A Data-Driven Model for Power Loss Estimation of Magnetic Materials Based on Multi-Objective Optimization and Transfer Learning

Li, Z.; Wang, L.; Liu, R.; Mirzadarani, R.; Luo, T.; Lyu, D.; Ghaffarian Niasar, M.; Qin, Z.

DOI
10.1109/OJPEL.2024.3389211

Publication date
2024

Document Version
Final published version

Published in
IEEE Open Journal of Power Electronics

IEEE Open Journal of
Power Electronics

Received 15 February 2024; revised 4 April 2024; accepted 11 April 2024. Date of publication 16 April 2024;
date of current version 2 May 2024. The review of this article was arranged by Associate Editor Telles Brunelli Lazzarin.

Digital Object Identifier 10.1109/OJPEL.2024.3389211

A Data-Driven Model for Power Loss Estimation of Magnetic Materials Based on Multi-Objective Optimization and Transfer Learning

Z. LI (Graduate Student Member, IEEE), L. WANG (Graduate Student Member, IEEE), R. LIU (Student Member, IEEE), R. MIRZADARANI (Graduate Student Member, IEEE), T. LUO (Graduate Student Member, IEEE), D. LYU (Student Member, IEEE), M. GHAFFARIAN NIASAR (Member, IEEE), AND Z. QIN (Senior Member, IEEE)

Department of Electrical Sustainable Energy, Delft University of Technology, 2628CD Delft, The Netherlands

CORRESPONDING AUTHOR: ZIAN QIN (e-mail: z.qin-2@tudelft.nl)

ABSTRACT Traditional methods such as Steinmetz's equation (SE) and its improved variant (iGSE) have demonstrated limited precision in estimating power loss for magnetic materials. The introduction of Neural Network technology for assessing magnetic component power loss has significantly enhanced accuracy. Yet, an efficient method to incorporate detailed flux density information—which critically impacts accuracy—remains elusive. Our study introduces an innovative approach that merges Fast Fourier Transform (FFT) with a Feedforward Neural Network (FNN), aiming to overcome this challenge. To optimize the model further and strike a refined balance between complexity and accuracy, Multi-Objective Optimization (MOO) is employed to identify the ideal combination of hyperparameters, such as layer count, neuron number, activation functions, optimizers, and batch size. This optimized Neural Network outperforms traditionally intuitive models in both accuracy and size. Leveraging the optimized base model for known materials, transfer learning is applied to new materials with limited data, effectively addressing data scarcity. The proposed approach substantially enhances model training efficiency, achieves remarkable accuracy, and sets an example for Artificial Intelligence applications in loss and electrical characteristic predictions with challenges of model size, accuracy goals, and limited data.

INDEX TERMS Power magnetics, core loss, data-driven method, neural network.

I. INTRODUCTION

The empirical Steinmetz Equation (SE), introduced in 1890, has long served as the basis of the standard modeling techniques for loss estimation in power magnetics. Despite various improvements, such as iGSE and i2GSE, the accuracy of these curve-fitting techniques is still relatively low [1], [2]. The imprecise model will lead to a rough magnetic design. Thus, a large design margin is needed to achieve an acceptable loss performance of the magnetic components.

The 2023 MagNet Challenge [3], [4], [5], [6], led by Princeton University, focuses on improving the Steinmetz equation by leveraging an extensive dataset encompassing diverse materials, frequencies, waveform shapes, and temperatures. The objective is to develop innovative and refined data-driven methods to deepen the power electronics community's grasp of magnetic core properties, particularly in core loss. Extensive data for ten known materials and limited data for five unknown materials were released consecutively. The final evaluation criteria are based on the accuracy of loss prediction for the five unknown materials and the complexity of the model.

To accurately predict the loss of magnetic material, one possible approach is using neural networks. Neural networks have been utilized in the modeling of magnetic hysteresis

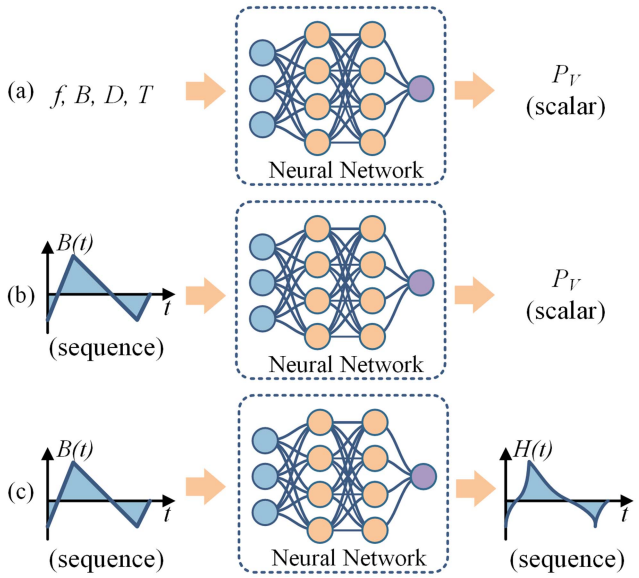

FIGURE 1. Three approaches for modelling magnetic material properties using neural networks: (a) scalar values to scalar values, (b) time series signals to scalar values, (c) time series signals to time series signals.

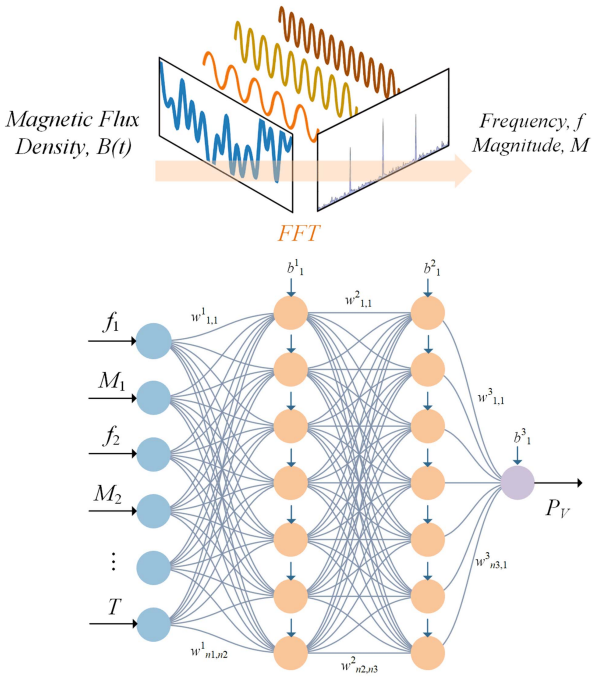

FIGURE 2. Proposed FNN with FFT structure. The Magnetic Flux Density is first decomposed into harmonics using FFT, then fed into the FNN.

loops, as referenced in [7], [8]. However, using hysteresis loops as an indirect predictor of power loss can introduce notable inaccuracies in the estimation of core losses due to minor phase discrepancies, as discussed in [5].

A more effective approach involves employing neural networks to directly predict the power loss and temperature of magnetic components [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. Reference [9] introduced Neural Network-aided Loss Maps (NNALMs) for both inductors and transformers, taking load conditions as inputs to predict both winding and core losses with an average error of less than 10%. Additionally, a Knowledge-Aware Artificial Neural Network (KANN) was developed in [11], [12], incorporating the output of the SE function or iGSE function as one of the inputs to a Feedforward Neural Network (FNN), thereby achieving high accuracy even with limited training datasets. Due to the minimal computational demands of trained FNNs, Reference [14] proposed utilizing two FNNs to predict real-time maximum temperatures and loss distributions in medium frequency transformers. Reference [10] employed a Convolutional Neural Network (CNN) to forecast inductance and core loss. One area for further investigation in these studies is the inclusion of detailed flux density waveform information, which has the potential to significantly enhance the accuracy of power loss predictions.

Based on extensive open-source magnetic materials data provided by MagNet [20], the current state-of-the-art methodologies in this field are those presented in [5], which include various neural network structures such as the Feedforward Neural Network (FNN), Long Short-Term Memory (LSTM), and the Encoder-Decoder structure, as depicted in Fig. 1(a), (b), and (c) , respectively. The FNN uses a four-layer FNN, taking peak flux density B, duty ratio D, and fundamental frequency f to predict core loss density P_V. The disadvantage of this method is that the B waveform shape information cannot be included by only using the peak flux density value. One possible approach is to train different FNNs for different B waveforms. However, there are an unlimited number of B waveform shapes. Thus, it is impractical to first categorize and then design separate FNNs for each case.

