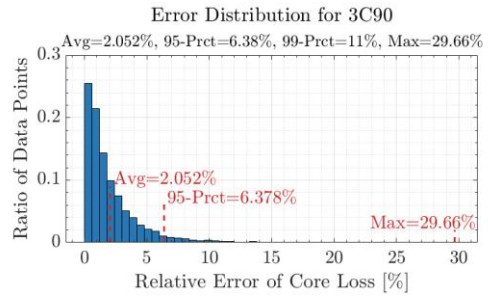
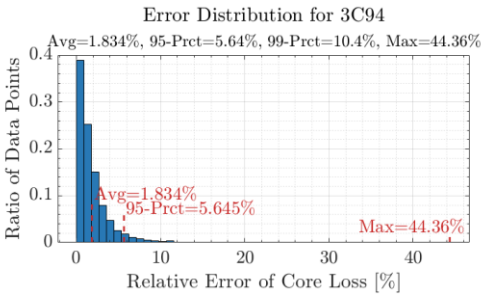
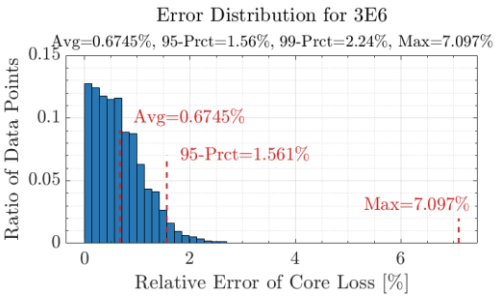
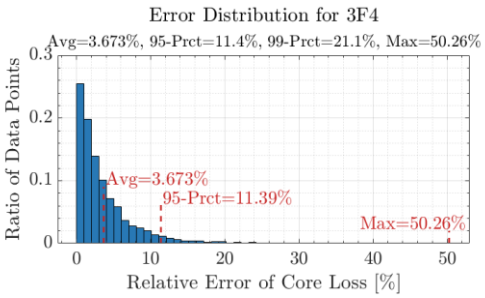
MagNet Challenge Pretest Results for 10 Known Materials – Due 11/10/2023

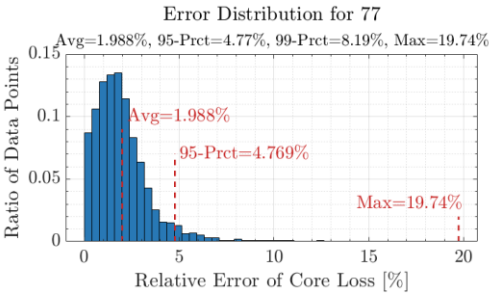
Vision transformer (ViT)-based modeling Method

