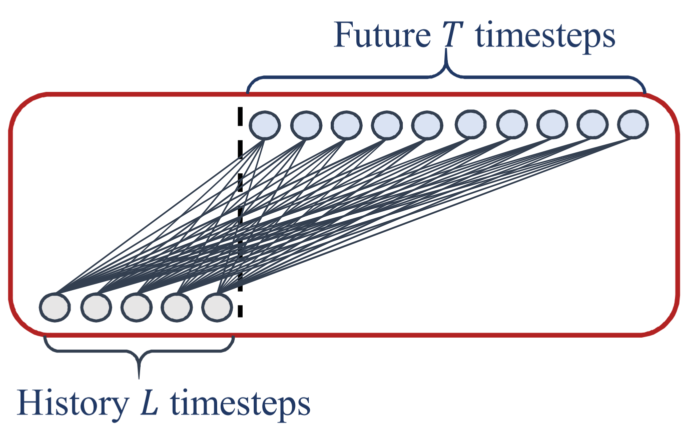
- Encoder-only (bi-directional attention) is more versatile (can be applied to different tasks)
- Decoder-only is more suitable for forecasting

Failures of transformers in forecasting

Are Transformers Effective for Time Series Forecasting?

Ailing Zeng[1*], Muxi Chen[1*], Lei Zhang[2], Qiang Xu[1]
[1]The Chinese University of Hong Kong
[2]International Digital Economy Academy (IDEA)
{alzeng, mxchen21, qxu}@cse.cuhk.edu.hk
{leizhang}@idea.edu.cn

- Comparison between simple linear predictor and SOTA transformers (at the time)



Source: “Are Transformers Effective for Time Series Forecasting?” (AAAI’23)

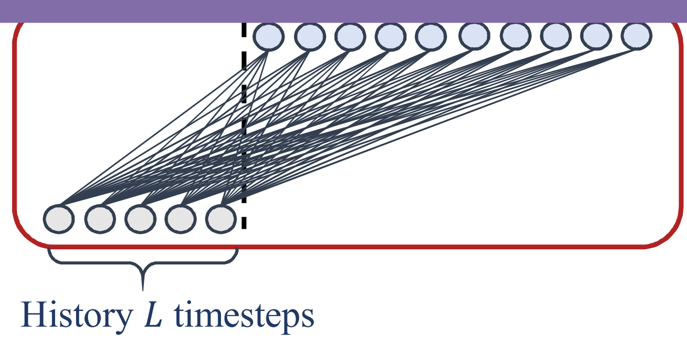
Failures of transformers in forecasting

Are Transformers Effective for Time Series Forecasting?

Ailing Zeng[1*], Muxi Chen[1*], Lei Zhang[2], Qiang Xu[1]
[1]The Chinese University of Hong Kong
[2]International Digital Economy Academy (IDEA)
{alzeng, mxchen21, qxu}@cse.cuhk.edu.hk

“LTSF-Linear surprisingly outperforms existing sophisticated Transformer-based LTSF models in all cases, and often by a large margin”

- Compa TA transfor



Source: “Are Transformers Effective for Time Series Forecasting?” (AAAI’23)

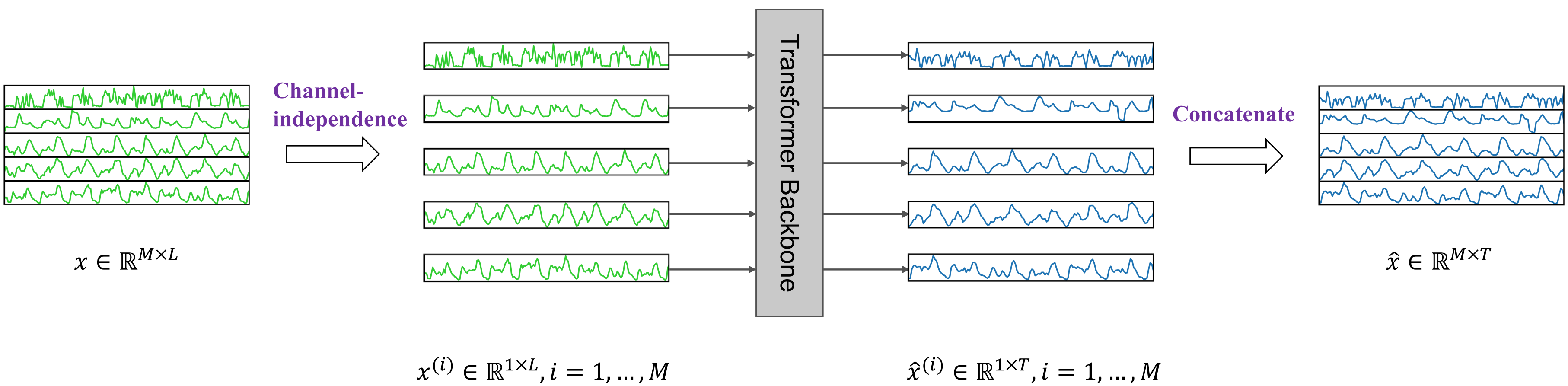
PatchTST: a transformer that works well

- Previous methods:
  1. each patch = 1 time point or some extracted information
  2. all variates are considered together
- PatchTST:
  1. patch = subseries
  2. channel-independence

