MagNet-AI: Neural Network as Datasheet for Magnetics Modeling and Material Recommendation

Haoran Li, *Student Member, IEEE*, Diego Serrano, *Member, IEEE*, Shukai Wang, *Student Member, IEEE*, and Minjie Chen, *Senior Member, IEEE*

***Abstract*—This article presents the MagNet-AI platform as an online platform to demonstrate the "neural network as datasheet" concept for B–H loop modeling and material recommendation of power magnetics across wide operation range. Instead of directly presenting the measured characteristics of magnetic core materials as time sequences, we employ a neural network to capture the B–H loop mapping relationships of magnetic materials under different excitation waveforms at different temperatures and dc bias. Long short-term memory and transformer-based neural network models are developed, verified, and compared. The neural network can be used to rapidly predict hysteresis loops and core losses under different operating conditions, compare materials, and recommend materials for design. The neural network model is also proved effective in reconstructing the raw measurement while accurately maintaining the magnetic characteristics, enabling rapid material evaluation and comparison.**

***Index Terms*—Core loss, hysteresis loop, machine learning, neural network, power magnetics.**

I. Introduction

POWER electronics systems depend heavily on magnetic components. Due to the large component volume and the significant power loss, power magnetics are typically the bottleneck of the system optimization in terms of the power density and power conversion efficiency. Despite significant advances in power semiconductor devices and circuit topologies, the development of corresponding approaches for designing and modeling power magnetic components and materials is still lacking [3], [4], [5], [6], [7]. Modeling magnetic materials, especially their hysteresis loops, poses a significant challenge due to the complicated excitation–response mechanisms inherent in these materials, the numerous factors involved (such as frequency, temperature, dc bias, and memory effects), and the lack of a fully satisfactory first-principle model. The ability to model and predict the hysteresis loop of power magnetics across a wide operation range has a profound impact on improving the power density and efficiency of power electronics systems.

Currently, the design of power magnetic components relies extensively on the classical datasheets provided by manufacturers, together with conventional modeling tools, such as analytical models or interpolated loss maps. Classical datasheets for magnetic materials are typically only valid for sinusoidal waveforms, and usually cannot provide sufficient information for all potential scenarios that designers may encounter with. The limited information of hysteresis loop and sparse core loss curves impede designers from making accurate predictions on the variations of magnetic permeability and core loss under different operating conditions, such as different amplitudes, frequencies, temperatures, and levels of dc bias. Conventional models for power magnetics, e.g., the Steinmetz equation, the iGSE [8], and the Jiles–Atherton model [9], are established based on empirical simplifications or physical approximations, which limit their modeling accuracy. The magnetic core material behavior is highly complex [10]. The limited complexity of these models limits their capability of capturing sophisticated waveform, temperature, and dc-bias determined impact.

Recent advancements in data-driven methods, specifically machine learning techniques, such as neural networks, have proven to be highly effective in resolving complex nonlinear multivariable regression problems [11], [12], [13], [14]. The primary advantage of utilizing neural networks is the ability to unify many intertwined influencing factors, such as temperature and dc bias, into a cohesive framework, which makes the neural network a good candidate for the data-driven modeling of power magnetics.

This article proposes the concept of "neural network as datasheet" for magnetic materials modeling, and demonstrate neural-network-aided material recommendation for rapid design. The effectiveness of this approach was validated by an open-source online research platform—MagNet-AI. The contributions of this article are as follows.

1) Provided a more comprehensive explanation on the design considerations of the neural network architecture.
2) Introduced a systematic data-processing and data-augmentation technique for neural network training.
3) Compared different neural network architectures for model accuracy and the training cost.

Manuscript received 11 April 2023; revised 20 July 2023; accepted 19 August 2023. Date of publication 28 August 2023; date of current version 23 October 2023. This work was supported in part by ARPA-E Differentiate Project and in part by the Schmidt DataX Fund at Princeton University through the Schmidt Futures Foundation. An earlier version of this paper was presented in part at the 2022 IEEE 23rd Workshop on Control and Modeling for Power Electronics [DOI: 10.1109/COMPEL53829.2022.9829998] and in part at the 2023 IEEE Applied Power Electronics Conference [DOI: 10.1109/APEC43580.2023.10131497]. Recommended for publication by Associate Editor H. S.-H. Chung. *(Corresponding author: Minjie Chen.)*

The authors are with the Department of Electrical and Computer Engineering and the Andlinger Center for Energy and the Environment, Princeton University, Princeton, NJ 08540 USA (e-mail: haoranli@princeton.edu; ds9056@princeton.edu; sw0123@princeton.edu; minjie@princeton.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2023.3309233.

Digital Object Identifier 10.1109/TPEL.2023.3309233

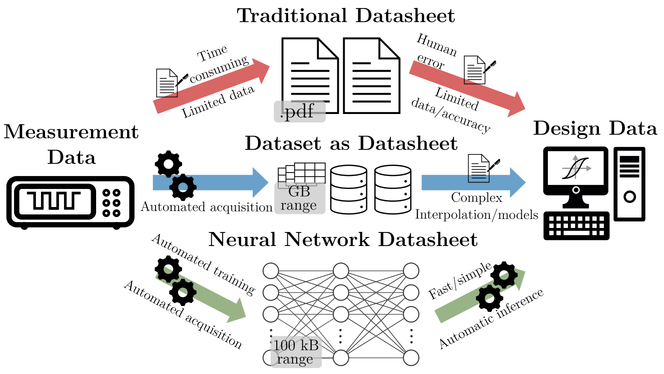


Fig. 1. Concept of neural network as datasheet. A neural network (NN) is better equipped to store the shape of B–H loops for different operating conditions as compared to traditional datasheets or datasets of B–H loops that can only contain limited information, providing an effective and efficient guidance for the design of power magnetic components.

4) Demonstrated the effectiveness of the "neural network as datasheet" concept, including the hysteresis loop and the core loss prediction, the neural network-aided material recommendation, and the online MagNet-AI platform.

Fig. 1 shows the concept of "neural network as datasheet." Traditional datasheets, as stated above, are typically deficient in terms of the information availability, which makes it difficult for designers to achieve an optimal design solely based on limited information. Some manufacturers also offer online dataset as a supplement of the datasheet, where massive measured data points are provided to users to search. However, due to the various influencing factors, the data size increases rapidly as the number of variables increases, typically in the range of tens or hundreds of GB. Meanwhile, it is still inconvenient to utilize these datasets for parameterization of the design models, which usually requires extensive or complicated data extraction and interpolation. The behavioral modeling methods proposed in [15] and [16] provide a feasible approach to overcome this challenge, in which analytical models are developed to map the excitation and operating conditions directly to the component characteristics.

Neural network-aided datasheet, on the other hand, exhibits advantages over the traditional datasheet or the dataset as datasheet. Enabled by the fully automated data acquisition system developed in [17] and [18], the targeted magnetic materials can be automatically characterized and measured, where human errors are minimized. Based on the measurement data, a neural network model will be trained, which encapsulates and compresses all the information of the time domain dataset in GB and results in a small model file with a size in kB or MB ranges, while maintaining a high accuracy. With the well-trained neural network, users only need to perform a quick inference of the model to predict the magnetics performance under specific operation conditions, or a series of inference to track the variations of hysteresis loop and core loss, providing an effective, efficient, and convenient reference for the magnetic components design.

Neural networks have been applied to modeling the core loss or the hysteresis loop of power magnetics [19], [20], [21], [22], [23], [24], [25], [26], [27]. However, existing neural network models have mostly been constructed using outdated or simplistic network structures, such as feedforward neural networks (FNNs), which limits the accuracy of hysteresis loop or core loss prediction. There was no large-scale open-source database available before MagNet [17], limiting the size of the neural network that can be trained effectively. In this work, an encoder–projector–decoder architecture is proposed to develop a sequence-to-sequence model for the hysteresis loop prediction, which takes the flux density sequence $B(t)$ as an input and output the field strength sequence $H(t)$ (or vice versa), incorporating other inputs variables, such as the frequency f, the temperature T, and the dc bias H_{dc}. More specifically, we implement the proposed architecture using two commonly used sequence-to-sequence neural network structures–the long short-term memory (LSTM) [28] and the transformer [29], two of the most successful sequence-to-sequence neural network architectures. After proper training, they can accurately and rapidly predict the B–H loop of power magnetics under a wide range of operating conditions while significantly reducing data size by storing trained parameters rather than raw time domain data.

The rest of this article is organized as follows. Section II provides an overview of the proposed encoder–projector–decoder architecture and its data flow. Sections III and IV describe the details of the LSTM-based and the transformer-based network implementations, respectively. Section V presents the data processing and augmentation techniques used during network training. Section VI evaluates and compares the testing results of each implementation. Sections VII, VIII, and IX demonstrate the applications of neural network-aided smart datasheet, including the hysteresis loop and core loss prediction, magnetic materials comparison, and the online platform. Finally, Section X concludes this article.

II. Encoder–Projector–Decoder Architecture

Hysteresis loops of power magnetics are determined by various influencing factors. Besides the different shapes of excitation waveform, other operating conditions, such as different frequencies, temperatures, and levels of dc bias will all result in different shapes of B–H loops. Fig. 2 compares multiple measured B–H loops for N87 ferrite material as an example, where the material characteristics differ significantly under different conditions. These various factors are quantified in [10], and in real-world applications, they often coexist and change concurrently, which renders the modeling of magnetic materials extremely difficult. To develop a neural network model that is capable of predicting the hysteresis loop under different operating conditions, an encoder–projector–decoder architecture is proposed in this article.

The structure and data flow of the proposed encoder–projector–decoder neural network architecture is shown in Fig. 3. The general concept of this architecture is to map a time series into another time series while incorporating other information about the operating conditions. In this work, the input sequence is $B(t)$, and the output sequence is $H(t)$, which define the basic shape of hysteresis loops. Scalar inputs, such as

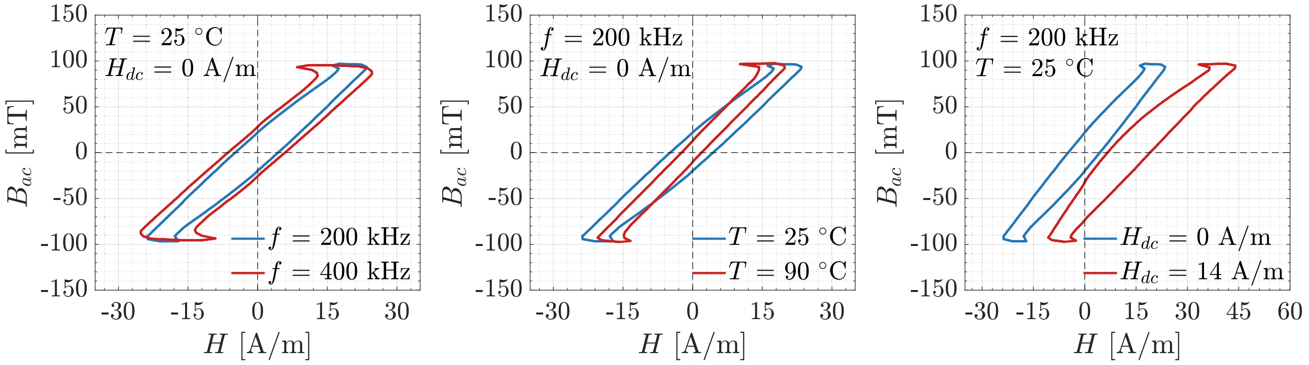


Fig. 2. Examples of B–H loops measured with N87 ferrite material under 50% duty ratio triangular excitations. The reference loop (blue) is measured at 200 kHz, 25 °C, and 0 A/m DC bias. Each of the three figures shows the variation of B–H loop at different frequencies, temperatures, and DC biases, respectively. The B (only AC) waveform is extracted from voltage measurement, and the H (both AC and DC) waveform is extracted from current measurement.

