Approximation Theory of a Class of Tensor-Train Neural Network

Bo Li,[1] *Jianwei Ma*,[1,2] *Eugene Tyrtyshnikov*[3]

1 Harbin Institute of Technology, 2 Peking University, 3 Lomonosov Moscow State University

Abstract

Tensor-Train neural network(TT-NN) is a class of network where inputs, weights and biases are in the form of TT-format. This study employs methods such as functional analysis and constructive techniques to mathematically analyze the approximation properties of one type of TT-NN. Numerical simulations are conducted to verify the equivalence between traditional TT-NN and normal neural network.

Introduction

TT-NN has become a substitute of the traditional neural network (such as RNN) in academic field. As it owns the properties of low-rank representation and approximation to functions, TT-NN can significantly enhance computational efficiency. However, theoretical research on TT-NN remains limited and it still lacks sufficient interpretability.

This poster provides a mathematical analysis for full TT-NN. I have tried to construct a full TT-NN to approximate a function model and give a qualitative proof of the universal approximation theory. This work provides some theoretical foundations for future research on TT-NN.

Mathematical Analysis

TT-NN is a class of network where inputs, weights and biases are in the form of TT-format. I focus on fully connected neural network with Relu activation function as the basic structure of neural network. The poster provides the exploration on 2 types of TT-NN, which I name as traditional TT-NN and full TT-NN.

Mathematical Representation of Traditional TT-NN

$$y = W^l_{Mat_{TT}}\sigma(W^{l-1}_{Mat_{TT}}\sigma(\cdots\sigma(W^1_{Mat_{TT}}x_{vec_{TT}} + b^1_{Mat_{TT}})) + b^{l-1}_{vec_{TT}}) + b^l_{vec_{TT}} \quad (1)$$

Mathematical Representation of Full TT-NN

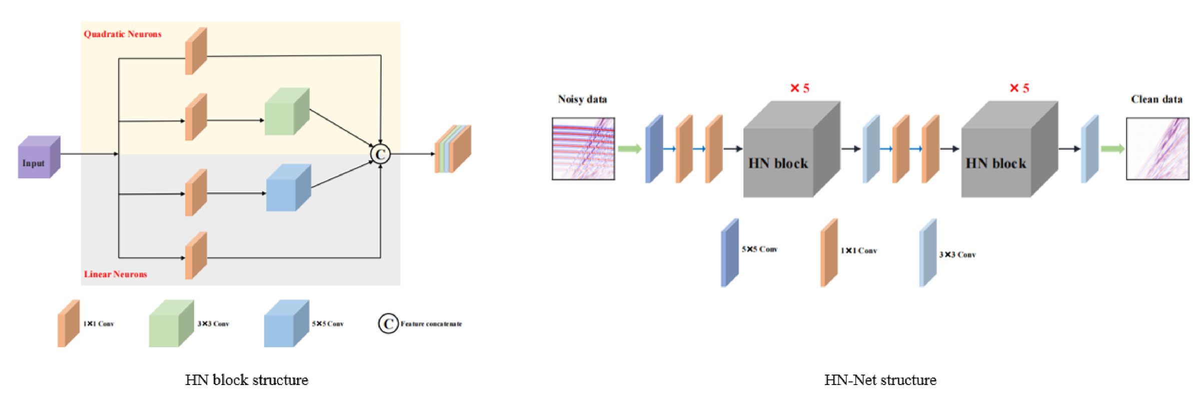
$$y = W^l_{TT}\bullet\sigma_{TT}(W^{l-1}_{TT}\bullet\sigma_{TT}(\cdots\sigma_{TT}(W^1_{TT}\bullet x_{TT} + b^1_{TT})) + b^{l-1}_{TT}) + b^l_{TT} \quad (2)$$

where $x_{vec_{TT}}$, $W^l_{Mat_{TT}}$ and $b^l_{vec_{TT}}$ are the TT-format input, weight of the lth layer and bias of the lth layer, which are expressed as a vector, matrix and matrix. $\sigma(\cdot)$ is the activation function.

