



An intelligence parameter classification approach for energy storage and natural convection and heat transfer of nano-encapsulated phase change material: Deep neural networks

Mohammad Ghalambaz¹ · Mohammad Edalatifar¹ · Sara Moradi Maryamnegari^{2,3} · Mikhail Sheremet¹

Received: 18 October 2022 / Accepted: 23 May 2023 / Published online: 3 July 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

A deep neural network is utilized to classify the parameters of a natural convection heat transfer of a nano-encapsulated phase change material suspension using the isotherm images for the first time. A natural convection flow and heat transfer simulation dataset were created and used as a training and validation tool. Then, a deep neural network, consisting of three parts, was used for the classification task. The first part was made of several conventional layers, and a rectified linear unit activation layer supported each layer. The second part was a preparation layer for reshaping from 2D images to 1D classification. The third layer was made of a classifier layer. The results showed that the impact of the Rayleigh number and volume concentrations of nanoparticles could be classified by 99.8 and 93.32% accuracy, respectively. However, the Stefan number was classified weakly. As a part of the current research, a transfer learning approach was used to improve accuracy. The learning transfer approach was quite effective and improved the accuracy of the Stefan number classification by 16.6%.

Keywords Deep learning · Natural convection heat transfer · Physical characteristics classification · Nano-encapsulated phase change suspension

1 Introduction

Novel engineered coolants such as nanofluids and hybrid nanofluids are promising working fluids for many industrial cooling systems. These enhanced coolants can be used for natural convection cooling applications. Natural convection cooling has advantages that make it popular for industrial applications. For example, the low cost for maintenance compared to the systems with moving parts and forced convection heat transfer. In addition, cooling systems with the natural convection heat transfer concept

benefit from safety and low noise. However, these systems suffer from low performance since the thermal conductivity of working fluids is low. Moreover, the cooling performance of natural convection systems depends on the ambient and system temperatures as well as the geometrical design of the cooling system. Thus, the thermal management of such systems could be a challenging task. Therefore, new approaches to monitoring and estimating the system characteristics are highly demanded.

Using nanofluids is an up-to-date method to tackle the low efficiency of systems with natural convection heat transfer. Nanofluids are also widely used in different thermal systems, such as porous media [1], magnetic fields [2], and solar energy systems [3]. Using Fe₃O₄ [4], Al₂O₃-water [5, 6], CuO-water [5], and oil-Cu-Al₂O₃ [7] nanofluids lead to enhancement in the thermal performance of different systems. The Nano-Encapsulated-Phase Change Material (NEPCM) suspensions are a state-of-the-art type of hybrid nanofluids in which the nanoparticles are made of phase change capsules. NEPCMs have been examined for improving the thermal efficiency of

✉ Mohammad Ghalambaz
m.ghalambaz@gmail.com

¹ Laboratory on Convective Heat and Mass Transfer, Tomsk State University, 634050 Tomsk, Russia

² Faculty of Mechanical Engineering, K. N. Toosi University of Technology, Tehran, Iran

³ Institute of Electrotechnology, Leibniz University of Hannover, Hannover, Germany

Photovoltaic systems [8, 9]. The NEPCM particles have two parts consisting of a core and a nano-shell. They are filled with Phase Change Materials (PCMs) and enclosed with a nano-shell. The NEPCM particles are suspended in the host fluid and can be exposed to temperature gradients. When the NEPCM particles flow in a hot zone, they absorb some of the heat in the form of latent heat, and then they move along with the liquid and release the absorbed latent heat to the cold regions [10]. The NEPCM suspensions are utilized in natural convection heat transfer flows such as [11, 12]. Using the NEPCMs nanoparticles in natural convection heat transfer is a complex phenomenon that makes the physics of this challenging phenomenon. Therefore, an advanced numerical technique requires to model the NEPCMs.

The nonlinear Partial Differential Equations (PDEs) for the momentum and energy are required to predict particles' flow and phase change in the computational domain. In numerical simulations, the domain and PDEs will be discretized; and then, the linearized equations will be solved for every computational cell. The algebraic equation systems predict the flow field by an iterative method. It is time-consuming and has a high computational cost since it starts with an initial condition or initial guess and repeats solving the equations until it reaches the final solution. This is while estimating the flow parameters requires solving inverse problems along with the governing equations, which computationally is much more costly.

Some authors tried to improve the accuracy of correlations for engineering applications using ranking distance analysis methods [13, 14], response surface methods [15, 16], or soft computing techniques [17, 18]. Artificial intelligence methods are alternative simulations that have recently been the center of attention for many researchers [19–21]. Deep Neural Networks (DNNs), as a part of artificial intelligence, thanks to many hidden layers, can receive images and process them to meaningful outcomes. Moreover, the potential of DDNs to generate images makes them unique. Until a few years ago, the NNs were unable to utilize many hidden layers because of the limitation of the training methods and parallel computational resources. However, Hastad and Goldmann [22] overcame this problem and provided a new approach to using many hidden layers. Therefore, nowadays, the DDNs with many hidden layers are an excellent method for a data-driven approach.