     (bonus: much easier for transfer learning)
  3. proper preprocessing (RevIN)

PatchTST: a transformer that works well

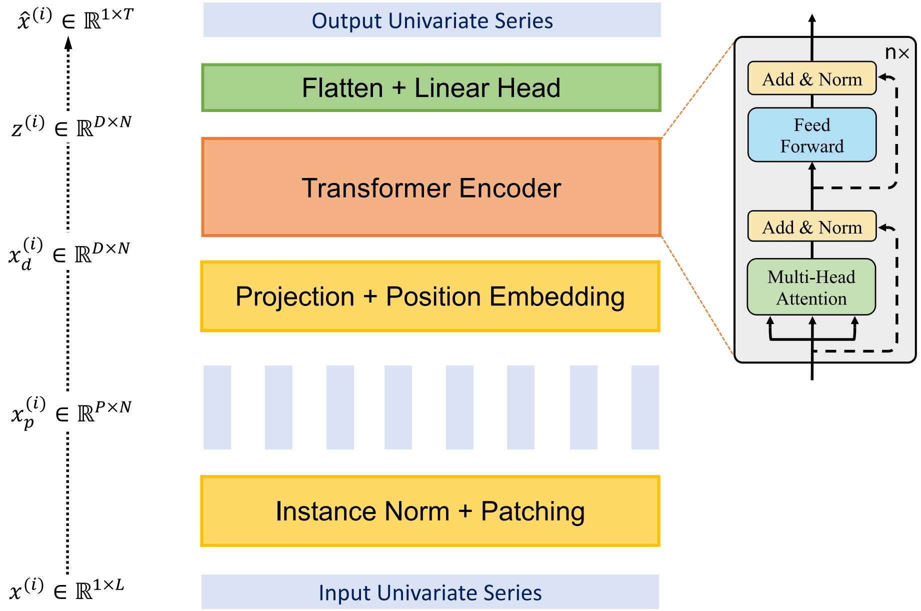
Channel independence



Source: "A Time Series is Worth 64 Words: Long-term Forecasting with Transformers" (ICLR'23)