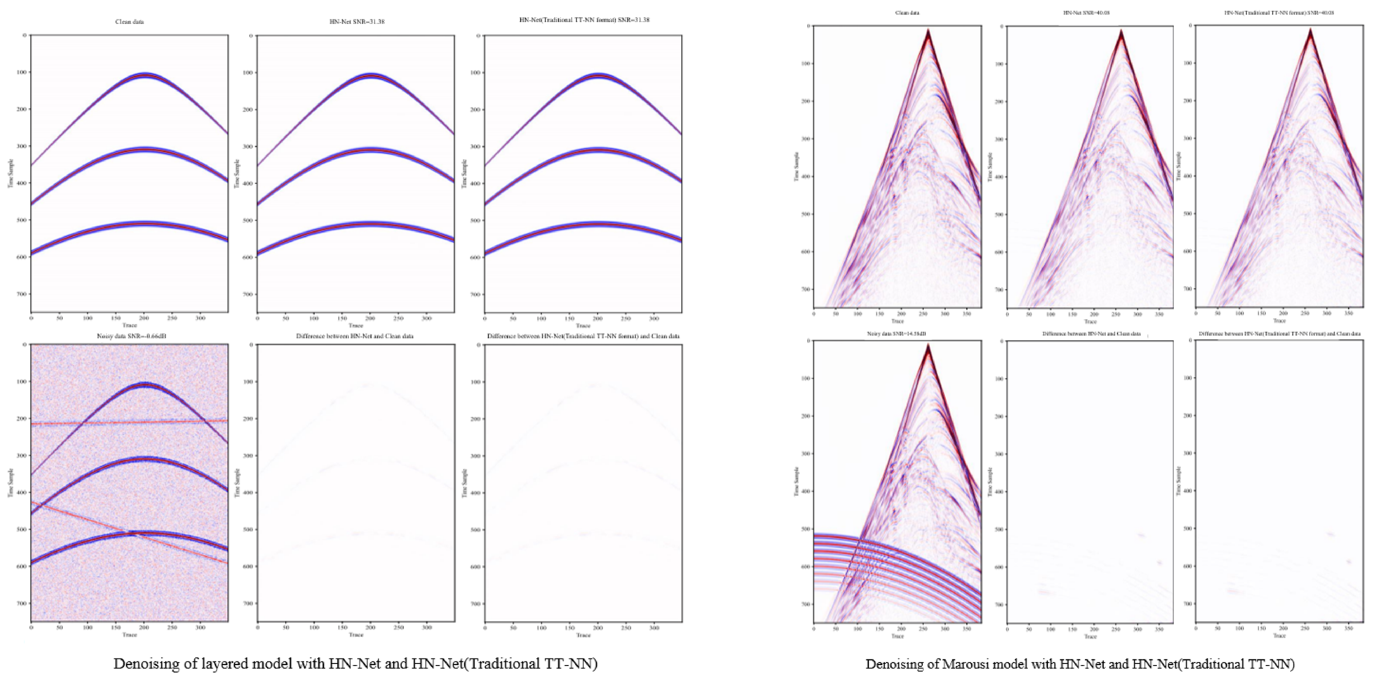
where x_{TT}, W^l_{TT} and b^l_{TT} are the TT-format input, weight of the lth layer and bias of the lth layer, which are expressed as a TT-format tensor, TT-format tensor and TT-format tensor. $\sigma(\cdot)_{TT}$ is the activation function with TT-format connection. $\bullet$ is the product of TT-format operation.

Equivalence between Traditional TT-NN and Neural Network

Traditional TT-NN is equivalent to normal neural network with the same structure. I designed a special network for denoising task to verify the equivalence between the traditional TT-NN and normal neural network.



Denoising Results:



The numerical results show that the error of 2 networks is almost zero, which proves the equivalence between traditional TT-NN and normal NN.

Approximation Theorem of Full TT-NN

For proving the universal approximation of neural networks, one usually carry out two steps: descomposition and construction.

Step 1: The first is to show that function model has a decomposition in terms of fundamental building blocks.

Step 2: The second is to show that each of the building blocks can be approximated by deep networks.

Proof Outline

Theorem: Suppose a function $f(x_1, x_2\cdots, x_{n_1n_2\cdots n_d})$ is defined on a compact set and can be approximated by a fully connected neural network with ReLU activation: $y = W^l\sigma(W^{l-1}\sigma(\cdots\sigma(W^1x+b^1)+b^{l-1}))+b^l$ and the function can also be approximated by a full TT-NN with a single layer.

