Designing RNNs for Explainability

Pattarawat Chormai
Supervised by Dr. Grégoire Montavon and Prof. Klaus-Robert Müller
Department of Machine Learning
This work is a part of master's thesis [4]. Contact email : pat.chormai@gmail.com

Technische Universität Berlin

Introduction

RNNs have become very popular in various sequence-processing applications. However, it is still unclear how their decisions/outputs are produced, raising concerns among stakeholders.

Because RNNs need to summarize information across timesteps, we hypothesize that well-structured RNNs would have an advantage of being more explainable although they perform equivalently in terms of objective function. In particular, we study and answer the following questions

- How does the structure of RNNs affect their explainability?
- Is LSTM more explainable than standard RNNs?

Explanation Methods

Neural networks or RNNs can be viewed as $\boldsymbol{x} \mapsto f(\boldsymbol{x})$.

Explanation methods aim to find relevance scores $R_i(\boldsymbol{x})$ quantifying the importance of every component in $x_i \in \boldsymbol{x}$ to $f(\boldsymbol{x})$.

- Sensitivity Analysis (SA) [1] : $R_i(\boldsymbol{x}) = \left(\frac{\partial f(\boldsymbol{x})}{\partial x_i} \right)^2$

- Guided Backprop (GB) [2] is proposed for ReLU-type architectures. It is based on computing the derivatives, but local gradients are backpropagated only when incoming activations and the signal are not positive.

- Deep Taylor Decomposition (DTD) [3] is derived specifically for explaining neural networks with ReLU activations. It redistributes $f(\boldsymbol{x})$ to $\boldsymbol{x}$ using certain propagation rules derived from the Taylor expansion.

Given j and k are neurons in two consecutive hidden layers, DTD distributes relevance scores as follows :

$$R_j(\boldsymbol{x}) = \sum_k \frac{a_j w_{jk}^+}{\sum_{j'} a_{j'} w_{j'k}^+} R_k(\boldsymbol{x})$$

where a_j is neuron j's activation and $w_{jk}^+ = \max(0, w_{jk})$.

Experimental Setup

We construct an artificial classification problem using MNIST and FashionMNIST. The classification is to determine the majority sample in the sequence. We name this problem **MNIST-MAJ**.