To effectively capture the waveform shape information of the flux density, other advanced neural networks can be applied, e.g., LSTM and encoder-decoder structures. LSTM and encoder-decoder architectures offer enhanced temporal data processing and sequence-to-sequence prediction capabilities [21], [22], outperforming traditional FNNs in handling long-term dependencies and complex patterns in time-series data. However, these models are more complex and require longer training time, which can be a critical drawback in practical applications where large volumes of data need to be processed with limited computational resources. Section III will analyze these advanced methods in detail.

To effectively encapsulate the information of the B waveform while reducing model complexity, employing FFT (Fast Fourier Transform) in conjunction with FNN presents a more efficient approach. This paper employs the FFT to convert the waveform data into its harmonic components. These harmonics are subsequently utilized as inputs for the FNN. This approach limits the number of parameters and decreases the complexity; thus, the training time is considerably shorter compared to other more complex methods. To the best of the authors' knowledge, this is the first research to integrate FFT with FNN for estimating magnetic power loss.

In the AI modeling process, a common dilemma is balancing model size with accuracy, where a model with more parameters typically offers higher accuracy but demands greater computational resources and larger dataset. The trade-off among various alternatives for generating the FNN dataset is analyzed in [13]. References [15], [16] are concentrated on optimizing the number of layers and neurons in neural networks. Reference [17] demonstrated that the GELU activation function and Huber loss function outperform the RELU activation function and MSE loss function. Despite these advancements, there remains an evident gap in the systematic optimization of neural network hyperparameters for magnetic component power loss estimation applications.

A viable solution to this challenge is implementing Multi-Objective Optimization (MOO) [23], where accuracy and the number of parameters are set as objectives. This approach facilitates the identification of optimal model hyperparameters from the generated Pareto Front, effectively resolving the trade-off between complexity and performance. In this study, Optuna is employed for MOO, which is an efficient hyperparameter optimization framework renowned for its simplicity and flexibility in various optimization tasks [24].

Although the methods mentioned above are sufficient for materials with extensive data, in real-world applications, engineers may not be able to measure a large amount of data when dealing with new materials. Yet, there is often a need to predict the performance of new materials quickly and accurately, which is exactly the goal of the Magnet competition. In such situations, transfer learning can play a significant role [25], [26]. Transfer learning allows the application of knowledge acquired from extensive datasets of known materials to predict the characteristics of new materials, even with limited data. This approach is particularly effective in reducing the need for extensive data collection for new materials, thereby accelerating the prediction process while maintaining accuracy.

In summary, the main contributions of this paper include the following:

1) We propose an FFT+FNN neural network structure for magnetic loss estimation, which can effectively capture key information of the B waveforms influencing the core loss. Compared to the advanced LSTM and encoder-decoder structure, the proposed structure has the simplest structure, comparable or better accuracy, and the shortest training time.
2) We demonstrate the necessity of conducting the MOO study and choosing hyperparameter combinations from the Pareto Front, which helps balance model complexity and prediction accuracy.
3) We demonstrate that applying transfer learning to new materials can mitigate the low accuracy issue of power loss prediction brought by data scarcity.

In this paper, we detail a neural network modelling approach that integrates MOO and transfer learning to accurately forecast the power loss in magnetic materials under conditions of limited data availability. Accordingly, the paper is organized as follows. Section II presents the employed FNN architecture and elaborates on the rationale behind the chosen input layer format and the number of harmonics. Section III conducts a comparative analysis with other sophisticated AI methodologies. Section IV details two approaches for predicting power loss in new materials: normal training and transfer learning, each integrating MOO. This section further compares the developed FNNs, aiming to identify the most effective model regarding parameter count and accuracy. Finally, Section V concludes the paper.

TABLE 1. Number of Data Points for the Ten Known Materials in the MagNet challenge: Each Data Point Includes the Complete B-H Loop, Frequency, Temperature, and Volumetric Loss information

Material	Sine	Triangular	Trapezoidal	total
TDK N27	479	3542	7375	11396
TDK N30	500	2691	5787	8978
TDK N49	334	2839	5429	8602
TDK N87	1426	13190	26000	40616
Ferroxcube 3C90	1601	13123	25989	40713
Ferroxcube 3C94	1776	12627	25665	40068
Ferroxcube 3F4	146	2314	4103	6564
Ferroxcube 3E6	503	2045	4448	6996
Fair-Rite 77	482	3528	7434	11444
Fair-Rite 78	472	3524	7384	11380

B waveforms are categorized into sine, triangular, and trapezoidal shapes.

TABLE 2. Number of Data Points of the Five New Materials of MagNet challenge

Material	Sine	Triangular	Trapezoidal	total
Material A	101	694	1637	2432
Material B	364	2253	4783	7400
Material C	215	1679	3463	5357
Material D	145	400	35	580
Material E	57	667	1289	2013

The identities of the new materials are undisclosed.

II. PROVIDED DATASET AND PROPOSED NEURAL NETWORK

A. MAGNET CHALLENGE PROVIDED DATASET

Table 1 describes the number of data points provided by the MagNet challenge. Each data point encompasses single-cycle B–H loop data with 1024 time steps per cycle. For each material, the provided data includes five CSV files, including B Field (N × 1024, in *T*), H Field (N × 1024, in *A/m*), Frequency (N × 1, in Hz), Temperature (N × 1, in °C), and Volumetric loss (N × 1, in W/m^3), where N refers to the number of data points. The B field data are presented in various waveforms — sinusoidal, triangular, and trapezoidal — to simulate different excitations in magnetic materials. Temperature values are distributed at 25, 50, 70, and 90 °C. The Frequency of the B field varies from 50 kHz to 800 kHz. The provided data have been open-sourced in GitHub [6].

For the final test, the competition committee has provided five new materials, as shown in Table 2. The identities of the new materials are undisclosed and designated as A, B, C, D, and E. The dataset has the complete information, including B Field, H Field, Frequency, Temperature, and Volumetric loss. As seen in the tables, the data points of the five new

TABLE 3. Comparison of Different FNN Input formats

Cases	Neurons in hidden layers	Total number of parameters	Avg. relative Error on test set
0	(32,64,32)	4929	1.17
1	(64,64,64)	10433	2.2
2	(32,64,32)	4641	1.69

Case 0 represents [f_1, M_1, f_2, M_2, …, f_n, M_n, T], with harmonics magnitudes and frequencies information. Case 1 represents [f_1, M_1, φ_1, f_2, M_2, φ_2,…, f_n, M_n, φ_n, T], with harmonics magnitudes, phases, and frequencies information. Case 2 represents [M_1, M_2, …, M_n, f_1, T], with harmonics magnitudes information and only the fundamental frequency.

materials are much fewer than the ten existing materials. This setup holds practical significance as there are countless types of magnetic materials in reality, and engineers often need to accurately predict the losses of these materials based on limited data obtained in a short period. The data scarcity scenario will be considered in the subsequent analysis.

In machine learning, datasets with complete information, encompassing parameters such as the B Field, H Field, Frequency, Temperature, and Volumetric loss, are typically divided into three sets: training, validation, and test. The training dataset is used to train the model, allowing it to learn and adapt to the data patterns. It usually comprises the largest data portion, often ranging from 50% to 80%. The validation dataset serves as a tool for tuning the model parameters and providing an unbiased evaluation of a model fit during the training phase. It typically constitutes about 10% to 25% of the data. Finally, the test dataset is used to assess the performance of the model after the training is complete. It provides an unbiased evaluation of the final model fit and is usually about 10% to 25% of the data. The exact proportions can vary based on the size and specifics of the dataset. A typical training process will be demonstrated in Section II-C.