PatchTST: a transformer that works well

Transformer backbone

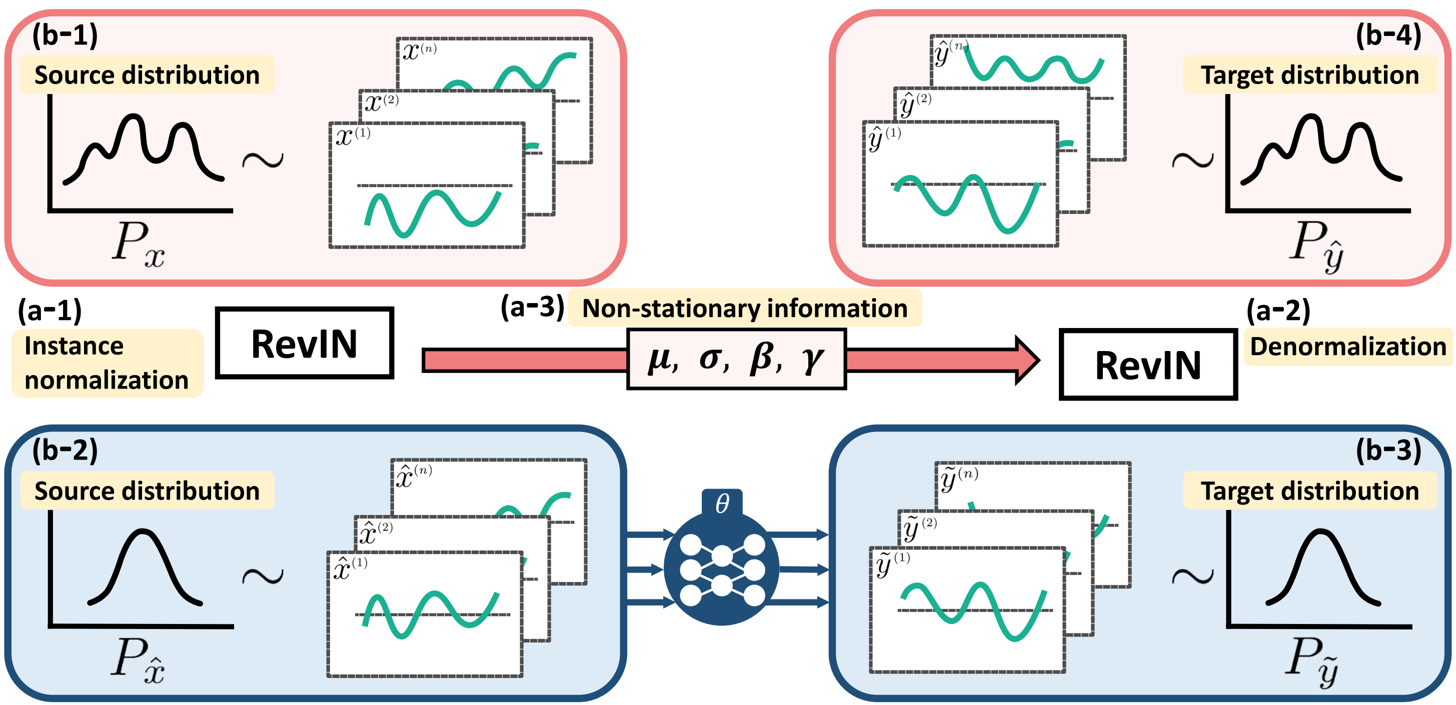


Source: “A Time Series is Worth 64 Words: Long-term Forecasting with Transformers” (ICLR’23)

PatchTST: a transformer that works well

Preprocessing (RevIN)

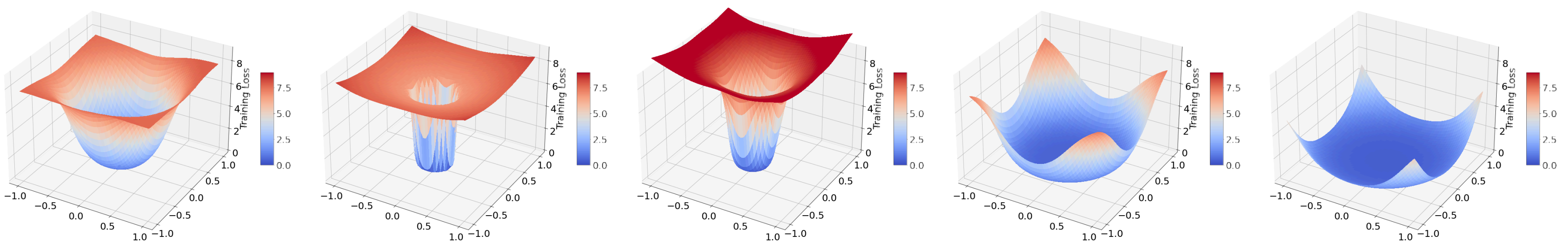
$$\hat{x}^{(i)}_{d,t} = \gamma_d \cdot \frac{x^{(i)}_{d,t} - \hat{\mu}^{(i)}_d}{\hat{\sigma}^{(i)}_d} + \beta_d \quad \text{and reverse before forecasting}$$



Source: "Reversible Instance Normalization for Accurate Time-Series Forecasting against Distribution Shift" (ICLR'22)

More insights on transformer failures

- Transformers have a sharp loss landscape:



(a) ResNet (b) ViT (c) Mixer (d) ViT-SAM (e) Mixer-SAM

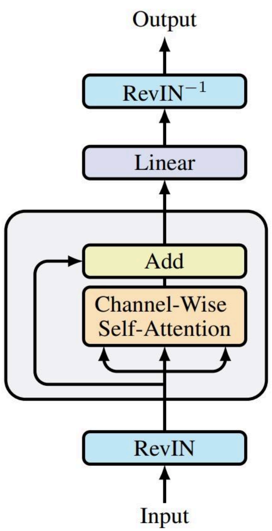
Source: “When Vision Transformers Outperform ResNets without Pre-training or Strong Data Augmentations” (ICLR’22)

- Train with SAM (Sharpness-Aware Minimization) to smooth the landscape:

$$\mathcal{L}_{\text{SAM}} = \max_{\|\varepsilon\|_2 \leq \rho} \mathcal{L}(w + \varepsilon)$$

More insights on transformer failures

- SAMformer: a forecasting Transformer trained with SAM



Source: “SAMformer: […]” (ICML’24)

Conclusion

- Transformers can work well for time series forecasting
- But need proper design choices:
  1. patch-based input
  2. channel-independence
  3. proper preprocessing (RevIN)
  4. smooth optimization (SAM)