Designing Transformer networks for sparse recovery of sequential data using deep unfolding

Brent De Weerdt, Yonina C. Eldar, Nikos Deligiannis

ETRO
ELECTRONICS & INFORMATICS

Use sparse priors to recover signals from compressed measurements

- Compressed measurement: $\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \eta_t, \quad \mathbf{A} \in \mathbb{R}^{m \times n}\,(m \ll n), t = 1, \ldots, T$
- Assume a sparse representation $\mathbf{h}_t$ in some dictionary: $\mathbf{s}_t = \mathbf{D}\mathbf{h}_t$
- Assume some correlation over time: $C(\mathbf{h}_t, \mathbf{h}_{t-1})$
- Solve $\min_{\mathbf{h}_1,\ldots,\mathbf{h}_T} \sum_t \left(\frac{1}{2}\|\mathbf{x}_t - \mathbf{A}\mathbf{D}\mathbf{h}_t\|_2^2 + \lambda_1\|\mathbf{h}_t\|_1 + \lambda_2 C(\mathbf{h}_t, \mathbf{h}_{t-1})\right)$
- The final reconstructed signal is $\mathbf{s}_t^* = \mathbf{D}\mathbf{h}_t^*$

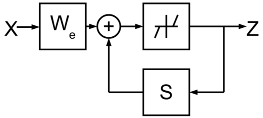
Deep unfolding Intro

- Deep unfolding designs neural network models by:
    1. Unrolling an iterative algorithm
    2. Mapping the algorithm's (sub)steps to neural network layers
    3. Training the resulting model on data
- Deep unfolding models have lower reconstruction errors and less iterations than the original iterative algorithm

LISTA

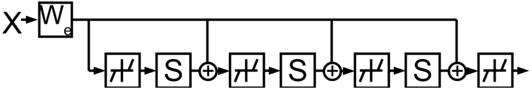
- Optimization problem: $\min_{\mathbf{h}} \frac{1}{2}\|\mathbf{x} - \mathbf{ADh}\|_2^2 + \lambda\|\mathbf{h}\|_1$
- Iterative Soft Thresholding Algorithm (ISTA):

$$\mathbf{h}^{(k+1)} = \phi_{\lambda/c}\left(\mathbf{h}^{(k)} + \frac{1}{c}\mathbf{D}^T\mathbf{A}^T\left(\mathbf{x} - \mathbf{ADh}^{(k)}\right)\right)$$



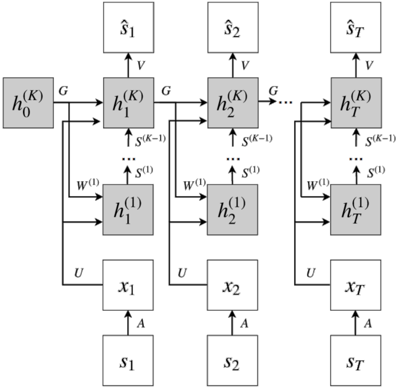
- Deep unfolding model: Learned ISTA (LISTA)

$$\mathbf{h}^{(k+1)} = \phi_{\lambda/c}\left(\mathbf{Sh}^{(k)} + \mathbf{Wx}\right)$$



Gregor and LeCun, "Learning fast approximations of sparse coding," *ICML*, 2010.

Deep unfolding RNNs



- SISTA-RNN:

$$\sum_t \left(\frac{1}{2}\|\mathbf{x}_t - \mathbf{ADh}_t\|_2^2 + \lambda_1\|\mathbf{h}_t\|_1 + \frac{\lambda_2}{2}\|\mathbf{Dh}_t - \mathbf{FDh}_{t-1}\|_2^2\right)$$

- ℓ_1-ℓ_1-RNN

$$\sum_t \left(\frac{1}{2}\|\mathbf{x}_t - \mathbf{ADh}_t\|_2^2 + \lambda_1\|\mathbf{h}_t\|_1 + \lambda_2\|\mathbf{h}_t - \mathbf{Gh}_{t-1}\|_1\right)$$

- Reweighted-RNN

$$\sum_t \left(\frac{1}{2}\|\mathbf{x}_t - \mathbf{ADZh}_t\|_2^2 + \lambda_1\|\mathbf{g} \circ \mathbf{Zh}_t\|_1 + \lambda_2\|\mathbf{g} \circ (\mathbf{Zh}_t - \mathbf{Gh}_{t-1})\|_1\right)$$

Wisdom et al., "Building recurrent networks by unfolding iterative thresholding for sequential sparse recovery," *ICASSP*, 2017.