B. PROPOSED FFT+FNN METHOD

The proposed Neural Network structure is shown in Fig. 2. Initially, the magnetic flux density B(t) is broken down into its harmonic components, characterized by frequencies f, magnitudes M, and phases φ. For training the FNN, we input the harmonic frequencies f and magnitudes M, along with temperature data T. The inputs of FNN are thus formatted as [f_1, M_1, f_2, M_2, … , f_n, M_n, T], as illustrated in Fig. 2. The rationale behind the selection of this input format is discussed further in Section II-D.

Apart from the input layer, as illustrated in Fig. 2, the FNN also has an output layer and several hidden layers. The output layer directly generates the predicted volumetric loss P_V. The interconnected parameters of the network, which form the core of the FNN's operational mechanics, are mathematically characterized as follows

$$z_i^j = \sigma\left(\sum_{k=1}^{n}\left(w_{k,i}^j \cdot z_k^{j-1} + b_k^j\right)\right) \tag{1}$$

where w represents the weight connecting each pair of hidden neurons, while b denotes the bias associated with each hidden neuron. The subscripts indicate the indices of hidden neurons, whereas the superscripts refer to the layer indices. Each hidden neuron employs a nonlinear activation function, $\sigma(x)$, enabling the network with the ability to learn nonlinear relationships. The term z computes the output of each hidden neuron, with its subscript and superscript following the same notation used for w and b. This architecture enables the FNN to effectively model complex relationships in magnetic materials, such as those between B, Frequency, and Temperature, ultimately allowing the output layer to predict power loss precisely.

In this study, as the analyzed B(t) and H(t) waveforms are in a steady state, FFT serves as an efficient method for decomposing these signals. Should non-stationary signals be analyzed in other studies, alternative sophisticated analytical tools, such as Wavelet Transform, Empirical Mode Decomposition (EMD), and Walsh-Hadamard Transform, could be employed to address their complexity.

C. TRAINING PROCESS OF THE PROPOSED METHOD

This section presents an example of the training process and explains the hyperparameters of the model. Suppose the FFT+FNN model aims to predict the N27 material loss, and N27 data is split into 60%, 20%, and 20% for train, validation, and test datasets, respectively.

In this example, we select [f_1, M_1, f_2, M_2, … , f_{10}, M_{10}, T] with ten harmonics information of B as the input format of the FNN. The number of neurons in the hidden layers is determined based on the experience, which is (32, 64, 32), and the activation function is ReLU. The batch size is set to 128, which refers to the number of training samples processed before the internal parameters of the model are updated. This size balances the training efficiency and the computational resource requirements. The training optimizer, set as Adam, is an algorithm used to adjust the weights of the neural network to minimize the loss function, thereby improving the performance of the model [27]. The model is trained in PyTorch, an open-source machine learning library widely used for deep learning and neural network modeling applications, known for its flexibility and ease of use [28]. 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 [5]. In Section IV-A, a hyperparameter optimization will be carried out.

Fig. 3 shows an example of the training process. In the model training process, training and validation losses are computed to monitor and enhance performance. Each epoch, defined as a complete pass through the training dataset, involves adjusting model weights to minimize training loss. During each epoch, the training dataset is shuffled to present the data in a randomized order, ensuring that the model does not learn any potential sequence biases and improving its ability to generalize from the training data. Simultaneously, validation loss is evaluated to assess the generalization capabilities of the model. A pattern where validation loss initially decreases and then increases suggests overfitting, indicating that the model excessively captures noise and fluctuations in the training data, impairing its generalization ability.

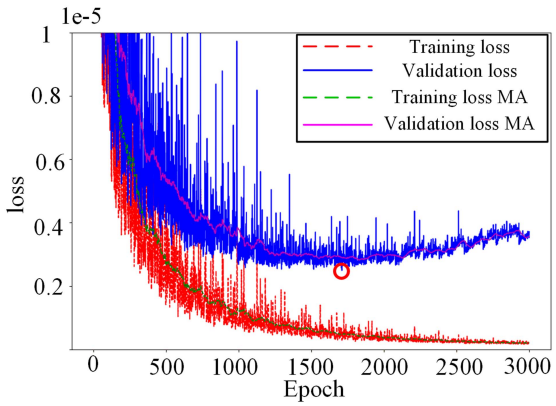

FIGURE 3. Example of training progress and early stopping method, with MA indicating Moving Average. The red circle marks the epoch with the lowest validation loss, and the corresponding model is saved during training.

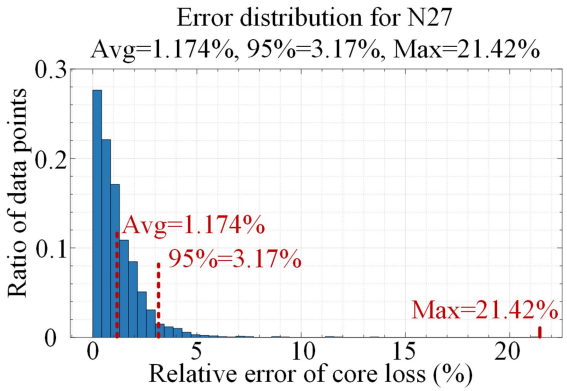

FIGURE 4. Distribution histogram of the N27 core loss relative errors, including average error, 95% error, and maximum error.

To prevent overfitting, an early stopping technique is applied. This approach terminates the training when no improvement in validation loss is observed over a predefined number of epochs, set here as 2000. If the validation loss fails to decrease over these epochs, the model may no longer be improving and could be overfitting. For instance, as illustrated in Fig. 3, where the point of lowest validation loss is circled in red, the model corresponding to this epoch is saved as the final model. This ensures that the most efficient version of the model, with optimal generalization capability, is retained for practical application. The early stopping method is used in the following model training process.

Upon finalization of the model, it was applied to the test dataset to evaluate performance. This process entailed comparing the predicted and measured core loss to ascertain model accuracy. Fig. 4 presents an example of the error distribution. The error analysis in this study utilizes three key metrics for assessing the accuracy of core loss predictions. Average Error is calculated as the mean of the relative differences between predicted and actual values across all data points, providing a general indicator of prediction accuracy. The 95% Error, defined as the value below which 95 percent of observed errors fall, offers insight into the typical error range for the majority of predictions. Maximum Error, representing the highest error in the dataset, indicates the largest deviation between predictions and actual measurements, illustrating the worst-case scenario in model performance. In subsequent sections, these three metrics are presented directly without repeating error distribution diagrams.

TABLE 4. Comparison of Different Numbers of Harmonics As Input

No. harmonics	Neurons in hidden layers	Total number of parameters	Avg. Error on test set	Max. Error on test set
3	(16,32,16)	1217	1.38	7
5	(16,32,16)	1281	1.44	12
10	(32,64,32)	4929	1.17	21

D. DESIGN OF INPUT LAYER

Different input formats of FNN can be applied. Apart from the one $[f_1, M_1, f_2, M_2, \ldots, f_n, M_n, T]$ as presented in Fig. 2, other input formats also have been considered, for example $[f_1, M_1, \varphi_1, f_2, M_2, \varphi_2, \ldots, f_n, M_n, \varphi_n, T]$, where φ_k represents the phase information of the kth harmonic; and $[M_1, M_2, \ldots, M_n, f_1, T]$, where in this case only the fundamental harmonic Frequency is included. These three cases are marked as case 0, case 1, and case 2, respectively.

A comparison between these three cases has been carried out based on N27 material. The number of neurons in the hidden layers is determined based on the experience. The data split method and other hyper-parameters are the same as in Section II-C.

Table 3 shows the comparison results of the three cases, and there are two conclusions. First, adding harmonics phase information does not help increase loss prediction accuracy. Second, when the model is trained using only a single fundamental frequency as input, its performance is inferior compared to when it is trained with both the frequencies and magnitudes of the harmonic components paired together. This observation holds under the scenario where the network parameters are initially selected based on empirical experience and are not subject to further optimization. Therefore, the inputs for the FNN are chosen to be in the format of $[f_1, M_1, f_2, M_2, \ldots, f_n, M_n, T]$.

A further question is how many numbers of harmonics are suitable to consider for model training and prediction. In the provided data of ten materials, there are three different types of B waveforms, i.e., sinusoidal, triangular, and trapezoidal. For sinusoidal waveforms, only a fundamental component exists. As for triangular and trapezoidal waveforms, two examples are shown in Fig. 5. A general impression is that the magnitude is negligible over the third harmonic.