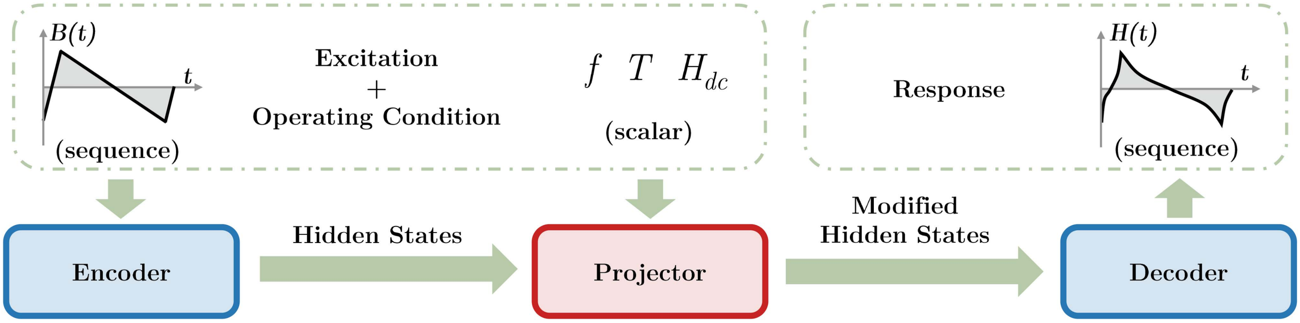


Fig. 3. Architecture and data flow of the encoder–projector–decoder neural network architecture.

frequency f, temperature T, and dc bias H_{dc}, also significantly affect the B–H loops. Therefore, an additional projector is implemented between the encoder and the decoder to take these scalar inputs into consideration and accurately predict the B–H loop under different operating conditions.

The encoder receives the $B(t)$ sequence as input and transforms it into a fixed-dimensional vector by capturing the sequential information and temporal correlations, including shape, pattern sequence, amplitude, and the relative change rate of the excitation waveform. The encoder outputs the hidden state vectors containing all the relevant information extracted from the input sequence and mapped into a hidden state domain. The hidden state vectors are then passed through the projector and adjusted according to the scalar inputs (frequency f, temperature T, and dc bias H_{dc}). The projector is necessary because the shape of the B–H loop is determined not only by the $B(t)$ sequence, but also by many other factors. Finally, the modified hidden state vectors are processed by the decoder to predict the output sequence $H(t)$. During model inference, the expected response sequence is produced in an autoregressive manner. At each time step, the prediction is generated not only based on both the current hidden state vectors, but also all the previously generated predictions. With the autoregression, the temporal information of the sequence is retained and reconstructed sequentially with hidden time causality.

The geometry of a specific magnetic component can significantly affect the component-level behaviors [30], [31], [32]. The model proposed in this work is limited to material-level modeling. The architecture introduced in this article can be further extended to cover component-level geometry impact, which is beyond the scope of this article.

The encoder and decoder modules can be implemented using different neural network architectures, recurrent neural networks (RNNs) [33], attention-based networks (transformer), or convolutional neural networks (CNNs) [34], [35], [36], all of which have shown success in modeling sequences with complex temporal dependencies. Wavelet-CNN-based neural network has been applied to model core loss in [37]. In this work, we specifically investigate and provide guidance on the use of both LSTM-based and transformer-based implementations for the encoder–projector–decoder neural network, which is designed to map a time series input to another time series output while incorporating information about external factors.

III. LSTM Neural Network Models

LSTM is a specialized type of RNN that is wellsuited for capturing the temporal relationships within time series data [38]. The effectiveness of LSTM networks in solving sequence-to-sequence tasks has been demonstrated, with the LSTM encoder–decoder architecture being one of the most widely adopted

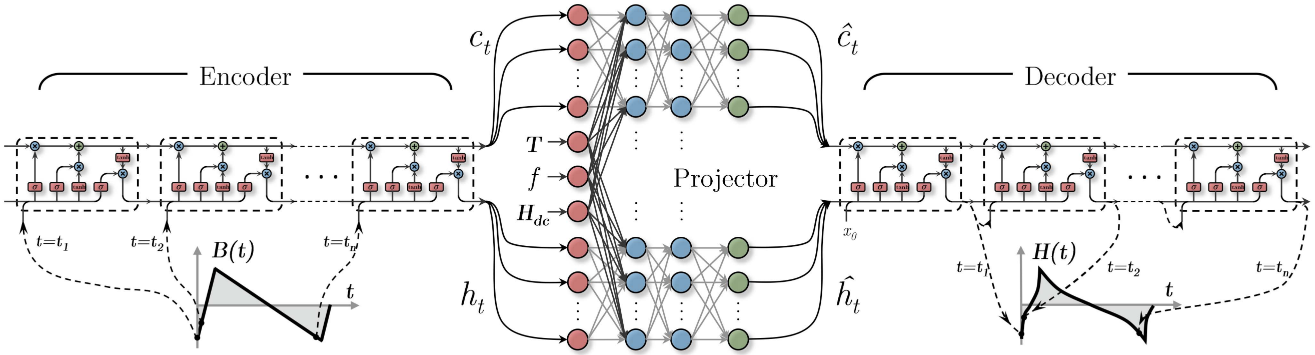


Fig. 4. Neural network structure of the LSTM-based encoder–projector–decoder architecture. Temperature (T), frequency (f), and DC bias (H_{dc}) information are mixed with the waveform information in the FNN projector after the encoder and before the decoder.

implementations [39]. The LSTM-based encoder–decoder architecture has a well-established ecosystem in popular deep learning frameworks, such as PyTorch and TensorFlow.

In the LSTM-based encoder–projector–decoder architecture shown in Fig. 4, both the encoder and the decoder are implemented as LSTM neural networks. At a given time step $t = t_i$, the input sequence of $B(t)$ is inputted to the LSTM network and processed through the input gate, forget gate, and output gate on the encoder side. The temporal information is stored in the cell states and the hidden states, which are fed back through the recurrent connections for processing the next input at $t = t_{i+1}$. By unwrapping the recurrent connections across the timeline, it is equivalent to passing the entire input sequence through a series of LSTM networks. Mathematically, the operation of the LSTM cell at time t can be described as

$$\begin{aligned}
f_t &= \sigma\left(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}\right)\\
i_t &= \sigma\left(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}\right)\\
g_t &= \tanh\left(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}\right)\\
o_t &= \sigma\left(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}\right)\\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t\\
h_t &= o_t \odot \tanh(c_t)
\end{aligned} \tag{1}$$

where x_t is the sequential input at time t. Intermediate variables f_t, i_t, and o_t represent the value of the input gate, forget gate, and output gate, respectively, and c_t and h_t refer to the cell states and the hidden states, which are the recurrent variables being fed back to the LSTM cell and thus providing the memorizing capability. The function $\sigma(x)$ is the Sigmoid function that operates as the activation function to provide the nonlinear learning capability. As in an FNN, W and b are the weights and biases, respectively, and the subscript refers to the source and target variables that they are applied to. The operator $\odot$ stands for the Hadamard product, which performs an element-wise product for all the elements of two matrices.

Then, the cell states and the hidden states at the last time step are concatenated with the additional inputs and fed into the projector, where these state vectors are modified using a FNN. The LSTM network in the decoder uses the modified states to predict the output $H(t)$ at $t = t_0$, which is then fed back as the input for the next prediction at $t = t_1$. The prediction continues until the entire output sequence is generated.

Algorithm 1: LSTM-Based Model.

```
Input:
    Flux Density $B(t)$, Frequency $f$, Temperature $T$,
    DC bias $H_{dc}$;
Output:
    Magnetic Field Strength $H(t)$;
Initialize hidden states $h_0$ and cell states $c_0$;
$x_1 \leftarrow B(t_1)$;
for $i = 1$ to L do
    $h_i,\ c_i \leftarrow \mathrm{LSTM}_1\big(h_{i-1},\ c_{i-1},\ B(t_i)\big)$;   [Encoder]
$h'_0 \leftarrow \mathrm{FNN}_1(h_L,\ f,\ T,\ H_{dc})$;
$c'_0 \leftarrow \mathrm{FNN}_2(h_L,\ f,\ T,\ H_{dc})$;   [Projector]
Initialize $y_0$;
for $i = 1$ to L do
    $h'_i,\ c'_i,\ y_i \leftarrow \mathrm{LSTM}_2\big(h'_{i-1},\ c'_{i-1},\ y(t_{i-1})\big)$; [Decoder]
return $H(t) \leftarrow \{y_1,\ y_2,\ \ldots,\ y_L\}$;
```

More details of the data flow in the LSTM model are described by the pseudocodes in Algorithm 1. Models and example codes are available on MagNet GitHub repository.[1]

IV. Transformer Neural Network Models

Transformer with the attention mechanism is another very successful network architecture that excels at modeling sequence-to-sequence problems, such as large language models represented by ChatGPT. Unlike RNNs, the transformer eschews recurrent connections, but instead relies entirely on attention mechanisms to capture temporal dependencies between the input and output sequences. Modified from the original structure in [29], we implement an encoder–projector–decoder architecture, as shown in Fig. 5.

The data point at each time step in the input sequence $B(t)$ is first passed through a shallow FNN and transformed to a d-dimension vector, which sets the representation dimension of the model. Given that the attention mechanism used in the transformer model is essentially the dot product of matrices, the

[1] MagNet repository: https://github.com/PrincetonUniversity/magnet/

Algorithm 2: Transformer-Based Model.

```
Input:
    Flux Density $B(t)$, Frequency $f$, Temperature $T$, DC bias $H_{dc}$, Field Strength $H(t)$ (only available in training);
Output:
    Magnetic Field Strength $H(t)$;
    $X \leftarrow \mathrm{FNN}_1\big(B(t)\big)$;   [Mapping]
    $X \leftarrow X + $ Positional Encoding;
    $X \leftarrow \mathrm{Norm}\big(X + \text{Self-Attention}_1(X)\big)$;
    $X \leftarrow \mathrm{Norm}\big(X + \mathrm{FNN}_2(X)\big)$;   [Encoder]
    $X' \leftarrow \mathrm{FNN}_3(X,\ f,\ T,\ H_{dc})$;   [Projector]
    if training then
        $Y \leftarrow \mathrm{FNN}_4\big(H(t)\big)$;   [Mapping]
        $Y \leftarrow Y + $ Positional Encoding;
        $Y \leftarrow \mathrm{Norm}\big(Y + \text{Self-Attention}_2(Y)\big)$;
        $Y' \leftarrow \mathrm{Norm}\big(Y + \text{Input-Output-Attention}(X', Y)\big)$;
        $Y' \leftarrow \mathrm{Norm}\big(Y' + \mathrm{FNN}_5(Y')\big)$;   [Decoder]
        $H(t) \leftarrow \mathrm{FNN}_6(Y')$   [Mapping]
    else if testing then
        Initialize $H_0(t) \leftarrow 0$;
        for $i = 1$ to L do
            $Y \leftarrow \mathrm{FNN}_4\big(H_{i-1}(t)\big)$;   [Mapping]
            $Y \leftarrow Y + $ Positional Encoding;
            $Y \leftarrow \mathrm{Norm}\big(Y + \text{Self-Attention}_2(Y)\big)$;
            $Y' \leftarrow \mathrm{Norm}\big(Y + \text{Input-Output-Attention}(X', Y)\big)$;
            $Y' \leftarrow \mathrm{Norm}\big(Y' + \mathrm{FNN}_5(Y')\big)$;   [Decoder]
            $H_i \leftarrow \mathrm{FNN}_6(Y')$   [Mapping]
        $H(t) \leftarrow H_L$;
    return $H(t)$;
```

time steps in the sequence are permutable. To ensure the model effectively captures temporal dependency, the input vector is combined with a positional encoding vector, providing information about the position of each time step in the sequence. The resulting vector is then fed into the self-attention module, which analyzes and captures the temporal dependency within the input sequence itself. Further processed by a FNN, a set of hidden vectors encapsulating the information of the input sequence is generated and passed to the projector.