Le et al., "Designing Recurrent Neural Networks by Unfolding an L1-L1 Minimization Algorithm," *ICIP*, 2019.

Luong et al., "Designing Interpretable Recurrent Neural Networks for Video Reconstruction via Deep Unfolding," *IEEE Trans. Img. Process.*, 2021.

Deep unfolding for a vanilla Transformer

Optimization problem designed to unfold into a Transformer architecture:

$$\mathbf{Y} = \left[\mathbf{y}_1, \ldots, \mathbf{y}_N\right], \ \psi(u) = \begin{cases} +\infty & \text{if } u < 0 \\ 0 & \text{if } u \geq 0 \end{cases}$$

$$\min_{\mathbf{Y}} \underbrace{\sum_{i,j} - \exp\left(-\frac{1}{2}\|\mathbf{W}_a\mathbf{y}_i - \mathbf{W}_a\mathbf{y}_j\|_2^2\right) + \frac{1}{2}\|\mathbf{W}_a\mathbf{Y}\|_{\mathcal{F}}^2}_{\text{softmax self-attention}} + \underbrace{\frac{1}{2}\operatorname{Tr}\left(\mathbf{Y}^T\mathbf{W}_b\mathbf{Y}\right) + \frac{1}{2}\|\mathbf{Y}\|_{\mathcal{F}}^2 + \psi(\mathbf{Y})}_{\text{linear layer + ReLU}}$$

- Design minimization steps for each part separately
- Alternating between these two steps minimizes the total optimization problem:

$$\mathbf{Y}^{(k+1)} = \operatorname{ReLU}\left(\mathbf{W}_b\mathbf{Y}^{(k)} \operatorname{softmax}_\beta\left(\mathbf{Y}^{(k)T}\mathbf{W}_a\mathbf{Y}^{(k)}\right)\right)$$

Yang et al., “Transformers from an Optimization Perspective,” *NeurIPS*, 2022.

Our deep unfolding Transformer for sparse recovery

- Incorporate priors for sequential sparse recovery
  - Model correlations across the whole video
  - Retain the sparsity constraint and data fidelity term

$$\min_{\mathbf{h}_1,\ldots,\mathbf{h}_T} \sum_t \underbrace{\lambda_2 \left(\sum_\tau -\exp\left(-\frac{1}{2} \|\mathbf{D}\mathbf{h}_t - \mathbf{D}\mathbf{h}_\tau\|_2^2 \right) + \|\mathbf{D}\mathbf{h}_t\|_2^2 \right)}_{\text{temporal correlations}} + \underbrace{\frac{1}{2} \|\mathbf{x}_t - \mathbf{A}\mathbf{D}\mathbf{h}_t\|_2^2 + \lambda_1 \|\mathbf{h}_t\|_1}_{\text{data fidelity and sparsity}}$$

The optimization algorithm

$$\min_{\mathbf{h}_1,\ldots,\mathbf{h}_T} \sum_t \underbrace{\lambda_2 \left(\sum_\tau -\exp\left(-\frac{1}{2} \|\mathbf{D}\mathbf{h}_t - \mathbf{D}\mathbf{h}_\tau\|_2^2 \right) + \|\mathbf{D}\mathbf{h}_t\|_2^2 \right)}_{\text{temporal correlations}} + \underbrace{\frac{1}{2} \|\mathbf{x}_t - \mathbf{A}\mathbf{D}\mathbf{h}_t\|_2^2 + \lambda_1 \|\mathbf{h}_t\|_1}_{\text{data fidelity and sparsity}}$$