Three different FNN models are built for three, five, and ten harmonic inputs, respectively. A comparison between these three FNNs is shown in Table 4. The model training and testing settings are the same as the tests in Table 3.

The difference in neuron settings in the middle layer is due to the different number of inputs. As can be seen from the comparison results, though with the total number of parameters differences, the results of using three harmonics, five

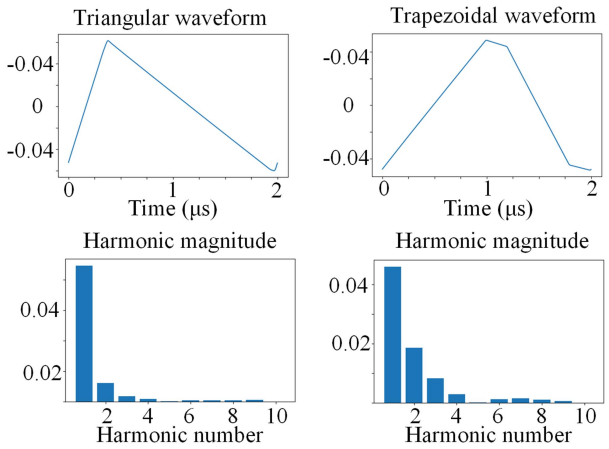

FIGURE 5. B waveform examples and harmonic magnitudes. The triangular and trapezoidal B waveforms are randomly selected from N27 dataset.

harmonics, and ten harmonics are similar. Furthermore, observing Fig. 5 suggests that reducing the number of considered harmonics is justified, as the magnitudes of higher harmonics appear negligible. For the simplicity of the model structure, which is essential for training and future optimization, three harmonics are chosen for the input. Consequently, the input format for the FNN in this study has been established as [f_1, M_1, f_2, M_2, f_3, M_3, T].

III. COMPARISON WITH THE STATE OF ART

With the given magnetic flux density $B(t)$, magnetic field strength $H(t)$, frequency f, temperature T, and power loss P information, different AI models can be applied to predict the power loss. Apart from the scalar-to-scalar model presented in this paper, [5] also offered two other models: sequence-to-scalar model using LSTM plus FNN and sequence-to-sequence model using encoder-decoder structure with LSTM. This section compares the methods introduced in this paper with those previously described in [5]. A detailed description of these two methods is provided in [5]. The following content briefly outlines their key concepts.

A. SEQUENCE-TO-SCALAR: LSTM + FNN MODEL

To effectively address the challenge posed by the diverse shapes of B waveforms, an alternative approach is to employ a sequence-based neural network structure.

The LSTM network is a key architecture in regression problems with sequential input. Its unique capability lies in memorizing information across data sequences, making it highly suitable for analyzing time series data. As shown in Fig. 6(a), the input gate i_t, forget gate f_t, and output gate o_t are essential to the functionality of the LSTM, which regulate information flow and enable selective memory storage over specific intervals. Detailed math descriptions can be referred to [5].

Based on LSTM cell structure, a sequence-to-scalar LSTM-based model is developed, using time sequences of flux density B(t) as input (1024-time steps per cycle) and predicting volumetric power loss as output. Fig. 6(b) displays the model structure, where the LSTM is taken as the input layer to process the entire flux density waveform. LSTM outputs are then concatenated with temperature and frequency information to feed into an FNN to perform core loss regression. This specific design features an LSTM with 18 cell states and 18 hidden states, and the FNN consists of three hidden layers (each with 12 neurons), resulting in an output representing the volumetric magnetic core loss. Overall, the model comprises 2089 parameters. To assess the performance of this LSTM-based core loss model, the network depicted in Fig. 6(b) has been synthesized and trained using PyTorch. N27 data are randomly split into 60%, 20%, and 20% for train, validation, and test datasets, respectively. The training results are shown in Table 5. In addition, the training time for each model is shown to compare the required computation resources. The neural network trainings are conducted in a PC equipped with an AMD Ryzen 7 5800H processor and an NVIDIA GeForce RTX 3060 graphics card. The results will be analysed in Section III-C.

TABLE 5. Comparison Between the FNN, LSTM Plus FNN, and Encoder-Decoder Based on N27 data

Cases		Total number of parameters	Training time	Avg. relative error on the test set
FFT+FNN	Original	1217	~1 hour	1.38
	Optimized	1419	~1 hour	0.75
LSTM+FNN		2089	~20 hours	1.37
Encoder-projector-decoder		28097	~20 hours	3.34

The optimized FFT+FNN represents the MOO results shown in section IV.

B. SEQUENCE-TO-SEQUENCE: ENCODER-PROJECTOR-DECODER MODEL

Beyond the direct prediction of power loss in magnetic components using B(t), Frequency, and Temperature data, an alternative approach involves initially forecasting the time sequence of H(t). Subsequently, both B(t) and H(t) can be utilized to compute the power loss:

$$P_V = \frac{1}{T}\int_{B(0)}^{B(T)} H(t)dB(t) \tag{2}$$

The encoder-decoder network framework [29], a leading-edge architecture, has gained considerable interest in recent years for its effectiveness in sequence-to-sequence regression challenges. The encoder-decoder network framework can also be applied to this magnetic B–H hysteresis loop prediction target.

To include Temperature and Frequency information, [5] proposed an encoder-projector-decoder structure as shown in Fig. 7. The encoder comprises a single-layer LSTM network with 32 states, employing input, forget, and output gates of LSTM to process the sequence B(t). This processing captures sequence characteristics in hidden and cell states. Additional inputs, i.e., Temperature T and Frequency f, are combined with these states to feed into a projector consisting of two

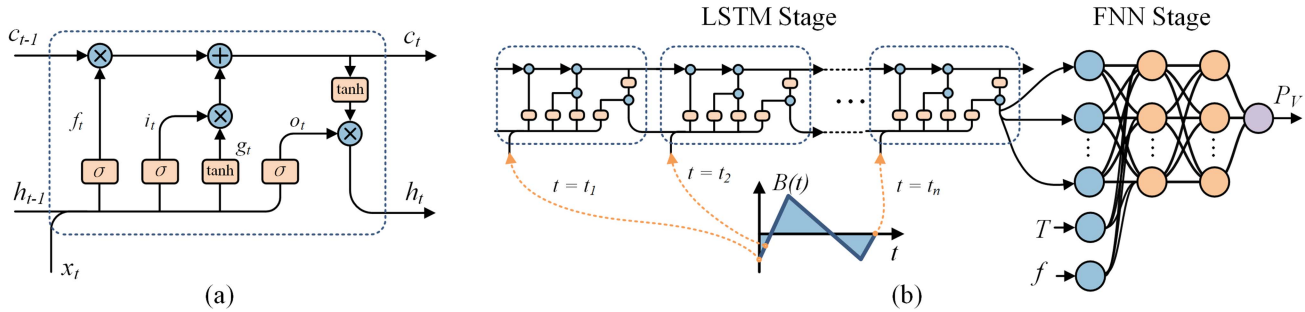

FIGURE 6. Sequence-to-scalar LSTM network structure. (a) Basic structure of the LSTM cell (b) Structure of the complete LSTM-FNN. The B waveform is initially input into the LSTM network, and then the output of LSTM is concatenated with temperature *T* and frequency *f* to feed into the FNN.