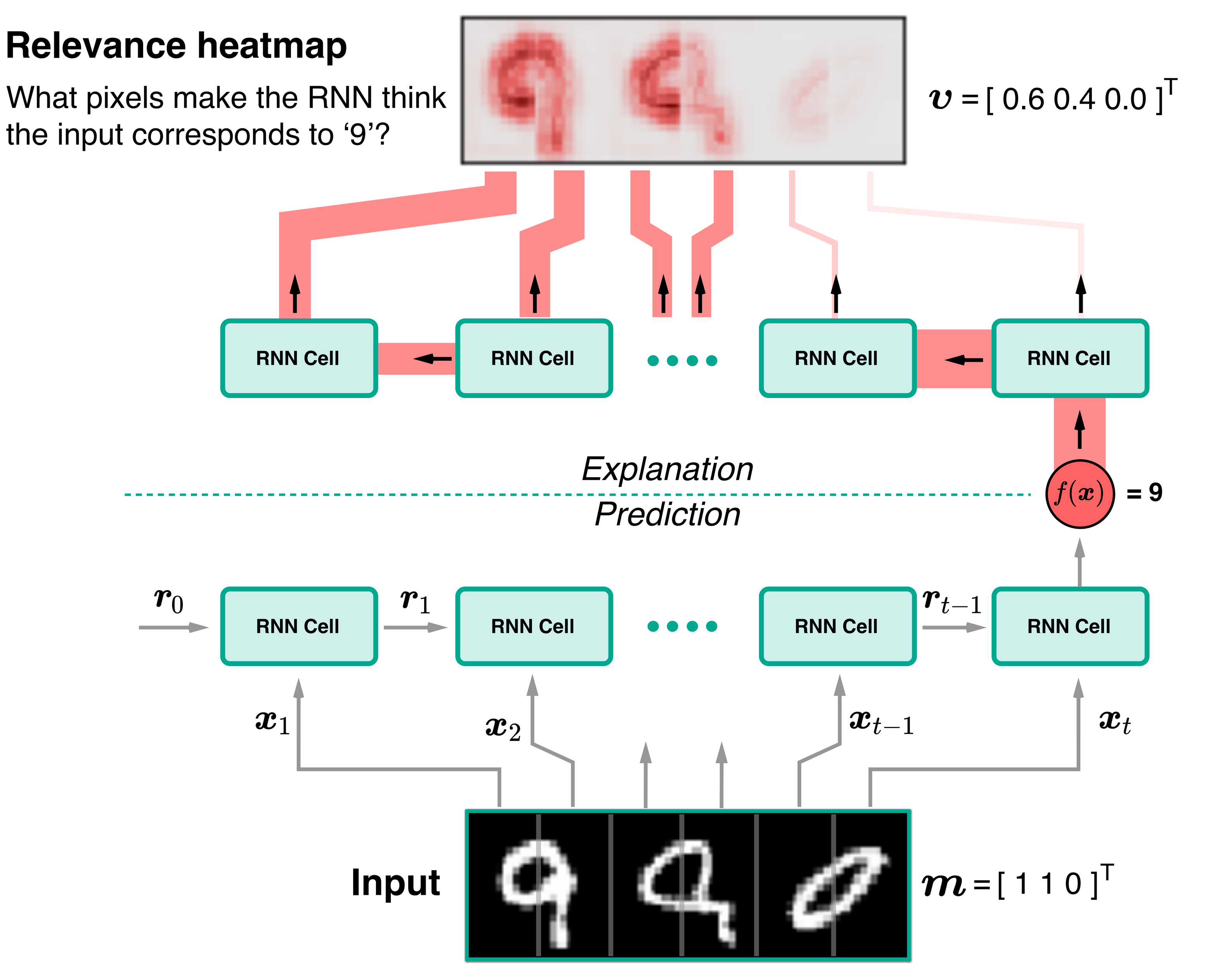

Fig. 1: MNIST-MAJ classification problem and how explanation methods are applied.

References

[1] - Simonyan et al. (2013). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps
[2] - Springenberg et al. (2014). Striving for Simplicity: The All Convolutional Net.
[3] - Montavon et al. (2017). Explaining nonlinear classification decisions with deep Taylor decomposition.
[4] - Pattarawat Chormai. Designing Recurrent Neural Networks for Explainability. http://bit.ly/pat-thesis-repo

Architectures

We consider five RNN architectures including Shallow, Deep, ConvDeep and a modified LSTM (R-LSTM) where tanh activations are replaced by ReLU. We also experiment R-LSTM with convolutional layers (ConvR-LSTM).