Proof: With width or depth increasing,

$$\|f(x) - W^l\sigma(W^{l-1}\sigma(\cdots\sigma(W^1x+b^1)+b^{l-1}))-b^l\| \to 0 \quad (3)$$

$\|\cdot\|$ is normal of the function. Multi-layer Fully connected ReLU neural network can be expanded as a single-layer Relu neural network.

$$\sum_i k_i\sigma(a^i_1x_i + \cdots + a^i_{n_1\cdots n_d}x_{n_1\cdots n_d} + b^i) \quad (4)$$

w is width of network. a^i_j and k_i is coefficient of expansion. b^i is bias. $i \le 2^{wl}$ means the number of hyperplanes is at most 2^{wl}. $x_1, \cdots, x_{n_1\cdots n_d}$ and $a^i_1, \cdots, a^i_{n_1\cdots n_d}$ can be seen as d dimension tensors.

Through TT-decomposition, function (4) canbe expressed as:

$$\sum_i k_i\sigma((\sum_{n_1} W_1\otimes X_1, b_1)(\sum_{n_2} W_2\otimes X_2, b_2)\cdots(\sum_{n_d} W_d\otimes X_d, b_d)) \quad (5)$$

$\otimes$ is Kronecker product. Compared with the TT-format function (2):

$$\sigma_{TT}(W_{TT}\bullet x_{TT} + b_{TT}) = \sigma(\sum_{n_1} W_1\otimes X_1, b_1)\cdots\sigma(\sum_{n_d} W_d\otimes X_d, b_d) \quad (6)$$

In fact, $(\sum W\otimes X, b) = \sigma(\sum W\otimes X + M, b) - (M, 0)$, which means one can choose a matrix M of which every entry is bigger than the entry of $-\sum W\otimes X$. Then one construct a single-layered full TT-NN with an outer activation, which proves the theorem.

Research Direction on Deep TT-NN

In the field of NN applications, deep neural networks offer more advantages over shallow ones. Consequently, I have delved into the exploration of deep TT-NN.

1. Direct Construction Direction

From the perspective of module composition

For example, shallow ReLU fully connected neural networks are piecewise linear continuous functions, and deep networks retain this characteristic.

From the perspective of tensor-train operations,

For example, continuous function can be approximated by linear composition of Fourier functions. Is there a TT-operation-composition of specific functions?

2. Spectral Analysis Direction

First, transform the input into TT-format and the function is transformed into the following format:

$$f(x_1, x_2\cdots, x_{n_1n_2\cdots n_d}) \to f(x^1_1, \cdots x^1_{rn_1}; x^2_1, \cdots x^2_{r^2n_2}; \cdots; x^d_1, \cdots x^d_{rn_d}) \quad (7)$$

Second, take $(x^1_1, \cdots x^1_{rn_1})$ as X_1, take $(x^2_1, \cdots x^2_{r^2n_2})$ as X_2, $\cdots$,take $(x^d_1, \cdots x^d_{rn_d})$ as X_d. Then the function is transformed into:

$$f(x_1, x_2\cdots, x_{n_1n_2\cdots n_d}) \to f(X_1, X_2\cdots, X_d) \quad (8)$$

Third, apply spectral decomposition theory and one can transform function $f(X_1, X_2\cdots, X_d)$ into:

$$f(X_1, X_2\cdots, X_d) = \sum_{r_1=1}^{\infty} f_1(X_1, r_1)\sum_{r_2=1}^{\infty} f_2(r_1, X_2, r_2)\cdots\sum_{r_{d-1}=1}^{\infty} f_2(r_{d-1}, X_d) \quad (9)$$

Reference

DeVore, R., Hanin, B., Petrova, G. (2021). Neural network approximation. Acta Numerica, 30, 327–444.

Oseledets, I. V. (2011). Tensor-train decomposition. SIAM Journal on Scientific Computing, 33(5), 2295–2317.

Wang, M., Li, X., Ma, J. (2025). Beyond linear: Hybrid neuron network for coherent noise suppression, Geophysics.