- First part: softmax self-attention

$$\mathbf{H}^{(k+\frac{1}{2})} = \lambda_2 \mathbf{H}^{(k)} \,\text{softmax}_\beta \left(\mathbf{H}^{(k)T} \mathbf{D}^T \mathbf{D} \mathbf{H}^{(k)} \right), \quad \mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \ldots & \mathbf{h}_T \end{bmatrix}$$

- Second part: parallel ISTA operations

$$\mathbf{h}_t^{(k+1)} = \phi_{\lambda_1/c} \left(\mathbf{h}_t^{(k+\frac{1}{2})} + \frac{1}{c} \mathbf{D}^T \mathbf{A}^T \left(\mathbf{x}_t - \mathbf{A}\mathbf{D}\mathbf{h}_t^{(k+\frac{1}{2})} \right) \right) \ \forall t$$

DUST: Deep Unfolding Sparse Transformer

- Start from:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t,\ \mathbf{h}_t^{(0)} = 0 \quad \forall t$$

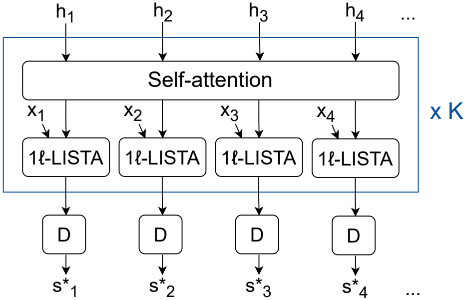
- For K times:

$$\mathbf{H}^{(k+\frac{1}{2})} = \lambda_2 \mathbf{H}^{(k)}\,\mathrm{softmax}\left(\mathbf{H}^{(k)T}\mathbf{D}^T\mathbf{D}\mathbf{H}^{(k)}\right)$$

$$\mathbf{h}_t^{(k+1)} = \phi_{\lambda_1/c}\left(\mathbf{U}\mathbf{h}_t^{(k+\frac{1}{2})} + \mathbf{V}\mathbf{x}_t\right) \quad \forall t$$

- Final reconstruction:

$$\mathbf{s}_t^* = \mathbf{D}\mathbf{h}_t^{(K)}$$



Experimental results

Average video reconstruction quality (PSNR) on the Avenue, UCSD and ShanghaiTech dataset.

	Avenue	UCSD	ST
SISTA-RNN	35.73	34.13	34.90
ℓ_1-ℓ_1-RNN	36.51	34.34	35.56
Reweighted-RNN	<u>36.94</u>	<u>35.22</u>	**36.03**
ViT	36.04	34.79	35.91
Unfolded Transformer	34.36	32.94	34.25
DUST (proposed)	**37.61**	**35.98**	<u>35.94</u>

Average video reconstruction quality (PSNR) on the Avenue dataset for different compression rates.

	50%	40%	30%	10%
SISTA-RNN	41.89	39.92	37.99	32.01
ℓ_1-ℓ_1-RNN	42.86	40.90	38.89	32.98
Reweighted-RNN	<u>43.23</u>	<u>41.16</u>	<u>39.12</u>	<u>33.88</u>
ViT	39.53	38.28	37.12	33.85
Unfold. Transf.	39.66	37.93	36.07	32.11
DUST (proposed)	**43.32**	**41.47**	**39.67**	**34.71**

Model size and computation complexity

- DUST and the other Transformer models can process videos twice as fast compared to the deep unfolding RNNs
  - More parallel computation
  - Less complex calculations
- DUST has 1.4M parameters, significantly smaller the next best performing model, reweighted-RNN (2.5M parameters)

Conclusion

- We designed a deep unfolding Transformer architecture for sparse recovery of sequential data
- This model has improved reconstruction quality and lower computational cost compared to deep unfolding RNNs
- Future work: different attention mechanisms, longer sequences, denoising, super-resolution