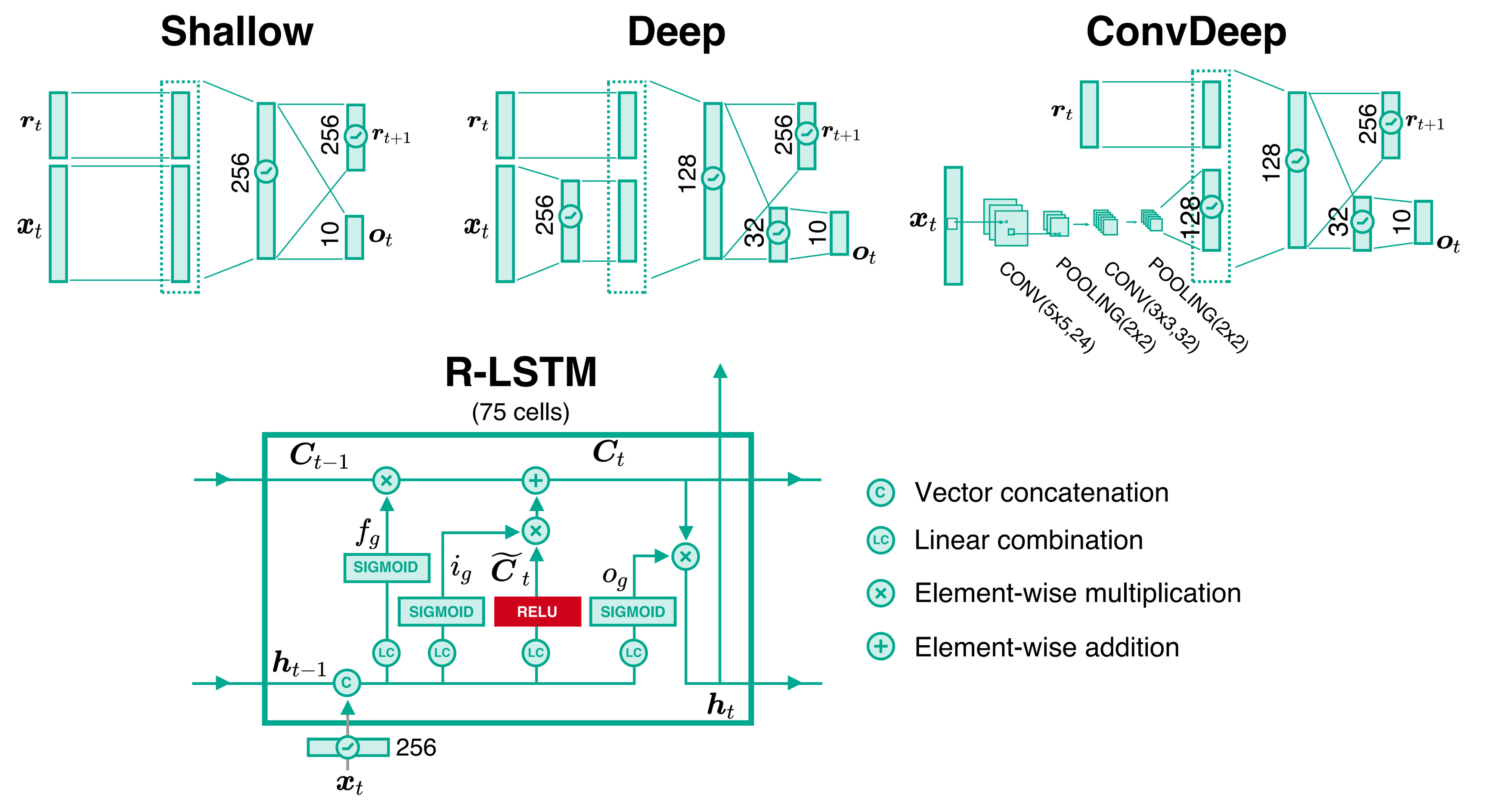

Fig. 2: Shallow, Deep, ConvDeep, R-LSTM architectures with the number of neurons in each layers depiced.

Results

We train models to reach accuracy approximately 98% for MNIST and 85% for FashionMNIST using sequence length 12 ($\boldsymbol{x}_t \in R^{28 \times 7}$).