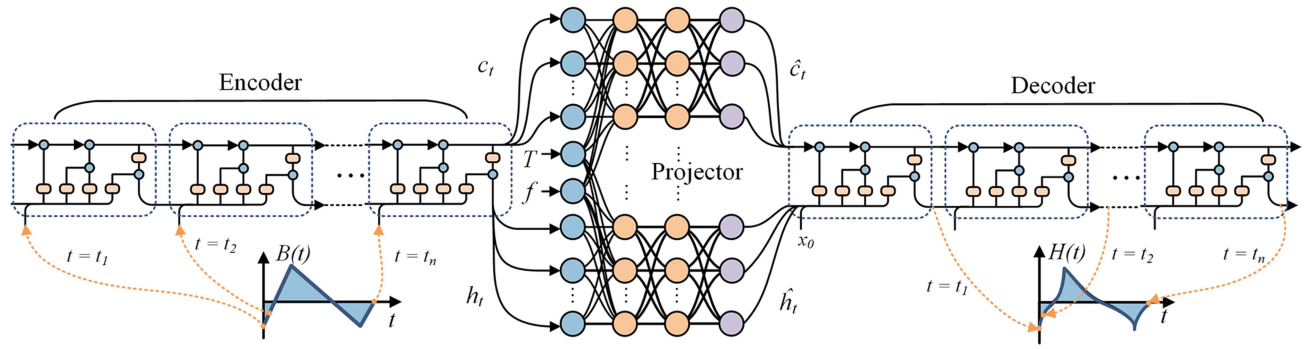

FIGURE 7. Encoder-projector-decoder network structure. Both the encoder and decoder comprise LSTM networks. The projector integrates the encoder's cell and hidden states with temperature and frequency as FNN inputs, generating the cell and hidden states for the decoder.

three-layer FNNs with 64 neurons in each layer. The projector adjusts state values in response to these additional inputs, resulting in updated states. These updated states are then utilized to initialize the decoder, which has a structure identical to the encoder's. Utilizing the updated hidden and cell states, the decoder produces the output sequence H(t).

The network employs mean-squared error as the loss function to measure the discrepancy between the predicted and target H(t). This function is used to adjust the weights and biases within the network. To manage the higher computational demands, the B(t) and H(t) are down-sampled to 102 time steps per cycle. Other settings and training environments are the same as LSTM+FNN model. After the model is finalized, the power loss is calculated according to (2) with the input B(t) and the predicted H(t). The results are shown in Table 5.

C. COMPARISON RESULTS AND ANALYSIS

Table 5 shows the training results of the LSTM+FNN, the encoder-projector-decoder structure, and the proposed FFT+FNN method. Additionally, the optimized FFT+FNN detailed in Section IV-A is also shown for benchmarking.

The encoder-projector-decoder is the most complex model, but the power loss estimation is less accurate than the other two. The reason is that the power loss is not directly generated as the output of the neural network. On the contrary, the power loss is indirectly calculated by integrating B(t) and H(t). A slight phase discrepancy in the predicted H(t) sequence might not cause a significant relative error in sequence-to-sequence regression, yet it could have a substantial effect on the accuracy of core loss prediction [5].

In contrast to the encoder-projector-decoder structure, LSTM+FNN and the original FFT+FNN, which directly correlate *B*, *f*, and *T* data with power loss, show equally high performance. This leads to the first insight: complexity alone does not guarantee higher accuracy in power loss prediction; direct prediction of power loss as the network's output can achieve better results, because in this case any intermediate errors, e.g., phase error of H(t), are avoided.

Although the LSTM+FNN effectively captures detailed information from B waveforms, its performance does not surpass that of FFT+FNN. This observation leads to the second insight: within the scope of the most general waveforms for power-electronic magnetic components, which is covered by the existing dataset, the results from FFT+FNN show that the harmonic magnitudes and frequencies can represent the dominant characteristics influencing the core loss. Besides, this sequence-to-scalar model presents two disadvantages compared to the proposed scalar-to-scalar model. First, it is more complex and requires longer training time, which is unsuitable for further optimization. Second, in the case of transfer learning using the data measured by other research groups, adjustments like down-sampling/up-sampling or modifying the model structure might be needed for transfer learning, especially when handling data sequences of different lengths.

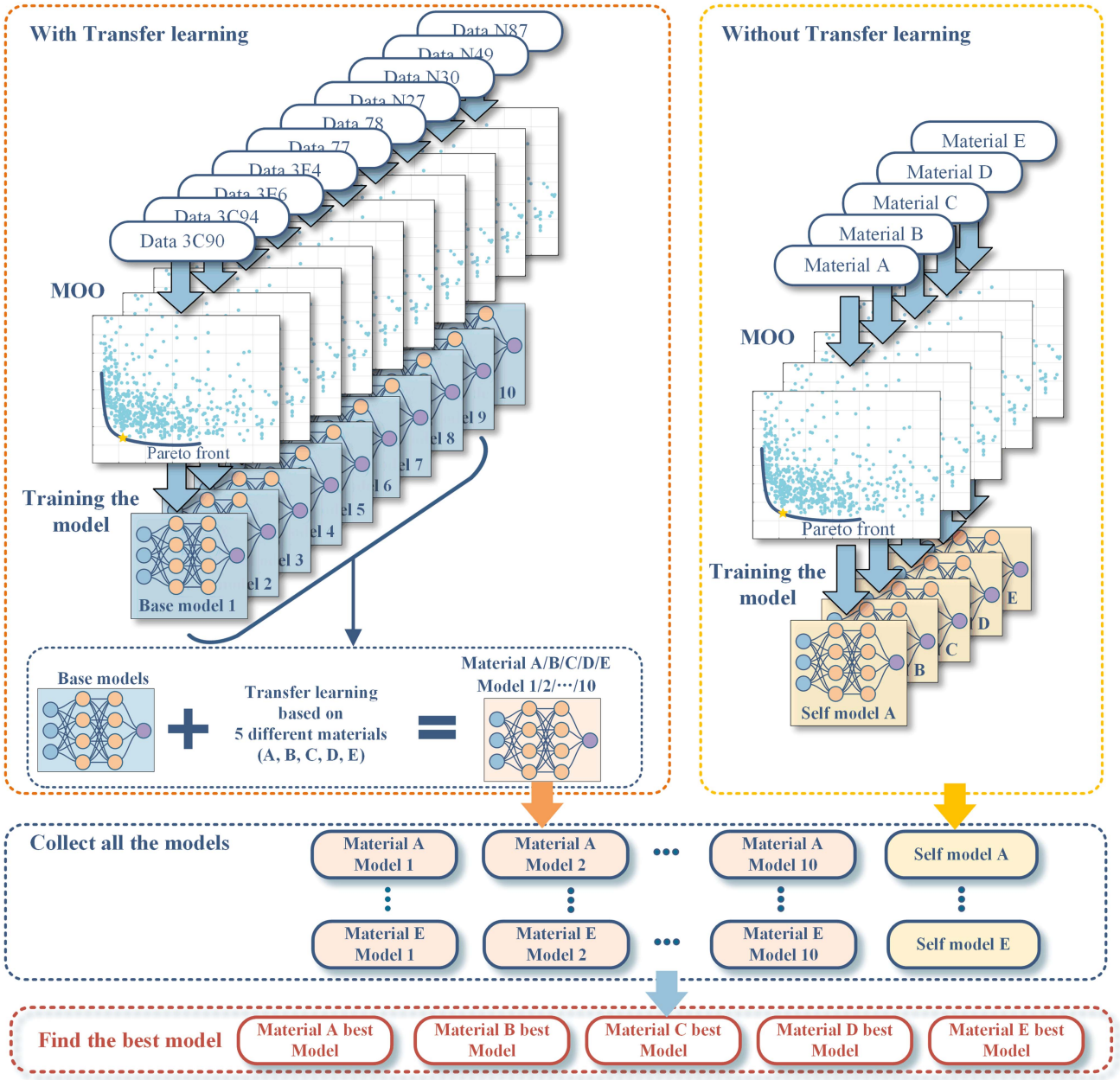

FIGURE 8. Overall model training and comparison process. Two training methods are compared. The first method optimizes and trains on ten existing materials' data before applying transfer learning to the five new materials' data, while the second directly optimizes and trains the model using data from five new materials.

In comparison, the proposed FNN model is the simplest and quickest for training. With hyperparameters optimization of FNN, the model's performance can significantly increase. Given its simplicity and high accuracy, the proposed FNN has been demonstrated to be suitable for the following development and optimization.

IV. MULTI-OBJECTIVE OPTIMIZATION, TRANSFER LEARNING AND COMPARISON

The overall model construction and selection methods for the five new materials are presented in Fig. 8. In the normal training approach without transfer learning, for the five new materials, the FNN model for each material is first optimized with the MOO tool Optuna, as detailed in the next section. New materials training data as shown in Table 2 are split into 70%, 20%, and 10% for training, validation, and test purposes, respectively. The proportion of the training dataset is slightly higher, at 70%, due to the limited quantity of data. This approach aims to maximize the use of available data for training purposes. Then, the most effective hyperparameter combination with a low number of parameters and low validation loss is selected. The model performance on the test dataset is recorded for later comparison.