Next, the hidden vectors obtained from the encoder are similarly concatenated with the additional inputs, such as frequency f, temperature T, and dc bias H_{dc}, and the resulting vectors are passed through a FNN-based projector. The projector modifies the hidden vectors by considering the influence of these additional inputs. The modified hidden vectors are then passed to the decoder for reconstructing the output sequence.

Besides the hidden vectors, the input of the decoder consists of a reference sequence. During the network training, it is the target output sequence; during the network testing, it is the sequence predicted by the model itself (initialized with zero), shown as the dashed line in Fig. 5. The reference sequence is similarly mapped to a d-dimension vector with a shallow FNN, summed with a positional encoding vector, and fed into the self-attention module to generate another set of hidden vectors. Both sets of hidden vectors from the projector and the self-attention module are further processed with the input–output attention module. Finally, the resulting output vectors are processed by a FNN to generate the desired output sequence $H(t)$.

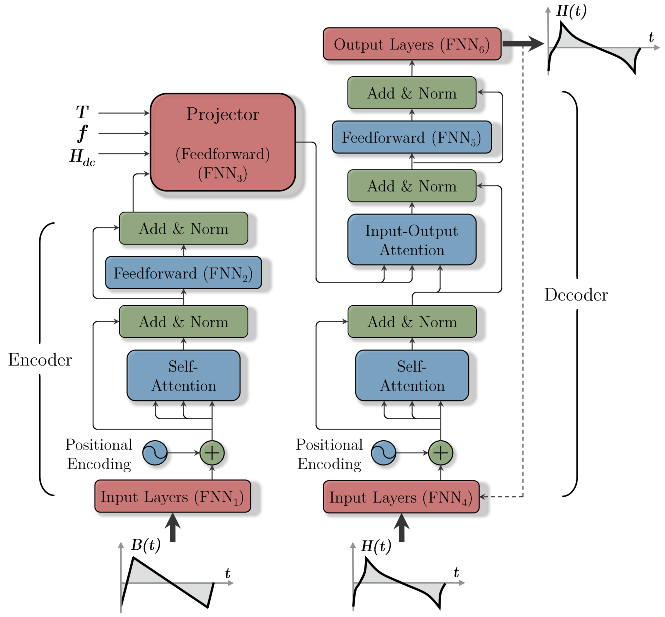


Fig. 5. Network structure of the transformer-based encoder–projector–decoder architecture. $B(t)$ waveform is the sequence input of the encoder. T, f and H_{dc} are the scalar inputs of the projector. During the model training, the targeting $H(t)$ is directly fed to the decoder as a reference input. During the model inference, the predicted sequence is fed back to the decoder, generating the entire output sequence in an autoregressive manner.

More details of the data flow in the transformer model are described by the pseudocodes in Algorithm 2. Models and codes are available on MagNet GitHub repository as well.

V. Data Processing and Augmentation

The prediction accuracy of a neural network model is fundamentally determined by the quality of the training data. In this work, the training data is constructed based on the massive measured dataset in the MagNet database [17], [18]. The database currently includes B–H loop measurements for ten different ferrite materials across a wide range of excitation and operation conditions, collected by an automated data acquisition system. All the measurements are captured in the periodic steady state operation.

The MagNet dataset comprises five data fields: the flux density waveform $B(t)$, the field strength waveform $H(t)$, the fundamental frequency f, the temperature T, and the dc bias H_{dc}. The fundamental frequency is determined based on the measured voltage waveform using Welch's frequency domain method [40], while the remaining four data fields are obtained directly from the measurements. In the following sections of this article, we present the example results based on the dataset of N87 ferrite material, which contains 142 871 measured data points (B–H loops) covering the range of flux density amplitude in

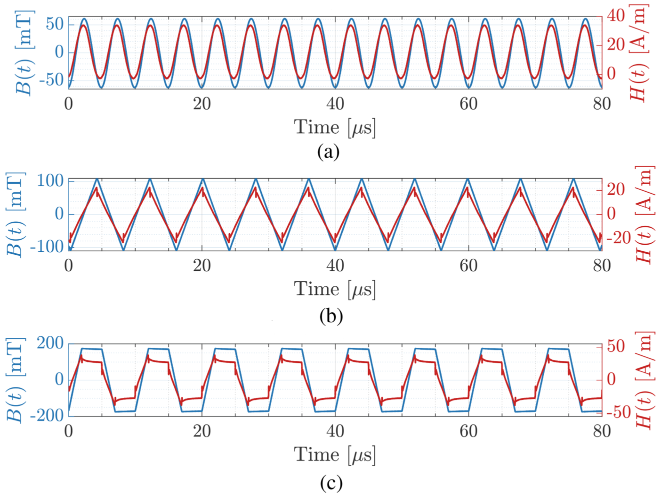


Fig. 6. Examples of the full-length waveforms measured with N87 ferrite material under different frequency, temperature, and DC bias conditions. (a) Sinusoidal wave: $f = 200$ kHz, $T = 25$ °C, $H_{\mathrm{dc}} = 15$ A/m. (b) Triangular wave: $f = 125$ kHz, $T = 70$ °C, $H_{\mathrm{dc}} = 0$ A/m. (c) Trapezoidal wave: $f = 100$ kHz, $T = 50$ °C, $H_{\mathrm{dc}} = 0$ A/m.

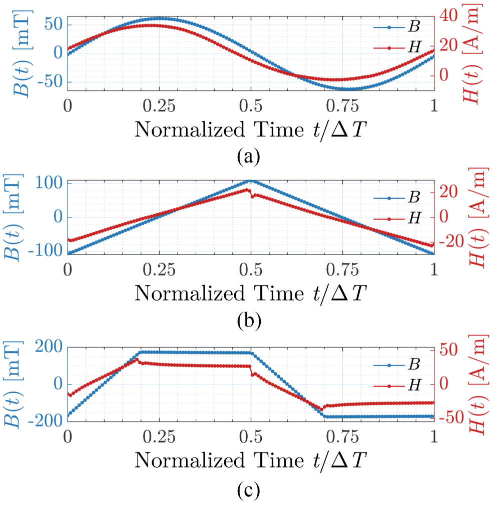


Fig. 7. Examples of the single-cycle waveforms with the sequence length of 128, corresponding to the full-length waveforms shown in Fig. 6. (a) Sinusoidal wave: $f = 200$ kHz, $T = 25$ °C, $H_{\mathrm{dc}} = 15$ A/m. (b) Triangular wave: $f = 125$ kHz, $T = 70$ °C, $H_{\mathrm{dc}} = 0$ A/m. (c) Trapezoidal wave: $f = 100$ kHz, $T = 50$ °C, $H_{\mathrm{dc}} = 0$ A/m.

10–300 mT, fundamental frequency in 50–500 kHz, temperature in 25 °C–90 °C, and dc bias field strength in 0–60 A/m, with various waveform shapes of various duty ratios. The measured waveform shapes can be categorized into three types: sinusoidal, triangular, and trapezoidal, according to the shape of the flux density waveform. Besides the amplitude, the duty ratio D of each section in the triangular and trapezoidal waves can be tuned independently. This training dataset serves as the basis for a neural network model capable of predicting the hysteresis loop under various operating conditions with different excitation waveform shapes. Fig. 6 shows several waveform examples contained in the dataset. Due to the unknown permeability and the relaxation effect, we cannot precisely estimate B_{dc} from H_{dc}. The flux density $B(t)$ here only contains the ac part, which is extracted from the voltage measurement, while the $H(t)$ contains both the ac and dc parts that are directly extracted from the current measurement. All the flux densities in the hysteresis loops shown below also only consider the ac part B_{ac}.

To ensure a better convergence of the model, all five data fields $\{B(t), H(t), f, T,$ and $H_{\mathrm{dc}}\}$ are normalized before being fed into the neural network. This is accomplished by subtracting the average and rescaling with the standard deviation for each data field. Parameters for standardization are saved and reused during model testing and inference.

The two sequence inputs $\{B(t)$ and $H(t)\}$ require more complex processing in order to ensure that the network achieves good prediction accuracy and sufficient generalization capability. Adding reasonable noise to the data to avoid overfitting is the most commonly used preprocessing technique. In this study, both the $B(t)$ and the $H(t)$ waveforms are superposed with white noise of uniform distribution within the range of ±0.1 mT and ±0.05 A/m, respectively. Moreover, the sequence inputs are further processed and augmented in three ways to improve the network performance.

A. Single-Cycle Interpolation

In the aforementioned original MagNet database, the $B(t)$ and $H(t)$ waveforms are directly calculated from the raw measurements of voltage and current signals. These waveforms consist of multiple cycles that are captured in the periodic steady state. Each waveform is a $1\times10\,000$ time sequence, with a sampling rate of $f_s = 125$ MHz. Each of the waveform comprises multiple waveform cycles that are captured in the steady state. Training the network with these multicycle waveforms; however, can be problematic. On the one hand, the large number of data points significantly increases the computational cost of network training and inference. Considering the fact that the waveform is captured in periodic steady state, the repeating cycles do not provide much valuable information to the network. On the other hand, waveforms with different frequencies have different numbers of samples in each cycle, leading to different numbers of points on the B–H plane. Networks trained with these waveforms are prone to magnify noises in the low-frequency waveforms while overlooking sharp transitions in the high-frequency waveforms.

To address these issues, a single-cycle interpolation algorithm is applied to all the $B(t)$ and $H(t)$ waveforms. Given the sampling rate f_s and the fundamental frequency f, the total number of cycles contained in each waveform can be calculated as $N = 10\,000 \times (f/f_s)$. The 10 000-sample waveform is first interpolated into $N \times 128$ samples using the spline algorithm. Then, the interpolated waveform can be evenly sliced into multiple sections, where each section contains a full cycle of the waveform with 128 sample points in total. Finally, all the sections are averaged into a single-cycle waveform. Fig. 7 shows the corresponding single-cycle waveform for each of the original waveforms shown in Fig. 6. By applying the single-cycle algorithm, the time stamp of the waveform is normalized by the period $\Delta T = 1/f$ into [0,1]. The single-cycle waveforms of $B(t)$ and $H(t)$ well describe the shape of the hysteresis loop in

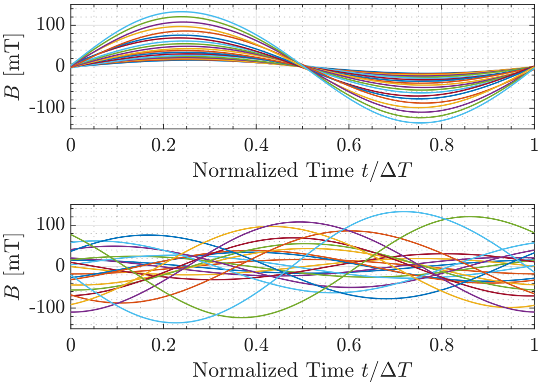


Fig. 8. Set of example $B(t)$ waveforms of N87 ferrite before and after the phase shifting augmentation. Waveforms are measured under sinusoidal excitations at 100 kHz, 25 °C, and zero DC bias.