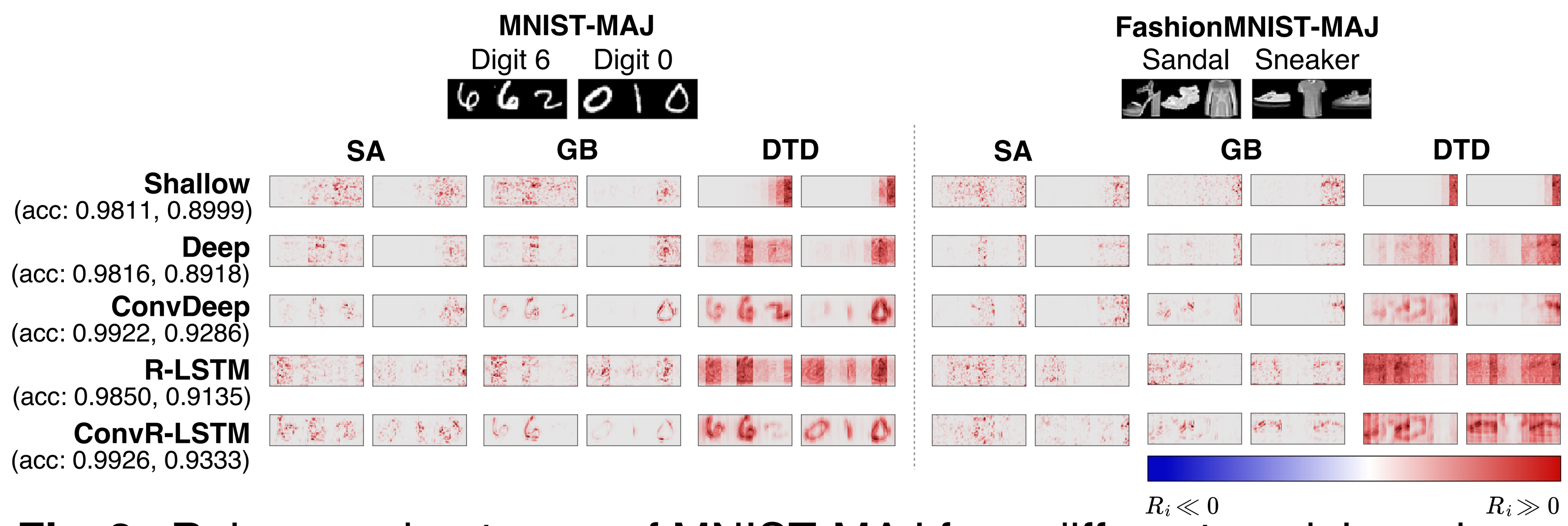

Fig. 3: Relevance heatmaps of MNIST-MAJ from different models and explanation methods.

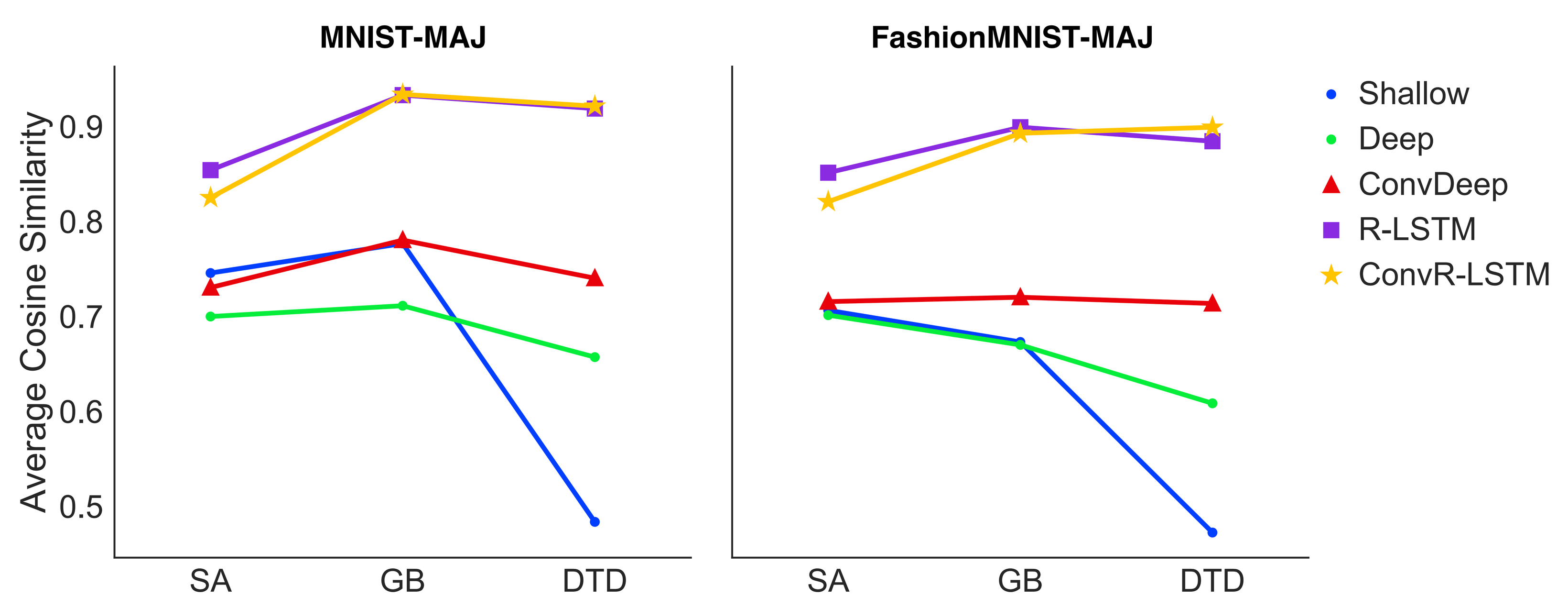

Fig. 4: Quantitative evaluation of explainability of RNNs using the cosine similarity at digit/item level between the binary vector $\boldsymbol{m}$ indicating whether each digit/item is relevant and the vector of relevance scores $\boldsymbol{v}$.

Discussion

- Our results show that deeper RNN and LSTM-type architectures have more explainable predictions even though their accuracy is equivalent.
- The results also suggest that DTD is more sensitive to the architecture of RNNs than SA and GB.

Acknowledgement

I would like to thank Prof. Df. Klaus-Robert Müller and Dr. Grégoire Montavon for guidance and support througout the study.