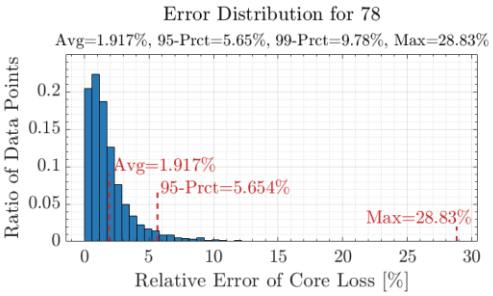


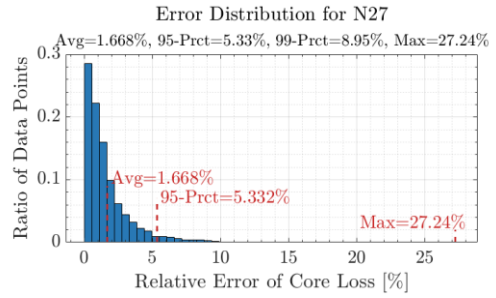


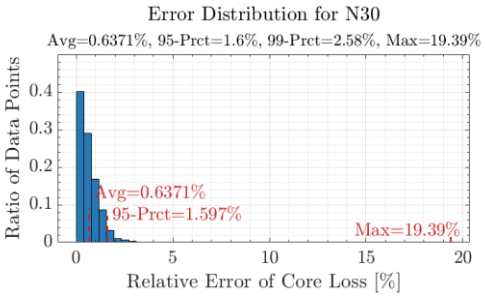


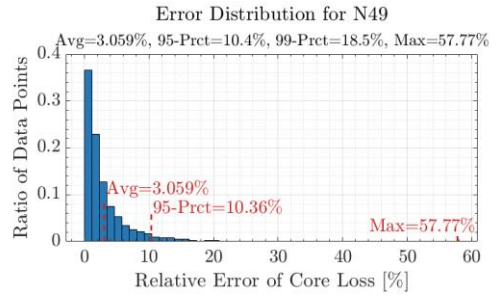












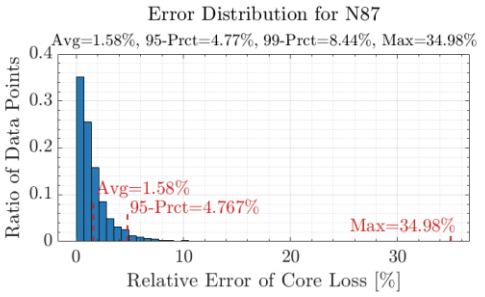


TABLE I
Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolution	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(\log_k n)$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n / r)$
Patched Self-Attention (ViT)	$O(\frac{n^2}{patch\_length} \cdot d)$	$O(1)$	$O(n / patch\_length)$

I. VIT-BASED MODELING METHOD

The Vision Transformer(ViT)-based modeling method provides a further improvement to the Transformer-based magnetic core loss modeling method. The improved method can be used for feature extraction and sequence reconstruction. Comparing with the conventional Transformer-based method, it has advantages such like lower memory consumption, higher modeling accuracy, and higher scalability. Our test scripts and models are available at https://github.com/moeKedama/dg-magnet-test-script.

A. Patch Embeddings Method

The input to the general Transformer is a 1D sequence of token embeddings. The patch embeddings method reshapes the linearly projected input sequence with D-dimensional latent vector size. The input sequence is patched every $patch\_length$ data points, where $patch\_length$ is the number of data points. The final outputs of the method are patch embeddings. Standard learnable 1D positional embeddings are added to replace absolute positional encodings.

In contrast to the down-sampling method in the example, the patch embeddings method preserves as much information as possible at high frequencies while reducing the complexity to an acceptable level. This model exhibits higher performance than the down-sampling method, at the expense of higher training costs.

Table I shows the maximum path lengths, the per-layer complexity and the minimum number of sequential operations for different layer types, where n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in Restricted Self-Attention. Compared to Self-Attention, Patched Self-Attention has a smaller per-layer complexity.

B. Training

The size of patch used by ViT-based modeling method is 8×32. Other detailed parameters can be found in the test script. All models will be trained for 4000 epochs. Learning rate will be manually reduced to 0.1x at 2700 and 3600 epochs. The model is trained at a rate of about 200Epoch/5Min on a PC with a 5.80GHz i9-13900k CPU,128G DDR5 4200MHz RAM and a NVIDIA RTX A6000 GPU (48GB VRAM GeForce RTX 3090). The model needs about 0.6GB of VRAM at 32 batch size.

The ViT-based modeling method uses mean-square error (MSE) as a loss function for training. To achieve the results as shown in the Pretest Results, the convergence should be at least of the order of 10e-5 to 10e-6 on the training set, and at least of the order of 10e-4 to 10e-5 on the validation set.

II. DANN-BASED DOMAIN GENERALIZATION METHOD

The Pretest Results of the domain generalization method are in the 10domain and subdomain directory.

Domain-Adversarial Neural Networks (DANN)-based Domain Generalization (DG) method pits feature extractors and domain discriminators against each other during training to learn domain-invariant features. However, it has poor performance. Here are a few possible reasons for this result.

1) The model size is limited in 10MB model size which reaches a bottleneck.

2) A simple exploratory data analysis of the material data reveals that the dispersion of the data in terms of frequency and temperature is large, leading to a certain degree of flaws in the dataset.

3) Data normalization method need to be improved. Currently used normalization method is performed by counting the means and standard deviations of all domains or sub-domains, which has a strong priori knowledge. The model becomes difficult to converge in cases where the prior knowledge is not fully applicable.

4) ViTs lack inductive biases, which can make it difficult to train them with limited data.