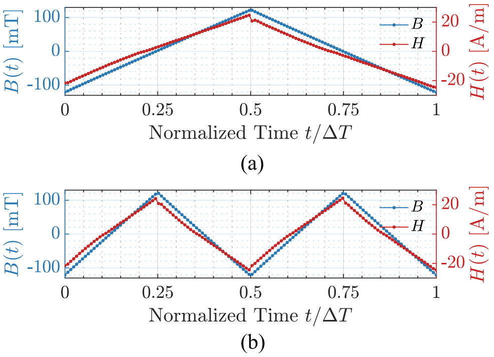


Fig. 9. Examples of the multicycle data augmentation. (a) Original single-cycle waveform at 125 kHz. (b) Augmented two-cycle waveform at the effective frequency of 62.5 kHz. A well designed and well trained neural network should be able to predict similar results for both two cases.

periodic steady state, while maintaining a constant sequence length across all frequencies. The single-cycle interpolation is essentially removing the time stamp information from the original waveform, which is another reason why the fundamental frequency f is included as one of the network inputs.

B. Phase-Shifting Augmentation

Through the single-cycle interpolation, it is assumed that the original waveform can be well reconstructed based on the single-cycle waveform since all the waveforms are captured during the periodic steady state. Furthermore, we hypothesize that the predicted magnetic behaviors, such as the B–H loop and core loss, remain the same regardless of where the waveform is sliced into sections since they all reconstruct to the same original waveform. This hypothesis also indicates that the predicted results should not be influenced by the starting phase of the single-cycle waveform.

To prevent the neural network from misunderstanding the phase information, phase-shifting data augmentation is applied to the single-cycle waveforms. Each pair of $B(t)$ and $H(t)$ waveforms is circularly shifted with a random phase at the same time, which changes the starting phase of the waveform while preserving the original phase difference between the $B(t)$ and $H(t)$ waveforms. Fig. 8 shows a set of example $B(t)$ waveforms of N87 ferrite before and after the phase shifting augmentation.

Phase-shifting augmentation is also applied to balance the dataset distribution. Given the fact that different shapes of waveforms have different degrees of freedom (e.g., amplitude, frequency, and duty ratio), the amount of data for each shape of waveform provided in the original MagNet database significantly differs from each other. For example, the N87 ferrite material dataset contains 142 871 pairs of $B(t)$ and $H(t)$ waveforms measured under different frequency f, temperature T, and dc bias H_{dc} conditions. Among them, the sinusoidal wave, triangular wave, and trapezoidal wave contribute 3,495 (2.45%), 46 973 (32.87%), and 92 403 (64.68%) pairs, respectively. The sinusoidal wave, in particular, has much fewer samples compared with the other two waveform shapes. As a result, training with this unbalanced dataset leads to biased accuracy for sinusoidal excitations. Through phase-shifting augmentation, one can assign multiple phase values for the sinusoidal waves to augment the data while keeping the augmented waveforms distinguishable from each other.

C. Multicycle Augmentation

In addition to the initial phase, the frequency is a crucial factor affecting the division of the full-length waveform into single-cycle sections. In Section V-A, the waveform sequence is sliced based on the fundamental frequency such that each section contains a complete cycle of the waveform. Alternatively, the sequence can also be divided into sections based on $1/N$ of the fundamental frequency, resulting in each section containing N cycles of the waveform. Theoretically, for any integer number of N, the sliced sections are always able to reconstruct the same full-length sequence, and thus, the same B–H loop, except that the resolution within each cycle is reduced due to the fixed-length interpolation.

Based on this hypothesis, the dataset is further augmented by incorporating multicycle waveforms. Fig. 9 shows an example of the two-cycle data augmentation, where the augmented sequence contains two cycles of the waveform and the frequency is halved, correspondingly. With the support of multicycle augmentation, the neural network model is expected to predict approximately equivalent B–H loops and core losses, regardless of whether single-cycle or multicycle input sequences are provided. This augmentation further enhances the model's generalization capability for certain types of waveforms that are not covered by the training dataset.

VI. Training and Testing Results

The LSTM-based and transformer-based models are synthesized using the PyTorch framework. Hyperparameters of the network are determined and optimized based on experimental training results. In the LSTM-based model, both the encoder and the decoder are implemented with a one-layer 32-D LSTM network. In the transformer-based model, the model dimension is set to 24 and the number of attention heads is set to four. In both models, the projector is implemented as a three-layer FNN, with 40 hidden neurons in each layer. These hyperparameters

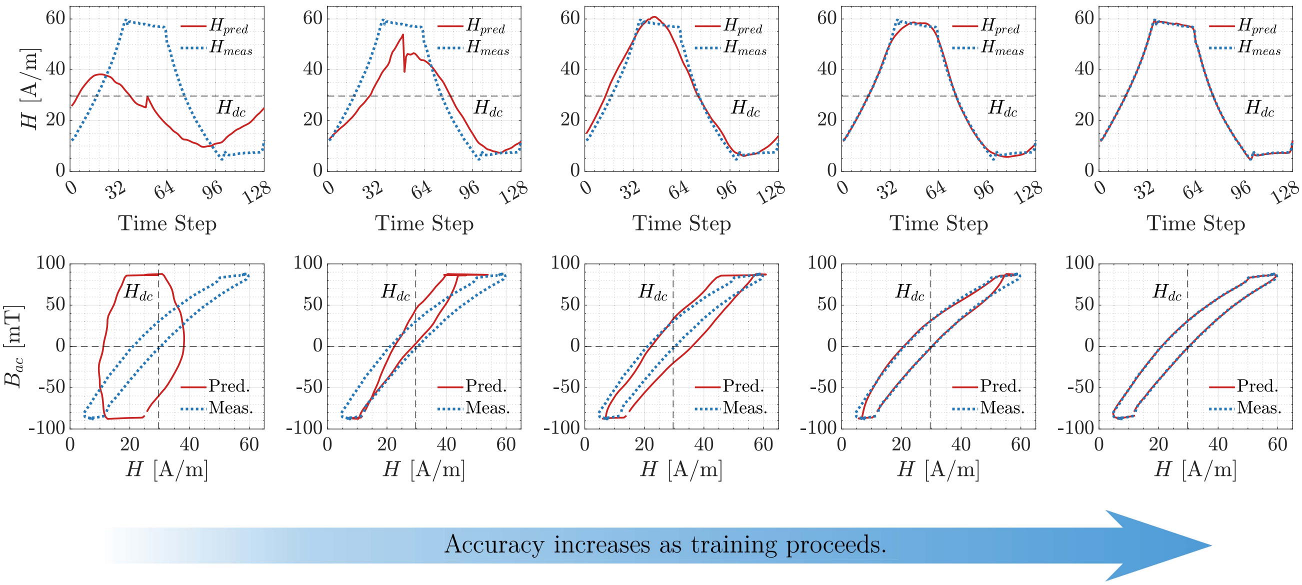


Fig. 10. Prediction results of the $H(t)$ waveform and the B–H loop of an example testing point (trapezoidal, 140 kHz, 90 °C, 30 A/m DC bias) at different stages of the training. The mismatch is minimized as the training proceeds until a good match is achieved between the predicted and the measured waveforms.

result in the total of 28 225 and 28 481 learnable parameters in the LSTM-based model and the transformer-based model, respectively.

The proposed neural network model is trained for 5,000 epochs on the standard Google Colab Pro GPU devices with the MagNet dataset. After applying the data augmentation, the size of the N87 ferrite dataset is expanded to 269 940. These data points are further randomly split into 70%, 20%, and 10% for the training, validation, and test sets. During training, the mean-squared error between the predicted sequence $H_{\text{pred}}(t)$ and the measured sequence $H_{\text{meas}}(t)$ is selected as the loss function for backpropagation. The test dataset, which is never used for training the model, is used to evaluate the model performance. The optimizer used in the model training is configured as Adam optimizer. An exponentially decayed learning rate strategy is implemented to yield a better model convergence, where the initial learning rate is 0.004 and the decaying rate is 90% per 150 epochs. The typical elapsed time for the network training is approximately 20 h for each material (using Google Colab Pro), which can be further accelerated by adopting parallel computing.

A. Hysteresis B-H Loop Prediction

We evaluate the performance of two trained models on the test set to validate their ability to predict the B-H hysteresis loop. Fig. 10 shows a series of particular prediction results generated by the transformer-based model for an example testing point (trapezoidal, 140 kHz, 90 °C, 30 A/m dc bias) at different stages of the training. As the training proceeds, the model gradually converges and the discrepancy between the predicted and the measured hysteresis loops is minimized, eventually achieving a good match.

To quantitatively evaluate the prediction accuracy of the models, the relative error between the predicted sequence $H_{\text{pred}}(t)$ and the measured sequence $H_{\text{meas}}(t)$, as is defined below, is used as the figure of metric to evaluate the results:

$$\text{Relative Err. of Sequence} = \frac{\text{rms}(H_{\text{pred}} - H_{\text{meas}})}{\text{rms}(H_{\text{meas}})}$$

$$= \frac{\sqrt{\frac{1}{n}\sum_{t=t_1}^{tn}(H_{\text{pred}}(t) - H_{\text{meas}}(t))^2}}{\sqrt{\frac{1}{n}\sum_{t=t_1}^{tn}(H_{\text{meas}}(t))^2}}. \quad (2)$$

Fig. 11 shows the distribution of relative errors in the $H(t)$ predictions generated by the LSTM-based model and the transformer-based model. As shown in the figure, both models accurately predict the $H(t)$ sequences. The average relative error for the LSTM-based model is 4.52% and the transformer model 2.99%, while the 95th percentiles are 10.92% and 6.48%, respectively, as listed in Table I. The test set covers data points with all three types of waveform shapes and across the same ranges of frequency, temperature, and dc bias as the training set. These statistics on the prediction results validate that both the proposed models are capable of making accurate predictions for the hysteresis loops under various operating conditions. Under the given hyperparameter settings, the transformer-based model outperforms the LSTM-based model, demonstrating lower overall relative error in terms of the hysteresis loop prediction.

B. Core Loss Prediction

We assess the performance of two trained models to validate their ability to predict the core loss. Based on the predicted B–H loop, one can directly calculate the predicted core loss P_V based on the following integral. Then, the relative error between the predicted core loss $P_{V,\text{pred}}$ and the measured core loss $P_{V,\text{meas}}$ can be calculated, which is used as another figure of metric for evaluating the model performance

$$P_V = \frac{1}{T}\int_{B(0)}^{B(T)} H(t)\,dB(t) \quad (3)$$

TABLE I
COMPARISON OF THE LSTM AND TRANSFORMER IMPLEMENTATION OF THE ENCODER–PROJECTOR–DECODER ARCHITECTURE

Model Type	Average of Relative Error	95th Percentile of Relative Error	Number of Parameters	Elapsed Time of Model Inference
LSTM	4.52%	10.92%	28,225	0.017 ms/data point
Transformer	2.99%	6.48%	28,481	2.584 ms/data point

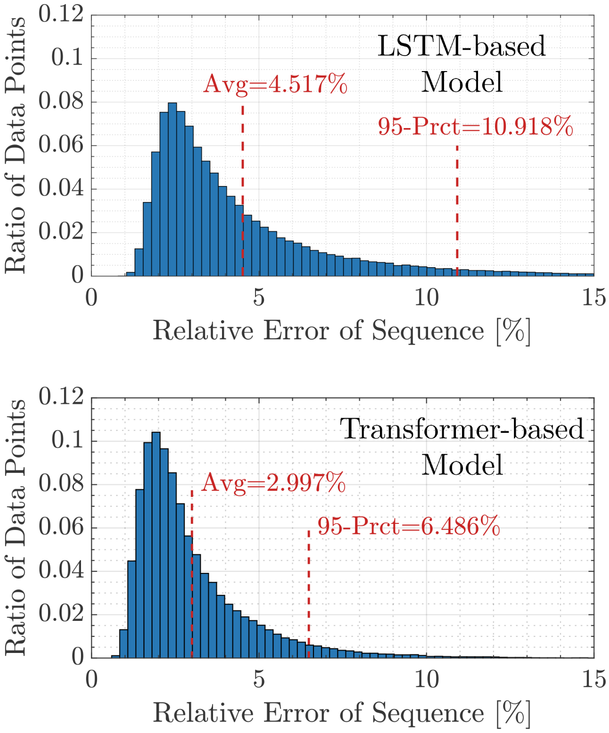


Fig. 11. Relative error distributions of the predicted $H(t)$ sequence generated by the LSTM-based and the transformer-based neural network models.