Although with MOO, the test performance is relatively satisfactory with the normal training process, due to the low amount of new material data, a better approach is to use transfer learning. This method holds significant importance in real-world scenarios, such as for designers who may lack adequate data on new materials required for normal training models. This part will be analyzed in IV.B. The second approach with transfer learning, as shown in Fig. 8, also starts with a MOO for the ten existing materials. This will increase the accuracy of the following step, which uses transfer learning.

The data on the ten existing materials are split into 70%, 15%, and 15% for training, validation, and test purposes, respectively. This distribution strategy differs slightly from the previous one, as in this case, the ten existing materials data are more abundant, providing a more comprehensive basis for analysis. Once the optimal model for each of the ten known materials is determined, the five new materials will be subjected to transfer learning across the ten models through fine-tuning. As a result, 50 new transfer-learned models for the five new materials will be acquired, and the test data results will be recorded. The transfer-learned and normal training results described previously will be compared to select the final best model.

A. MULTI-OBJECTIVE OPTIMIZATION

In assessing the neural network model, prediction accuracy and the number of parameters are crucial metrics. While accuracy reflects the model's effectiveness, the parameter count directly indicates its complexity and significantly impacts the time required for training and fine-tuning. To optimize the model considering both values, a MOO is needed.

MOO aims to find the optimal set of parameters, simultaneously optimizing multiple objectives, leading to trade-offs and compromises. Optuna, a popular Python library, provides a comprehensive framework for conducting MOO. Utilizing Optuna for MOO, we focus on two primary objectives: reducing the validation loss to enhance model accuracy and minimizing the number of parameters. Different hyperparameters tuning methods have been analyzed in [30], [31], and NSGA-II (Non-dominated Sorting Genetic Algorithm II) is favored in this study as the optimization engine for its effective balance in ranking and diversity preservation in MOO contexts [32]. The search space is defined as follows: the number of middle layers in the FNN structure varies between 2 and 5, each containing 8 to 64 neurons. The activation functions considered include ReLU, LeakyReLU, Tanh, Sigmoid, ELU, SELU, and SiLU. These activation functions, each with distinct characteristics, play a crucial role in introducing nonlinearity into the neural network, enabling it to learn complex patterns in the data [33], [34]. Possible optimizers are Adam, SGD, RMSprop, and AdamW [35]. The batch size options are set at 64, 128, or 256.

For each of the five new materials, MOO involves 500 trials, each comprising 3000 epochs. In contrast, for the ten existing materials, which have much larger datasets, MOO consists of 200 trials with 1000 epochs each. An example MOO result is presented in Fig. 9, where yellow and blue stars indicate the possible hyperparameter combinations on the Pareto Front. In addition, the black star represents the previous hyperparameter combinations selected based on experience, as shown in Section II-D. The following compares these combinations and describes how to choose the most effective hyperparameter combinations on the Pareto front.

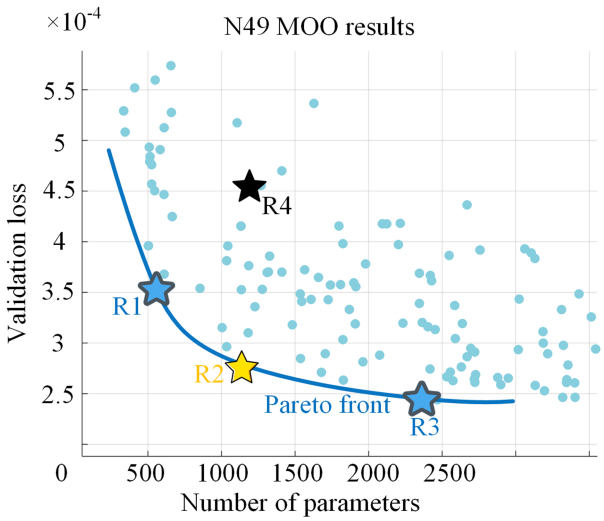

FIGURE 9. MOO Result Example. The depicted Pareto front represents the optimization boundary, balancing the trade-off between the number of parameters and validation loss, illustrating the most efficient solutions where neither objective can be improved without worsening the other.

TABLE 6. Four Different Points on n49 Moo Results Shown in Fig. 9

case	Batch size	Hidden layers neurons	No. params.	Act. Func.	N49 avg.	Material E TF avg.
R1	64	(21, 17)	560	Tanh	2.01	2.75
R2	128	(29, 29)	1132	Tanh	1.9	2.49
R3	128	(38, 19, 33, 19)	2371	SiLU	1.46	2.35
R4	128	(16,32,16)	1217	ReLU	2.2	3

Table 6 summarizes these combinations training results, and Adam is the optimizer for four cases. It is important to note that each trial during the MOO was limited to 1000 epochs. For a fair comparison of the accuracy of the four combinations in Table 6, the early stopping method, detailed in Section II-C, is employed. As a result, when training these combinations, the epochs go beyond 1000. Table 6 initially suggests that the R4 combination, chosen based on prior experience, does not perform as well as those on the Pareto Front. Despite R4 having more parameters than R1 and R2, it exhibits a higher average error on the test dataset than R1 and R2. This also leads to a lower transfer learning accuracy, a point further elaborated in Section IV-B. This highlights the necessity of conducting the MOO study and choosing hyperparameter combinations from the Pareto Front. For R1,

TABLE 7. All Materials Optimized Hyper Parameters

Material	Batch size	Hidden layers neurons	No. params.	Activation function
N27	64	(30,18,31)	1419	SiLU
N30	64	(33,38,16)	2197	SiLU
N49	128	(29,29)	1132	Tanh
3F4	64	(17,30,45)	2117	Tanh
3E6	64	(18,34,14)	1295	Tanh
3C90	256	(47,35,14,9)	2705	Tanh
3C94	256	(59,27,32)	3021	Tanh
N87	64	(35,26,11)	1525	SiLU
77	128	(44,24,15,38)	2454	Tanh
78	64	(26,17,55,11)	2285	SiLU
A	64	(21,27,31)	1662	Tanh
B	64	(19,19)	552	Tanh
C	256	(36,17)	935	Tanh
D	64	(62,12,13,29)	1857	SiLU
E	64	(42,11,57)	1551	SiLU

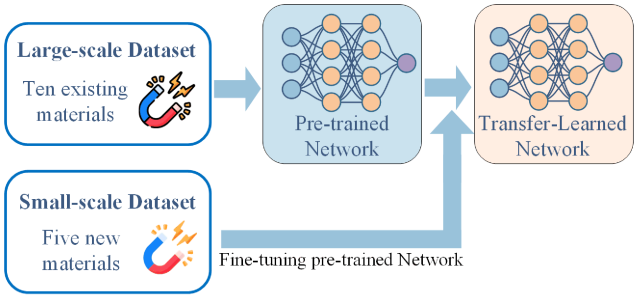

FIGURE 10. Transfer learning process for the five new materials. Once the network is pre-trained, data from the five new materials is used to fine-tune the network parameters of the pre-trained model.

R2, and R3 on the Pareto Front, the test dataset of N49 demonstrates that accuracy increases with the number of parameters. However, to strike a balance between the number of parameters and accuracy, R2 is selected as the most effective choice for further transfer learning. A similar analysis approach is applied to the other materials.

A summary of all MOO results is shown in Table 7. The optimizer for all cases is Adam. The optimization results reveal variability in the number of neurons of hidden layers, indicating that optimal network structures differ for each material. The consistent use of the Adam optimizer across models emphasizes its effectiveness in handling the complexities of magnetic material properties. Moreover, the frequent selection of Tanh and SiLU as activation functions reflects their capability in accurately modeling the nonlinear behavior inherent in magnetic materials loss prediction.