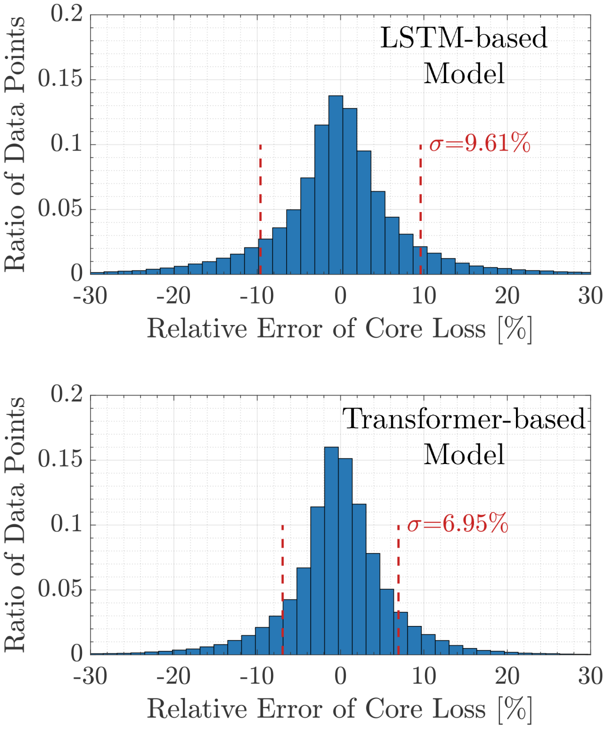


Fig. 12. Relative error distributions of the predicted core loss generated by the LSTM-based and the transformer-based neural network models.

$$\text{Relative Err. of Core Loss} = \frac{|P_{V,\text{pred}} - P_{V,\text{meas}}|}{P_{V,\text{meas}}}. \quad (4)$$

Fig. 12 shows the error distribution of the core loss predicted by the LSTM-based and the transformer-based models. As shown in the figure, for most of the data points in the test set, the model is capable of accurately predicting the core loss as well. The standard deviation of the relative error for the LSTM-based model is 9.61% and the transformer model 6.95%, while the 95th percentiles are 13.69% and 10.02%, respectively. Under the given hyperparameter settings, the transformer-based model also outperforms the LSTM-based model and demonstrate lower overall relative error in terms of the core loss prediction.

It is noticed that the overall relative error of core loss prediction is higher than that of hysteresis loop prediction. According to the selected loss function during the training, the model is optimized to minimize the shape discrepancy between the predicted sequence $H_{\text{pred}}(t)$ and the measured sequence $H_{\text{meas}}(t)$, while any information about the core loss is not directly available to the network for reference. However, core loss calculation is highly sensitive to phase mismatch between $H(t)$ and $B(t)$, and an approximately matched sequence does not necessarily result in a good match for core loss. To further improve the accuracy of core loss prediction while maintaining high accuracy for hysteresis loop prediction, one can introduce the core loss information or the phase information into the loss function of training, which however may inevitably increase the computational cost.

C. Comparison and Discussion

Table I presents a comparative analysis between the LSTM and transformer implementations in terms of their hysteresis loop prediction accuracy, model size, and approximate computational cost. Both models are trained and tested with the same training set and test set for the same number of epochs.

With the given hyperparameter settings, the overall prediction accuracy of the transformer-based model is better than that of the LSTM-based model. Fig. 13 shows a few examples of the predicted B–H loops generated by each model against the measured ones under various frequency, temperature, and dc bias conditions, with multiple waveform shapes. Both models accurately predict the shape and location of the majority part of B–H loops, while the sharp corners are better captured by the transformer-based model, benefiting from the attention mechanism.

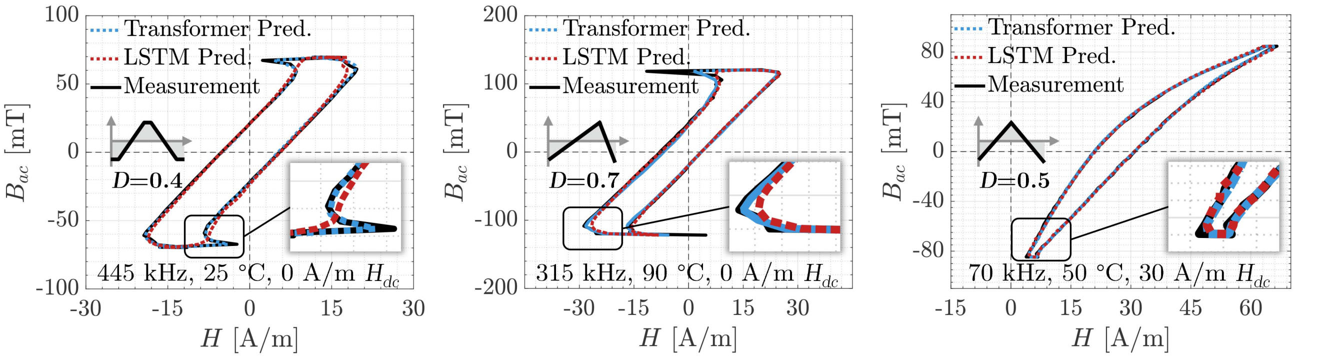


Fig. 13. Examples of the predicted B–H loops under different frequency, temperature, and DC bias conditions, with multiple waveform shapes. Both the LSTM-based and the transformer-based models accurately predict the majority part of B–H loops, while the sharp corners are better captured by the transformer-based model.

TABLE II
COMPARISON OF THE THEORETICAL COMPUTATIONAL COST BETWEEN THE LSTM AND THE SELF-ATTENTION (TRANSFORMER)

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Recurrent (LSTM)	$\mathcal{O}(n \cdot d^2)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Self-Attention (Transformer)	$\mathcal{O}(n^2 \cdot d)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$

The LSTM-based model requires less elapsed time for the model inference. Table II provides a theoretical comparison of the computational complexity of both models [29], where n represents the sequence length and d is the dimension of the model. In our testing cases, both models have $n = 128$, while the LSTM model and transformer model have $d = 32$ and 24, respectively. Therefore, despite that the transformer-based model reduces the sequential operation and maximum path length by avoiding recurrent operations, it suffers a much higher complexity per layer, resulting in a longer training and execution time.

Considering the tradeoff between the prediction accuracy and the execution time, the transformer-based model is selected to establish the neural network-aided smart datasheet. All the results in the following sections are generated by the transformer-based model.

VII. MagNet-AI: A NN-Aided Smart Datasheet

Evaluations results validate the neural network model's effectiveness of predicting the hysteresis loop and core loss under various operating conditions and excitation waveforms. To establish a neural network-aided smart datasheet, the neural network model is packaged into a function for rapid inference, where the inputs are the waveform of flux density $B(t)$, the frequency f, the temperature T, and the dc bias field strength H_{dc} while the output is the waveform of the field strength $H(t)$. The flowchart of the neural network-aided smart datasheet is shown in Fig. 14.

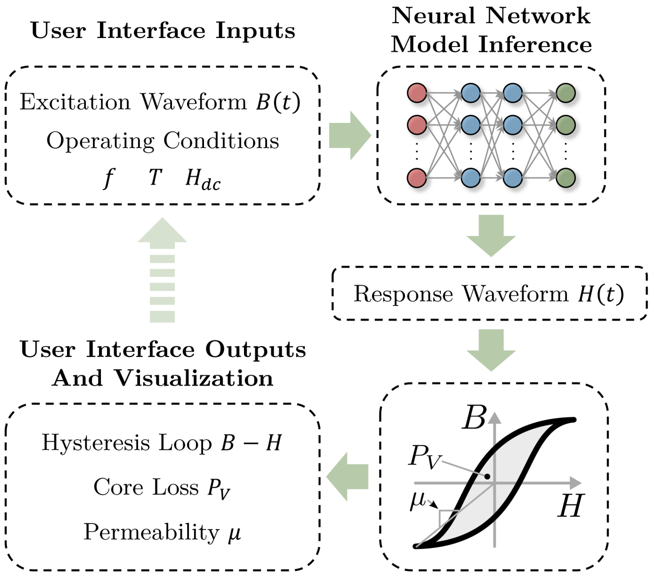


Fig. 14. Flowchart of the neural network-aided smart datasheet. Users can specify the excitation waveform and the operating conditions through the user interface as the inputs of the neural network model. The model inference is executed to predict the response waveform. After post-processing, the prediction results, e.g., hysteresis loop, core loss, and permeability, will be visualized and provided to users.

Here are several prediction examples to demonstrate different ways of using the NN-aided smart datasheet. In each example, a manually generated dataset is fed into the neural network model as the inputs, where the excitation waveforms are in ideal shapes and the operating conditions are swept. Note the waveforms in the manually generated datasets are pure waves without any nonideal effects, such as switching transitions, which naturally leads to a slightly different prediction results compared with the measurements, despite the close resemblance between the two.

1) *Example-1:* Predicting the hysteresis loop at different flux density amplitudes. In this example, the excitation waveforms are a set of 50% duty ratio pure triangular waves, where the amplitude is swept from 30 to 240 mT. The frequency, the temperature, and the dc bias are fixed at 100 kHz, 25 °C, and 0 A/m, respectively. Fig. 15 shows the predicted B–H loops with this manually generated

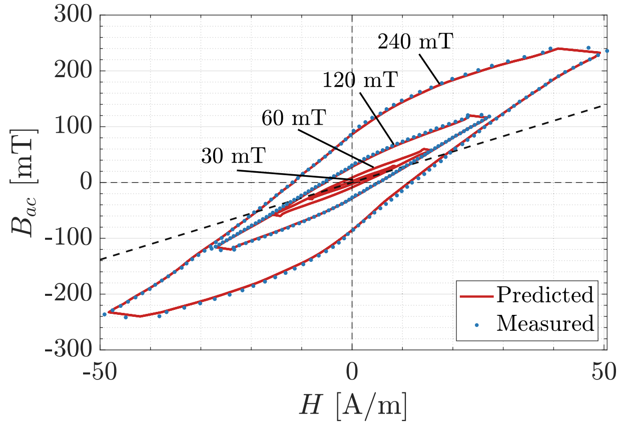


Fig. 15. Predicted B–H loops with the manually generated model inputs with 50% duty ratio pure triangular waves, where the amplitude of flux density is sweeping from 30 to 240 mT. The frequency, the temperature, and the DC bias are fixed at 100 kHz, 25 °C, and 0 A/m, respectively.

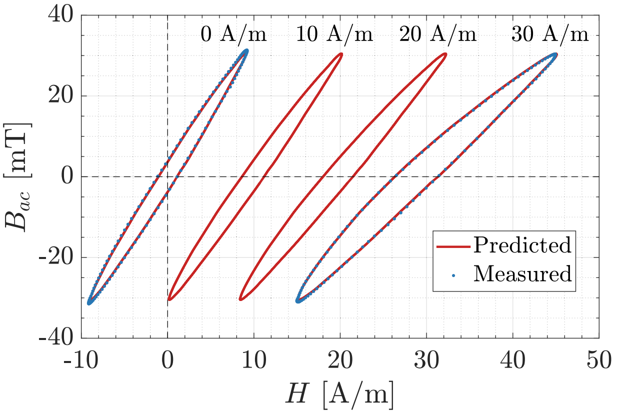


Fig. 17. Predicted B–H loops with the manually generated model inputs with pure sinusoidal waves, where the DC bias is sweeping from 0 to 30 A/m. The amplitude, the frequency, and the temperature are fixed at 30 mT, 200 kHz, and 25 °C, respectively.

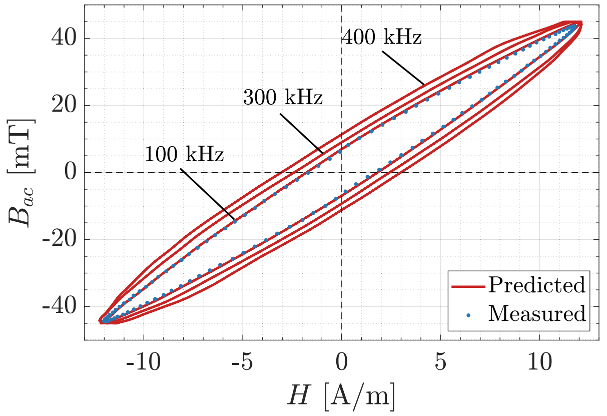


Fig. 16. Predicted B–H loops with the manually generated model inputs with pure sinusoidal waves, where the fundamental frequency is sweeping from 100 to 400 kHz. The amplitude, the temperature, and the DC bias are fixed at 45 mT, 25 °C, and 0 A/m, respectively.

dataset as model inputs. It is observed that the impact of the flux density amplitude on the hysteresis loop is well captured and predicted by the neural network model, and a good match is achieved with respect to the adjacent measured hysteresis loops. At small amplitude, the B–H approximately aligns with the straight line $B = \mu_i H$, where μ_i is the initial permeability of the material. As the amplitude increases, the B–H loop is enlarged and gradually saturated, resulting in a much larger core loss and very different permeability.

2) *Example-2:* Predicting the hysteresis loop at different frequencies. In this example, the excitation waveforms are a set of pure sinusoidal waves, with an amplitude of 45 mT. The temperature and the dc bias are fixed at 25 °C and 0 A/m, respectively, while the frequency is sweeping from 100 to 400 kHz. Fig. 16 shows the predicted B–H loops with this manually generated dataset as model inputs. It is observed that the impact of the fundamental frequency on the hysteresis loop is well captured and predicted by the neural network model, and a good match is achieved with respect to the adjacent measured hysteresis loops. As the frequency increases, the B–H loop is enlarged, resulting in a larger core loss energy per cycle.

3) *Example-3:* Predicting the hysteresis loop at different levels of dc bias. In this example, the excitation waveforms are a set of pure sinusoidal waves, with an amplitude of 30 mT. The frequency and the temperature are fixed at 200 kHz and 25 °C, respectively, while the dc bias is swept from 0 to 30 A/m. Fig. 17 shows the predicted B–H loops with this manually generated dataset as model inputs. It is observed that the impact of the dc bias on the hysteresis loop is well captured and predicted by the neural network model, and a good match is achieved with respect to the adjacent measured hysteresis loops. As the dc bias increases, the B–H loop is enlarged and tilted.

4) *Example-4:* Predicting the core loss under triangular waves with different duty ratios. In this example, the excitation waveforms are a set of pure triangular waves, with an amplitude of 43.5 mT while the duty ratio is swept from 10% to 90%. The frequency, the temperature, and the dc bias are fixed at 315 kHz, 25 °C, and 0 A/m, respectively. Fig. 18 shows the predicted core loss curves with this manually generated dataset as model inputs. It is observed that the basic relationship between the duty ratio and the core loss is well captured and predicted by the neural network model, and a good match is achieved with respect to the adjacent measured core loss. For triangular waves, the core loss reaches a minimum when duty ratio $D = 0.5$, and increases when it approaches 0 or 1. The core losses for duty ratio of D and $1 - D$ are approximately the same, resulting in a symmetric core loss curve versus the duty ratio.

5) *Example-5:* Predicting the core loss at different temperatures. In this example, the excitation waveforms are a set of pure trapezoidal waves, whose duty ratios for rising and falling are both 20%. The amplitude of flux density is fixed at 35, 70, and 140 mT, separately. The frequency and the dc bias are fixed at 100 kHz and 0 A/m, respectively,

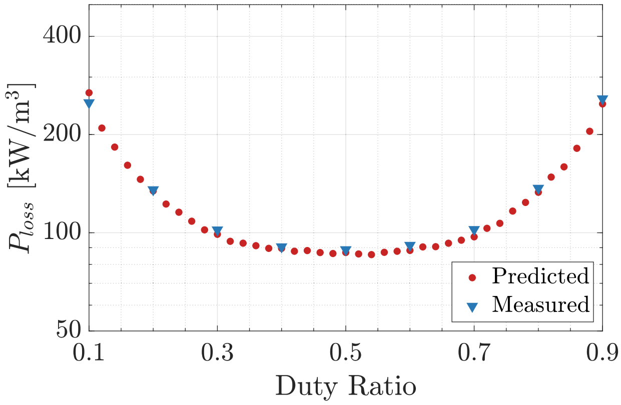


Fig. 18. Predicted core loss curves with the manually generated model inputs, where the duty ratio of the triangular wave is sweeping from 10% to 90%. The amplitude, the frequency, the temperature, and the DC bias are fixed at 43.5 mT, 315 kHz, 25 °C, and 0 A/m, respectively.

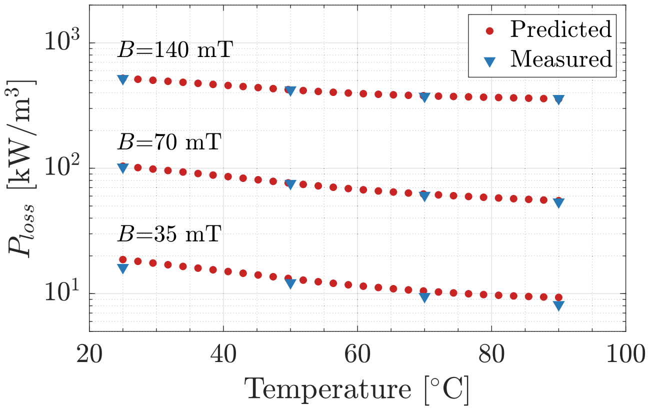


Fig. 19. Predicted core loss curves with the manually generated model inputs, where the flux density is fixed at 35, 70, and 140 mT, and the temperature is sweeping from 25 °C to 90 °C. The frequency and the dc bias are fixed at 100 kHz and 0 A/m, respectively.

while the temperature is swept from 25 °C to 90 °C. Fig. 19 shows the predicted core loss curves with this manually generated dataset as model inputs. It is observed that the basic relationship between the temperature and the core loss is well captured and predicted by the neural network model, and a good match is achieved with respect to the adjacent measured core loss. As the temperature rises, the core loss is reduced.

With the capability of predicting the hysteresis loop and core loss under various operating conditions, as demonstrated in the above examples, the proposed neural network model can be potentially used as an alternative to the conventional datasheet or measurement dataset. Notably, the neural network models can greatly reduce the size of the dataset with negligible loss of accuracy. For the N87 material in this work, the size of the postprocessed dataset for model training is 3.8 GB, while the size of the transformer model is only 204 kB, which almost equivalently describes the behaviors of magnetic materials, and much more comprehensive than the conventional datasheet with similar file sizes. Users of the neural network aided datasheet can rapidly predict the behavior of the magnetic material, such as the hysteresis loop, the permeability, and the core loss, by specifying the excitation waveforms and the operation conditions, without any needs of time consuming data extraction and complicated interpolation as dealing with the conventional datasheet. Distinguished from the conventional datasheet, the neural network-aided smart datasheet is well packaged into a function, which makes it feasible to be integrated into any other iteration calculations, such as multiobjective optimization algorithm or AI-mag [41].

VIII. NN-Aided Material Recommendation

Benefiting from the fast model inference capabilities of neural networks, the proposed model can also be used to assist the material comparison and selection for specific excitations and operating conditions. It can rapidly rank magnetic materials across a wide operation range. The transformer-based neural network model has been trained on all the ten materials from the MagNet database, including TDK{N27, N30, N49, N87}, Ferroxcube{3C90, 3C94, 3E6, 3F4}, and Fair-Rite{77, 78}. The model was trained using a similar large measurement dataset as shown in Section V, with the same network hyperparameters and data augmentation techniques. When a specific operating condition is provided, these neural network models can be executed sequentially to sweep across all the materials and calculate the corresponding core loss for each material. By sorting the core loss values among all the candidate materials, MagNet-AI can recommend the best-performing material candidates for the given operating condition. Two examples of material ranking maps are provided here to illustrate the effectiveness of the neural network-aided material comparison.

1) *Example-1:* Selecting the optimal material at different levels of dc bias across a wide range of flux density and frequency. In this example, the excitation waveforms consist of a set of 50% duty ratio pure triangular waves. The amplitude and frequency of the waveforms are swept from 30 to 200 mT and 50 to 500 kHz, respectively. The dc bias is selected from three different levels, namely 0, 10, and 20 A/m, while the temperature remains fixed at 25 °C. Fig. 20 shows the material ranking maps for each level of dc bias, where different colors stand for different materials, whichever achieve the lowest core loss under each operating conditions. Each material has its optimal operation range in terms of frequency, flux density, and dc bias. N30 ferrite demonstrates lower core loss at low frequency, whereas N49 ferrite dominates at higher frequency. The boundary is moving as the dc bias changes.
2) *Example-2:* Selecting the optimal material at different temperatures across a wide range of flux density. In this example, the excitation waveforms also consist of a set of 50% duty ratio pure triangular waves. The amplitude of the waveforms and the temperature are swept from 30 to 200 mT and 25 °C–90 °C, respectively. The frequency remains fixed at 150 kHz and the dc bias is zero. Fig. 21 shows the corresponding material ranking map. As depicted in the map, different materials also have their optimal operation ranges in terms of temperature.

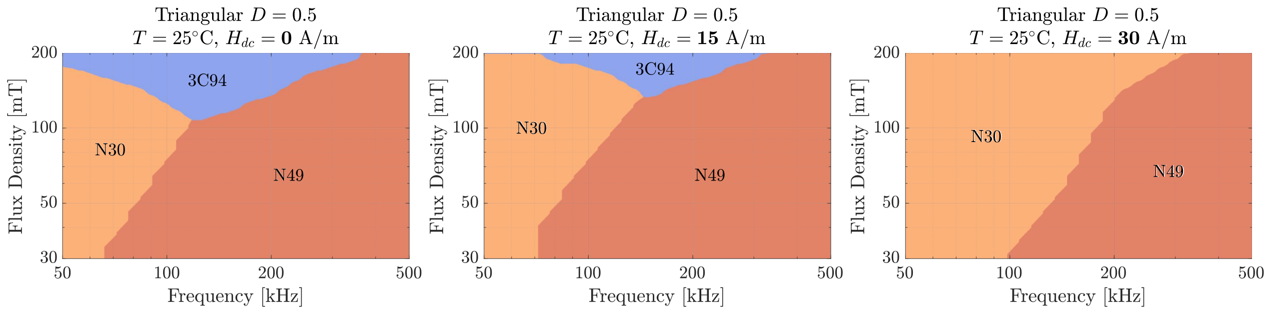


Fig. 20. Material ranking map at different levels of DC bias across a wide range of flux density amplitude and frequency.