B. TRANSFER LEARNING

The transfer learning process is shown in Fig. 10. This method typically begins by pre-training a neural network on a large-scale dataset. Following this, a smaller dataset is employed to fine-tune the network's parameters. This fine-tuning step is crucial, as it adjusts the model to align more closely with the specific characteristics of the smaller dataset, enhancing its performance and accuracy in tasks related to this dataset. This approach leverages the generic features learned from the large dataset, applying them effectively to the more specialized requirements of the smaller dataset.

TABLE 8. Comparison of Transfer Learning Accuracy With Different Base Materials Combinations

Base model training materials	Total number of datapoints	Transfer learning material A accuracy	
		Avg. Error	Max. Error
3C90	40713	2.16	16.04
3C90+N27+77	63553	2.86	19.2

Transfer learning's efficacy hinges on the relatedness of the source and target domains. A substantial discrepancy between these domains can render the transferred knowledge ineffective or even detrimental. In our study, however, the large-scale dataset features ten existing materials, while the small-scale dataset includes five new materials. Both datasets focus on magnetic materials, ensuring a significant correlation in domain-specific characteristics and thus, a high level of applicability for the transferred knowledge. This core similarity greatly reduces the chance of the performance drop-off that usually happens when there's a big gap between the two domains.

Further, to establish the pre-trained network, a critical decision is whether to independently create ten individual pre-trained models using each of the ten existing materials or construct a single pre-trained model by combining datasets from multiple materials.

As shown in Table 8, we have experimented with training the base model using a combined dataset of several materials (such as 3C90, N27, and 77). However, the outcomes of transfer learning with this approach did not surpass those achieved when training the base model exclusively with data from a single material (3C90). This observation leads to the speculation that new materials may exhibit closer properties to a specific existing material rather than a collective representation of several. When combining data from multiple materials, there is a potential for inter-data interference, which might hinder the model's ability to learn and predict the properties of a new material accurately. Based on this insight, we have decided to utilize a database of a single material as the starting point for transfer learning to ensure a more focused and effective model training process. After the MOO, ten base models trained on existing materials are ready for transfer learning. New material data will be directly input into the pre-trained base model to fine-tune the parameters of the network.

A test is conducted to testify the MOO results and the transfer learning effectiveness. Based on N49 trained models, as shown in Table 6 in Section IV-A, the new material E data are applied for fine-tuning. The transfer learning results are shown in the last column of Table 6, which yields two

TABLE 9. Transfer Learning Results Based on Moo Models of the Ten Existing materials

material (number of parameters)	Self-training result		Transfer learning results									
			Material A		Material B		Material C		Material D		Material E	
	Average error (%)	95% error (%)	Average error (%)	95% error (%)	Average error (%)	95% error (%)	Average error (%)	95% error (%)	Average error (%)	95% error (%)	Average error (%)	95% error (%)
N27(1419)	0.75	1.97	2	6.7	0.75	1.89	1.39	3.93	3.16	10	2.4	7.25
N30(2197)	0.41	1.1	2.08	7.85	0.7	1.65	1.1	3.24	3.7	10	2.28	7
N49(1132)	1.9	5.78	2.688	8.9	0.79	2.2	1.37	3.9	5.11	19	2.49	7
3F4(2117)	1.45	3.8	2.58	7.8	0.84	2.17	1.23	3.4	5.2	15.7	2.66	10.9
3E6(1295)	0.455	1.19	2.25	6.3	0.74	1.98	1.23	3.62	4.26	10.65	2.77	8
3C90(2705)	0.83	2.35	2.23	8.1	0.76	1.97	1.2	3.78	3.86	10.9	2.38	9.24
3C94(3021)	0.75	2	3.1	10.4	0.75	2.02	1.37	4.2	5.9	22.4	2.63	7.09
N87(1525)	0.74	2	2.54	7.67	0.73	1.9	1.19	4	4.6	14.9	2.74	10.7
77 (2454)	0.77	2.1	2.2	6.37	0.92	2.48	1.21	3.54	4.77	14	2.2	6.4
78(2285)	0.72	1.89	2.19	7.1	0.69	1.9	1.21	3.72	5.23	15.38	2.69	8.17

The best-performing model for each new material, after transfer learning, is highlighted in green.

TABLE 10. Comparison of the Results with and Without Transfer Learning for the Five New materials

New Material	Results without transfer learning			Results with transfer learning			
	Number of parameters	Average error (%)	95% error (%)	Number of parameters	Average error (%)	95% error (%)	Base model
A	1662	2.18	6.73	1419	2	6.7	N27
B	552	0.9	2.2	2197	0.7	1.65	N30
C	935	1.65	4.4	2197	1.1	3.24	N30
D	1857	6.3	21.8	1419	3.16	10	N27
E	1551	2.47	7	2454	2.2	6.4	77

The training methods with and without transfer learning are corresponding to Fig. 8.

key insights. The first insight is that in the absence of MOO and when hyperparameters are chosen solely based on human experience, transfer learning is less efficient than those derived from the Pareto Front of MOO. This inefficiency is marked by an increased number of network parameters with a concurrent decrease in accuracy. Second, it can be concluded that for the results on the Pareto Front, an increase in the number of parameters enhances the accuracy of transfer learning results. To strike an optimal balance between the number of parameters and accuracy, the knee point on the Pareto front is identified as the most effective combination of hyperparameters.

C. RESULTS WITH AND WITHOUT TRANSFER LEARNING COMPARISON

Based on the overall process shown in Fig. 8, the results of the transfer learning approach and the normal training approach for the five new materials are presented in Tables 9 and 10. Table 9 compares the transfer learning results for the five new materials, and the best results are highlighted in green. Table 10 presents test dataset accuracy for the MOO models for the five new materials to compare with the transfer learning results.

As seen in Table 10, from an accuracy perspective, transfer learning results significantly outperform those without transfer learning, mainly due to the limited dataset available for the new materials. Although the number of parameters in transfer learning approaches is slightly higher than in normal training methods, this is not a significant concern for applying transfer learning to new materials. The rapidity of the transfer learning process, as demonstrated in our tests, enables quick adaptation to the characteristics of new materials. Consequently, despite the larger parameter size, the speed of transfer learning makes it a more practical choice. Therefore, considering both the superior accuracy and the efficiency in learning new material properties, the transfer learning models are deemed the most effective for new materials.

To ensure unbiased and reliable prediction results, the highlighted transfer learning approach, as discussed in Table 10, was subjected to further validation using a 10-fold cross-validation method [36], [37]. This process involves dividing the dataset into 10 equal parts, or "folds", where each fold is used once as a validation while the remaining 9 folds are used for training. This cycle is repeated 10 times, with each fold serving as the validation set once. The accuracy metrics from each fold are then averaged to produce a composite measure of model performance. The summarized results in Table 11 demonstrate the robustness of the proposed method across new materials, highlighting its effectiveness even with datasets of varying sizes.

The results from Table 11 clearly indicate that, despite the varied number of data points across new materials, the model consistently exhibits low average and 95% errors. This validates the effectiveness of the proposed MOO and transfer learning method.

TABLE 11. Model Performance Metrics on New Materials with 10-fold Cross Validation

New Material	Total number of data points	Number of model parameters	Avg. Error (%)	95% Error (%)
A	2432	1419	2.0	6.75
B	7400	2197	0.69	1.81
C	5357	2197	1.17	3.39
D	580	1419	3.53	10.58
E	2013	2454	2.12	6.50

V. CONCLUSION

This paper comprehensively presents a neural network-based modelling approach for accurately predicting the power loss of magnetic materials with limited data, incorporating MOO and transfer learning techniques. The fundamental neural network employed in this study is an FNN. Through a detailed comparison of various input structures and the number of harmonics, the chosen inputs are the magnitudes and frequencies of the first, second, and third order harmonics, plus temperature. Compared to other advanced methods, such as LSTM or encoder-decoder models, the proposed method achieves satisfactory accuracy with lower complexity. The model's performance is further enhanced by MOO, targeting both a reduction in the number of parameters and an increase in accuracy. Finally, a comparative analysis between results with and without transfer learning, both integrated with MOO, is conducted for new materials with limited data. The transfer learning shows a better performance with considerably lower error and marginal increase in the number of parameters.