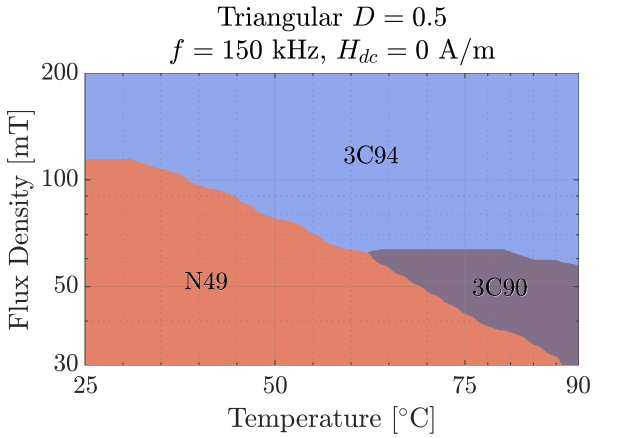


Fig. 21. Material ranking map across a wide range of flux density amplitude and temperature.

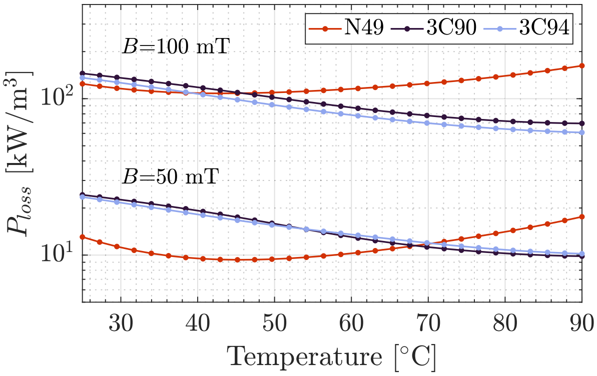


Fig. 22. Core loss curve comparisons at different levels of flux density amplitude and temperatures.

At low temperature, the material map is dominated by N49 and 3C94 ferrite. As the temperature increases, 3C90 ferrite begins to show the superiority. More specifically, Fig. 22 shows the predicted core loss curves for the three aforementioned materials across different temperatures. It can be observed that N49 ferrite achieves its minimum core loss at a relatively low temperature, while the other two materials are more suitable for high temperature applications.

As demonstrated, given a targeted operation range, the neural network model can effectively assist the designer to determine, which material offers the most desirable performance for the particular operating conditions. With the constantly expanding material category, the neural network model will provide design recommendations among various materials with only linearly increasing computational cost.

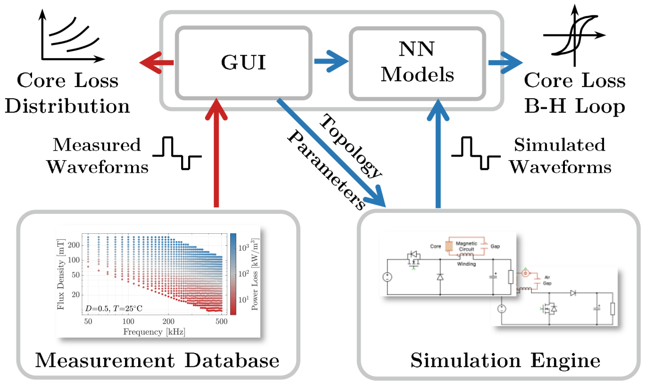
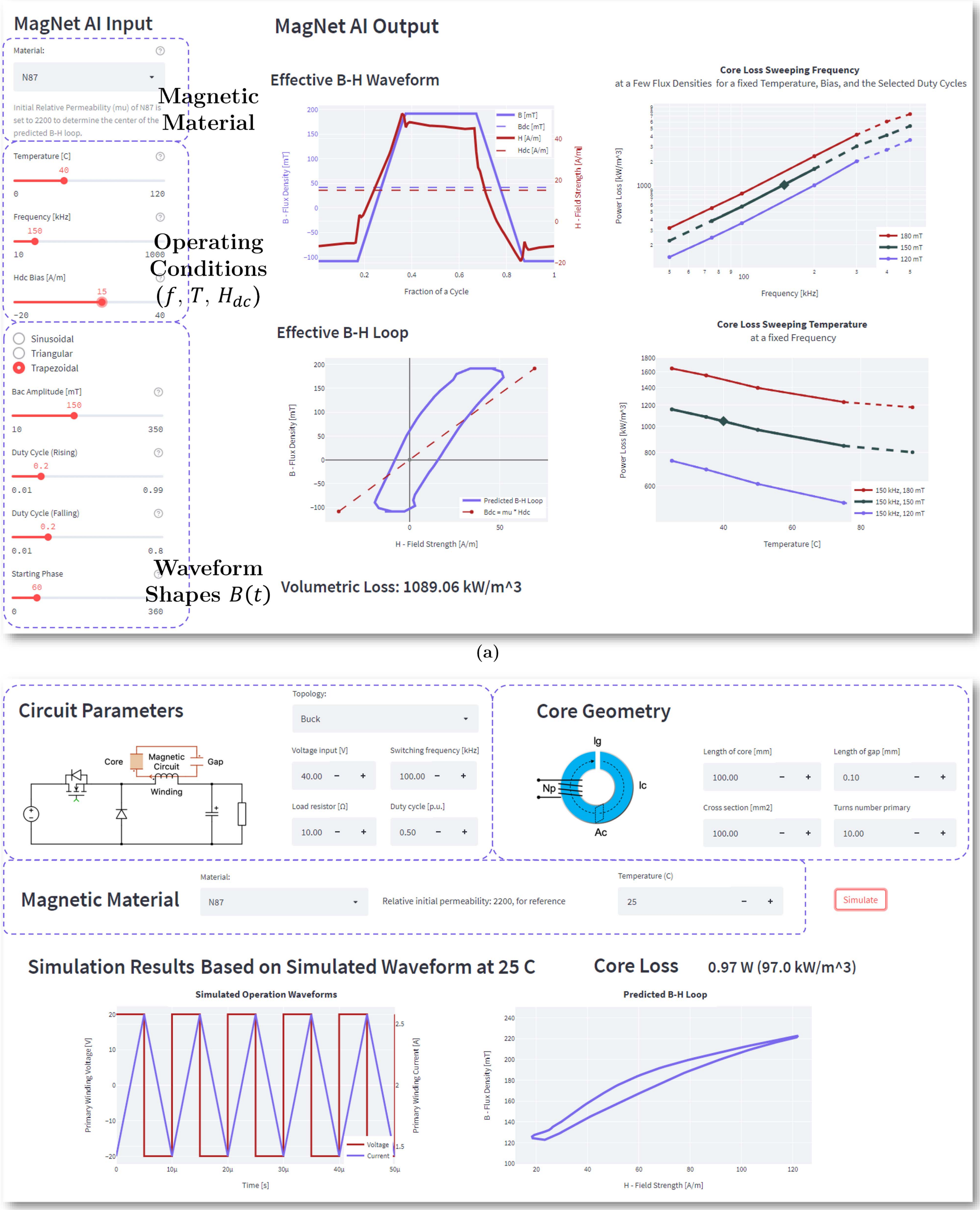


Fig. 23. Website architecture and information flow of the MagNet webpage platform, which provides users with access to download and visualize the measured data in the MagNet core loss database, as well as analyze and simulate the magnetic behaviors with the deployed neural network models and the PLECS simulation engine.

IX. Online Smartsheet Platform

To enable the interactive datasheet inference based on the proposed neural network model, an open-source webpage-based platform with graphic user interface (GUI) is designed and developed. It is powered by Streamlit (an open-source app framework of Python for website deployment), and shared in GitHub, offering a variety of data-visualization tools with a GUI for the database access, magnetic core loss estimation, hysteresis loop prediction, and circuit simulation, as well as the access to download all the measured data points. The neural network model and the circuit simulation engine are deployed on the website, which allow users to predict the hysteresis loop under any user-defined conditions or simulation conditions. The website architecture and information flow of the website platform are shown in Fig. 23.

Fig. 24(a) shows an example screenshot of the smartsheet session of the neural network model. With the GUI, users may specify the type of magnetic material, operating conditions (temperature, frequency, and dc bias), and the excitation waveform



MagNet AI Input

Material: N87

Initial Relative Permeability (mu) of N87 is set to 2200 to determine the center of the predicted B-H loop.

Temperature [C]: 40 (0 – 120)

Frequency [kHz]: 150 (10 – 1000)

Hdc Bias [A/m]: 15 (-20 – 40)

○ Sinusoidal
○ Triangular
● Trapezoidal

Bac Amplitude [mT]: 150 (10 – 350)

Duty Cycle (Rising): 0.2 (0.01 – 0.99)

Duty Cycle (Falling): 0.2 (0.01 – 0.8)

Starting Phase: 60 (0 – 360)

MagNet AI Output

Volumetric Loss: 1089.06 kW/m^3

(a)

Circuit Parameters

Topology: Buck

Voltage input [V]: 40.00

Switching frequency [kHz]: 100.00

Load resistor [Ω]: 10.00

Duty cycle [p.u.]: 0.50

Core Geometry

Length of core [mm]: 100.00

Length of gap [mm]: 0.10

Cross section [mm2]: 100.00

Turns number primary: 10.00

Magnetic Material

Material: N87

Relative initial permeability: 2200, for reference

Temperature (C): 25

Simulate

Simulation Results Based on Simulated Waveform at 25 C

Core Loss 0.97 W (97.0 kW/m^3)

(b)

Fig. 24. Example screenshot of the MagNet-AI webpage-based magnetics analysis and prediction platform. (a) Smartsheet session. (b) Simulation session. MagNet-AI is available at: https://mag-net.princeton.edu, and open-sourced at: https://github.com/PrincetonUniversity/magnet

(either standard shapes or user defined waves). The neural network engine can rapidly predict the B–H loop under the given conditions, simultaneously generating a sequence of core loss curves around the operating point. This procedure provides a far more comprehensive dataset in comparison with conventional datasheets.

The webpage is also connected to a circuit simulation server hosted by Plexim. The webpage feeds information to the server, and the server returns inputs to the machine learning algorithms in combination with power converter operations, as shown in Fig. 24(b). Users can choose from a pool of common topologies (buck, boost, flyback, dual-active bridge), specify the circuit parameters, magnetic component specifications, and operating conditions, then the simulation engine simulates and outputs the excitation waveform of the magnetic component. The MagNet server collects the waveform and predicts the core loss using the proposed neural network models. Iterations between the neural network model and the simulation engine will be implemented in the future to achieve more accurate simulation results by capturing the nonlinear effects. Note the flux density is calculated based on the specified geometrical parameters assuming a uniform flux distribution. The geometry impact is not considered in this work. The neural network models can be integrated with a circuit simulator to enable magnetic-in-circuit simulations.

Besides the neural network-aided smart datasheet, the website also provides the database session, which allows the raw measurement dataset to be visualized in many ways, and enables rapid comparison of the core loss and B–H loop data of different materials. The user may specify the type of magnetic material, together with the excitation waveforms and the operating conditions. The website backend searches for the requested data in the database and visualizes it in the way that the user selects. The website also provides download access to the raw data being collected from the equipment before any postprocessing with the test conditions documented, and the postprocessed dataset files for data-driven modeling applications.

The MagNet platform is constantly maintained and updated with new data and neural network models. Extended details are included on the website to enable trustworthy repeating measurements and cross validation of the dataset.

X. Conclusion

This article introduces the concept of neural network as datasheet for modeling magnetics across wide operation range. We proposes an encoder–projector–decoder neural network architecture for B–H loop modeling of power magnetics. The proposed architecture is implemented based on both the LSTM network and the transformer network, effectively combining both sequence inputs (excitation waveforms) and scalar inputs (operation conditions) for hysteresis loop modeling. Experimental results prove that the neural network is capable of accurately predicting the B–H loop and the corresponding core loss for ferrite materials. Several applications of the neural network-aided datasheet are demonstrated, including B–H loop and core loss prediction, and material recommendation across wide operation range, with a fully functioning webpage-based online smartsheet platform. The neural network-aided datasheet can offer much more comprehensive information and convenient accessibility compared with the conventional datasheet, while maintaining a comparably small file size.

References

[1] D. Serrano et al., "Neural network as datasheet: Modeling B-H loops of power magnetics with sequence-to-sequence LSTM encoder-decoder architecture," in *Proc. IEEE 23rd Workshop Control Model. Power Electron.*, 2022, pp. 1–8.

[2] H. Li, D. Serrano, S. Wang, T. Guillod, M. Luo, and M. Chen, "Predicting the B-H loops of power magnetics with transformer-based encoder-projector-decoder neural network architecture," in *Proc. IEEE Appl. Power Electron. Conf. Expo.*, 2023, pp. 1–8.

[3] D. J. Perreault et al., "Opportunities and challenges in very high frequency power conversion," in *Proc. 24th Annu. IEEE Appl. Power Electron. Conf. Expo*, 2009, pp. 1–14.

[4] C. R. Sullivan, B. A. Reese, A. L. F. Stein, and P. A. Kyaw, "On size and magnetics: Why small efficient power inductors are rare," in *Proc. Int. Symp. 3D Power Electron. Integr. Manuf.*, 2016, pp. 1–23.

[5] Y. Han, G. Cheung, A. Li, C. R. Sullivan, and D. J. Perreault, "Evaluation of magnetic materials for very high frequency power applications," *IEEE Trans. Power Electron.*, vol. 27, no. 1, pp. 425–435, Jan. 2012.

[6] A. J. Hanson, J. A. Belk, S. Lim, C. R. Sullivan, and D. J. Perreault, "Measurements and performance factor comparisons of magnetic materials at high frequency," *IEEE Trans. Power Electron.*, vol. 31, no. 11, pp. 7909–7925, Nov. 2016.

[7] M. Chen, M. Araghchini, K. K. Afridi, J. H. Lang, C. R. Sullivan, and D. J. Perreault, "A systematic approach to modeling impedances and current distribution in planar magnetics," *IEEE Trans. Power Electron.*, vol. 31, no. 1, pp. 560–580, Jan. 2016.

[8] J. Li, T. Abdallah, and C. Sullivan, "Improved calculation of core loss with nonsinusoidal waveforms," in *Proc. Conf. Rec. IEEE Ind. Appl. Conf. 36th IAS Annu. Meeting*, 2001, vol. 4, pp. 2203–2210.

[9] D. Jiles and D. Atherton, "Theory of ferromagnetic hysteresis," *J. Magnetism Magn. Mater.*, vol. 61, pp. 48–60, 1986.

[10] D. Serrano et al., "Why MagNet: Quantifying the complexity of modeling power magnetic material characteristics," *IEEE Trans. Power Electron.*, early access, Jun. 30, 2023, doi: 10.1109/TPEL.2023.3291084.

[11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[12] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[13] B. Widrow and M. Lehr, "30 years of adaptive neural networks: Perceptron, madaline, and backpropagation," in *Proc. IEEE*, vol. 78, no. 9, pp. 1415–1442, Sep. 1990.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[15] K. Stoyka, N. Femia, and G.Di Capua, "Power inductors behavioral modeling revisited," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 67, no. 12, pp. 5636–5649, Dec. 2020.

[16] A. Oliveri, G. Di Capua, K. Stoyka, M. Lodi, M. Storace, and N. Femia, "A power-loss-dependent inductance model for ferrite-core power inductors in switch-mode power supplies," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 66, no. 6, pp. 2394–2402, Jun. 2019.

[17] H. Li et al., "MagNet: An open-source database for data-driven magnetic core loss modeling," in *Proc. IEEE Appl. Power Electron. Conf. Expo.*, pp. 588–595.

[18] H. Li et al., "How magnet: Machine learning framework for modeling power magnetic material characteristics," *IEEE Trans. Power Electron.*, accepted, 2023, doi: 10.1109/TPEL.2023.3309232.

[19] H. Saliah, D. Lowther, and B. Forghani, "A neural network model of magnetic hysteresis for computational magnetics," *IEEE Trans. Magn.*, vol. 33, no. 5, pp. 4146–4148, Sep. 1997.

[20] C. Serpico and C. Visone, "Magnetic hysteresis modeling via feed-forward neural networks," *IEEE Trans. Magn.*, vol. 34, no. 3, pp. 623–628, May 1998.

[21] E. Fallah, J. Moghani, and H. Talebi, "Modeling of magnetic hysteresis using counter propagation neural network," in *Proc. IEEE Int. Magn. Conf.*, 2003, pp. DP–07.

[22] H. Saliah, D. Lowther, and B. Forghani, "Modeling magnetic materials using artificial neural networks," *IEEE Trans. Magn.*, vol. 34, no. 5, pp. 3056–3059, Sep. 1998.

[23] F. Sixdenier, R. Scorretti, R. Marion, and L. Morel, "Quasistatic hysteresis modeling with feed-forward neural networks: Influence of the last but one extreme values," *J. Magnetism Magn. Mater.*, vol. 320, no. 20, pp. e992–e996, 2008.

[24] Z. Zhao, F. Liu, S. L. Ho, W. N. Fu, and W. Yan, "Modeling magnetic hysteresis under dc-biased magnetization using the neural network," *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3958–3961, Oct. 2009.

[25] F. Riganti Fulginei and A. Salvini, "Neural network approach for modelling hysteretic magnetic materials under distorted excitations," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 307–310, Feb. 2012.

[26] C. Grech, M. Buzio, M. Pentella, and N. Sammut, "Dynamic ferromagnetic hysteresis modelling using a Preisach-recurrent neural network model," *Materials*, vol. 13, no. 11, 2020, Art. no. 2561.

[27] S. Quondam Antonio, F. Riganti Fulginei, A. Laudani, A. Faba, and E. Cardelli, "An effective neural network approach to reproduce magnetic hysteresis in electrical steel under arbitrary excitation waveforms," *J. Magnetism Magn. Mater.*, vol. 528, 2021, Art. no. 167735.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[29] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, vol. 30, pp. 6000–6010.

[30] G. Skutt and F. Lee, "Characterization of dimensional effects in ferrite-core magnetic devices," in *Proc. PESC Record. 27th Annu. IEEE Power Electron. Specialists Conf.*, 1996, vol. 2, pp. 1435–1440.

[31] A. Nabih, F. Jin, R. Gadelrab, F. C. Lee, and Q. Li, "Characterization and mitigation of dimensional effects on core loss in high-power high-frequency converters," *IEEE Trans. Power Electron.*, early access, Jun. 13, 2023, doi: 10.1109/TPEL.2023.3285633.

[32] M. Kacki, M. S. Rylko, J. G. Hayes, and C. R. Sullivan, "Analysis and experimental investigation of high-frequency magnetic flux distribution in Mn-Zn ferrite cores," *IEEE Trans. Power Electron.*, vol. 38, no. 1, pp. 703–716, Jan. 2023.

[33] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Phys. D: Nonlinear Phenomena*, vol. 404, 2020, Art. no. 132306.

[34] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015. Accessed: Aug. 13, 2018. [Online]. Available: http://arxiv.org/abs/1511.08458

[35] J. Wang, Q. Liu, Z. Liu, and S. Wu, "Towards accurate and interpretable sequential prediction: A CNN & attention-based feature extractor," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, 2019, pp. 1703–1712.

[36] Y. LeCun and Y. Bengio, *Convolutional Networks for Images, Speech, and Time Series*. Cambridge, MA, USA: MIT Press, 1995.

[37] H. Li, S. R. Lee, M. Luo, C. R. Sullivan, Y. Chen, and M. Chen, "MagNet: A machine learning framework for magnetic core loss modeling," in *Proc. IEEE 21st Workshop Control Model. Power Electron.*, 2020, pp. 1–8.

[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[39] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 338–342.

[40] P. Welch, "The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. 15, no. 2, pp. 70–73, Jun. 1967.

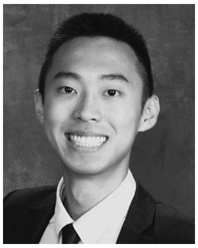
[41] T. Guillod, P. Papamanolis, and J. W. Kolar, "Artificial neural network (ANN) based fast and accurate inductor modeling and design," *IEEE Open J. Power Electron.*, vol. 1, pp. 284–299, 2020.

Haoran Li (Student Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2019 and the M.A. degree in electrical and computer engineering in 2021 from Princeton University, Princeton, NJ, USA, where he is currently working toward the Ph.D. degree in power electronics.

His research interests include the measurement, characterization, and data-driven modeling of power magnetics materials, as well as the design and optimization of dc/dc converters, and the magnetic components for point-of-load and data center applications.

Diego Serrano (Member, IEEE) received the B.S. degree in industrial electronic engineering from the Universidad de Granada, Granada, Spain, in 2015, the master's degree in electronic engineering from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 2016 and the Ph.D. in power electronics from the Universidad Politécnica de Madrid (UPM), in 2021.

He was a Researcher with Centro de Electrónica Industrial, UPM, where he was a Researcher for another year. He is currently a Postdoctoral Research Associate working on the measurement and modeling of magnetic materials with Princeton University, Princeton, NJ, USA. His research interests include multilevel conversion, applications with GaN and SiC devices, and magnetics.



Shukai Wang (Student Member, IEEE) received the B.S. degree in electrical engineering from Michigan State University, East Lansing, MI, USA, in 2021, and the M.A. degree in electrical engineering in 2023 from Princeton University, NJ, USA, where he is currently working toward the Ph.D. degree in power electronics.

His research interests include magnetic modeling and related topics, as well as high-frequency wireless power transfer applications.

Mr. Wang was the recipient of Outstanding Presentation Award in 2023 IEEE APEC.

Minjie Chen (Senior Member, IEEE) received the B.S. in electrical engineering from Tsinghua University, Beijing, China, in 2009, and the S.M., E.E., and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2015.

He is currently an Assistant Professor of electrical and computer engineering with the Andlinger Center for Energy and the Environment, Princeton University, Princeton, NJ, USA. His research team developed the MagNet database, built the MagNet-AI platform, launched the MagNet Challenge, and is leading the open-source MagNet community. His research interests include high-frequency power electronics, power architecture, power magnetics, advanced packaging, data-driven methods, design automation, and the design methods of high-performance power electronics for emerging and important applications.

Dr. Chen was the recipient of the IEEE PELS Richard M. Bass Outstanding Young Power Electronics Engineer Award, the Princeton SEAS E. Lawrence Keyes, Jr./Emerson Electric Co. Junior Faculty Award, the NSF CAREER Award, six IEEE Transactions on Power Electronics Prize Paper Awards, a COMPEL Best Paper Award, an ICRA Best Poster Award, three ECCE Best Demo Awards, a 3D-PEIM Rao R. Tummala Best Paper Award, an OCP Best Paper Award, a Siebel research Award, a C3.AI Research Award, a First Place Award of Princeton Keller Center Innovation Forum, and the MIT EECS D. N. Chorafas Ph.D. Thesis Award. He was listed on the Princeton Engineering Commendation List for Outstanding Teaching multiple times. Dr. Chen is an IEEE Senior Member, the Vice Chair of PELS TC10 Design Methodologies, and the TPC Member of a few flagship PELS conferences including APEC, ECCE, COMPEL, and ICDCM.