To enhance the practicality and accessibility of our developed model, especially for engineers without extensive AI expertise, we are currently developing an optimization tool for magnetic material power loss prediction. This tool aims to streamline the application of our model in real-world scenarios. Furthermore, our proposed methodology, combining MOO and transfer learning, not only establishes a benchmark for similar studies in power electronics but also serves as a versatile template for addressing a variety of prediction tasks. These include the forecasting of thermal and electrical characteristics in power electronics, with a focus on achieving high accuracy while managing model complexity and overcoming the challenges of limited data availability.

REFERENCES

[1] F. Kral, "Analysis and implementation of algorithms for calculation of iron losses for fractional horsepower electric motors," Master's thesis, Graz Univ. Technology, Graz, Austria, 2017.

[2] C. N. M. Smailes, R. Fox, J. Shek, M. Abusara, G. Theotokatos, and P. McKeever, "Evaluation of core loss calculation methods for highly nonsinusoidal inputs," in *Proc. IET Int. Conf. AC DC Power Transmiss.*, 2015, pp. 1–7.

[3] H. Li, D. Serrano, S. Wang, and M. Chen, "MagNet-AI: Neural network as datasheet for magnetics modeling and material recommendation," *IEEE Trans. Power Electron.*, vol. 38, no. 12, pp. 15854–15869, Dec. 2023, doi: 10.1109/TPEL.2023.3309233.

[4] D. Serrano et al., "Why MagNet: Quantifying the complexity of modeling power magnetic material characteristics," *IEEE Trans. Power Electron.*, vol. 38, no. 11, pp. 14292–14316, Nov. 2023, doi: 10.1109/TPEL.2023.3291084.

[5] H. Li et al., "How MagNet: Machine learning framework for modeling power magnetic material characteristics," *IEEE Trans. Power Electron.*, vol. 38, no. 12, pp. 15829–15853, Dec. 2023, doi: 10.1109/TPEL.2023.3309232.

[6] M. Chen, E. Deleu, and H. Li, "MagNet challenge 2023," *Github*, Accessed: Dec. 28, 2023. [Online]. Available: https://github.com/minjiechen/magnetchallenge?tab=readme-ov-file#magnet-challenge-2023

[7] 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, doi: https://doi.org/10.1016/j.jmmm.2021.167735.

[8] 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. [Online]. Available: https://www.mdpi.com/1996-1944/13/11/2561

[9] N. Rasekh, J. Wang, and X. Yuan, "Artificial neural network aided loss maps for inductors and transformers," *IEEE Open J. Power Electron.*, vol. 3, pp. 886–898, 2022, doi: 10.1109/OJPEL.2022.3223936.

[10] X. Liu et al., "Convolutional neural network (CNN) based planar inductor evaluation and optimization," in *Proc. IEEE Appl. Power Electron. Conf. Expo.*, 2022, pp. 1506–1511, doi: 10.1109/APEC43599.2022.9773675.

[11] J. Deng, W. Wang, P. Venugopal, J. Popovic, and G. Rietveld, "Knowledge-aware artificial neural network for loss modeling of planar magnetic components," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2022, pp. 1–6, doi: 10.1109/ECCE50734.2022.9947398.

[12] J. Deng, W. Wang, Z. Ning, P. Venugopal, J. Popovic, and G. Rietveld, "High-frequency core loss modeling based on knowledge-aware artificial neural network," *IEEE Trans. Power Electron.*, vol. 39, no. 2, pp. 1968–1973, Feb. 2024, doi: 10.1109/TPEL.2023.3332025.

[13] D. Santamargarita, G. Salinas, D. Molinero, E. J. Bueno, and M. Vasić, "Tradeoff between accuracy and computational time for magnetics thermal model based on artificial neural networks," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 11, no. 6, pp. 5658–5674, Dec. 2023, doi: 10.1109/JESTPE.2022.3203934.

[14] D. Santamargarita, D. Molinero, E. Bueno, M. Marrón, and M. Vasić, "On-line monitoring of maximum temperature and loss distribution of a medium frequency transformer using artificial neural networks," *IEEE Trans. Power Electron.*, vol. 38, no. 12, pp. 15818–15828, Dec. 2023, doi: 10.1109/TPEL.2023.3308613.

[15] X. Shen, H. Wouters, and W. Martinez, "deep neural network for magnetic core loss estimation using the MagNet experimental database," in *Proc. IEEE 24th Eur. Conf. Power Electron. Appl.*, 2022, pp. 1–8.

[16] E. A. Juarez-Balderas, J. Medina-Marin, J. C. Olivares-Galvan, N. Hernandez-Romero, J. C. Seck-Tuoh-Mora, and A. Rodriguez-Aguilar, "Hot-spot temperature forecasting of the instrument transformer using an artificial neural network," *IEEE Access*, vol. 8, pp. 164392–164406, 2020, doi: 10.1109/ACCESS.2020.3021673.

[17] X. Shen and W. Martinez, "Machine learning model for high-frequency magnetic loss predictions based on loss map by a measurement kit," in *Proc. IEEE 25th Eur. Conf. Power Electron. Appl.*, 2023, pp. 1–8, doi: 10.23919/EPE23ECCEEurope58414.2023.10264272.

[18] 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, doi: 10.1109/OJPEL.2020.3012777.

[19] Z. Xiao, Y. Jiang, T. Sun, Y. Wu, and Y. Tang, "Refining power converter loss evaluation: A transfer learning approach," *IEEE Trans. Power Electron.*, vol. 39, no. 4, pp. 4313–4324, Apr. 2024, doi: 10.1109/TPEL.2023.3349178.

[20] H. Li, D. Serrano, and H. Li, "MagNet AI for research, education and design," *princeton.edu*, Accessed: Dec. 30, 2023. [Online]. Available: https://magnet.princeton.edu

[21] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[22] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[23] S. Roth, A. Gepperth, and C. Igel, "Multi-objective neural network optimization for visual object detection," in *Multi-Objective Machine Learning*, Berlin, Germany: Springer, 2006, pp. 629–655.
[24] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 2623–2631.
[25] Z. Xu, C. Sun, T. Ji, J. H. Manton, and W. Shieh, "Feedforward and recurrent neural network-based transfer learning for nonlinear equalization in short-reach optical links," *J. Lightw. Technol.*, vol. 39, no. 2, pp. 475–480, Jan. 2021. [Online]. Available: https://opg.optica.org/jlt/abstract.cfm?URI=jlt-39-2-475
[26] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. 27th Int. Conf. Artif. Neural Networks–Artificial Neural Netw. Mach. Learn.*, 2018, pp. 270–279.
[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[28] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8024–8035.
[29] Y. Wu et al., "Google's neural machine translation system: Bridging the gap between human and machine translation," 2016, *arXiv:1609.08144*\.
[30] N. Shawki, R. R. Nunez, I. Obeid, and J. Picone, "On automating hyperparameter optimization for deep learning applications," in *Proc. IEEE Signal Process. Med. Biol. Symp.*, 2021, pp. 1–7, doi: 10.1109/SPMB52430.2021.9672266.
[31] P. Nair et al., "AI-driven digital twin model for reliable lithium-ion battery discharge capacity predictions," *Int. J. Intell. Syst.*, vol. 2024, 2024, Art. no. 8185044, doi: 10.1155/2024/8185044.
[32] Y. Li, Z. Xie, S. Yang, and Z. Ren, "A hybrid algorithm based on NSGA-II and MOPSO for multi-objective designs of electromagnetic devices," *IEEE Trans. Magn.*, vol. 59, no. 5, May 2023, Art. no. 7001804, doi: 10.1109/TMAG.2023.3250319.
[33] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022.
[34] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," 2018, *arXiv:1811.03378*.
[35] R. Castro, I. Pineda, and M. E. Morocho-Cayamcela, "Hyperparameter tuning over an attention model for image captioning," in *Proc. Conf. Inf. Commun. Technol. Ecuador*, 2021, pp. 172–183.
[36] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 569–575, Mar. 2010, doi: 10.1109/TPAMI.2009.187.
[37] V. Vakharia, V. K. Gupta, and P. K. Kankar, "A comparison of feature ranking techniques for fault diagnosis of ball bearing," *Soft Comput.*, vol. 20, pp. 1601–1619, 2016.