Some authors used NNs to estimate several parameters and variables to find a relationship between the design variables, e.g., flow and heat transfer parameters and output data (Nusselt number, skin fraction, or stored energy). In such investigations, the input and output data are numbers or vectors, but they are not images or filed variables. For example, Ermis et al. [23] used the NNs with one hidden

layer in thermal energy storage units. The authors used NNs and found a relationship between the design parameters and the amount of total thermal energy storage. The relationship could provide estimations with a 5.58% absolute relative mean error.

Some researchers utilized NNs to estimate the flow or temperature fields. In such studies, the input parameters are the material coordinates and the characteristics parameters (design variables), and the output is an image (such as the flow field or temperature distribution). For example, Azwadi et al. [24] predicted the flow and temperature fields in a lid-driven enclosure using an adaptive network-based fuzzy inference system. They used the Lattice Boltzmann method to produce the dataset for training the ANFIS Neural Network. Moreover, they chose material coordinates (x and y), Reynolds number, and Rayleigh number as input parameters for NN. Akbari et al. [25] used the Jaya optimization algorithm to predict the amount of heat transfer (Nusselt number) in a cavity. They defined geometrical specifications and the Rayleigh number as inputs to calculate the Nusselt number.

Selimefendigil et al. [26] used NNs to predict the natural convection in the porous cavity in the presence of magnetic field effects. They chose the material coordinates (x and y), the porosity of the medium, Darcy Number, Hartmann number, and Rayleigh number as input parameters, and velocity and temperature as output parameters. They also used Long Short-Term Memory (LSTM) for training the NN. Their results showed that the prediction of temperature distribution is better than the velocity field with this method. The temperature distribution in a cavity with a natural convection heat transfer was calculated by Zhou et al. [27]. They numerically calculated the velocity field and temperature distribution to build the dataset to train the NN.

As seen, the literature review mainly focused on using the machine learning approaches to find a relationship between the design variables (such as Rayleigh number and Reynolds number) and the design goal (heat transfer rate in the form of Nusselt number or flow/temperature fields). However, estimating the flow characteristics parameters through observations of temperature field or velocity distributions without prior knowledge or the physical governing equations has not been addressed. Such estimation is advantageous in thermal management systems. Thus, the current study aims to use the temperature field of natural convection heat transfer (θ) as an input observation and estimate the design (the flow and heat transfer characteristics) parameters through an intelligence classification approach for the first time. Here, a DNN is employed to perform a parameter classification of the natural convection heat transfer of NEPCMs suspensions and identify the flow and heat transfer characteristics for the first time.

2 Mathematical model and the classification approach

The aim of the current research is to use a DNN to identify classes of flow and heat transfer characteristics using temperature distribution images. Thus, a DNN is trained using a database of input temperature images and output classes of non-dimensional parameters. The current research is made of two major steps. Firstly, a dataset of images corresponding to non-dimensional flow and heat transfer characteristics is required. This database could be obtained using a mathematical model of free convection flow in a square cavity. The cavity was filled by NEPCM suspensions, and its top and bottom walls were well insulated. The right and left vertical walls were kept as hot and cold isothermal temperatures, respectively. Due to a temperature difference between the vertical walls, a natural convection flow and heat transfer could occur.

Moreover, the phase change nanoparticles also contribute to the convection heat transfer by absorbing the latent heat and migrating along with the liquid flow. The mathematical model and its verification have been discussed and reported in our previous investigation [10] using the finite element method. Thus, a code and model similar to [10] are utilized to build a dataset of temperature distribution images for a class of input non-dimensional parameters. The non-dimensional parameters for the dataset and numerical code were Rayleigh number (Ra), Stefan number (Ste), non-dimensional phase change temperature TTF, and the volume fraction of nanoparticles Fi. The developed dataset includes 3290 sets of records with temperature distribution for various non-dimensional parameters.

Secondly, a DNN should be developed to receive the temperature distribution as the input and the classes of non-dimensional parameters as the outputs. Then, a training procedure should be executed to teach the DNN how to classify the non-dimensional parameters for a temperature distribution image. The training procedure manipulates the DNN coefficients to map the temperature images fairly to a class of non-dimensional parameters. An essential step in the training process is defining a loss function, which will be discussed later.

2.1 Dataset

The dataset of Edalatifar et al. [28], which consisted of 3290 samples, is utilized with some changes to generate a new dataset in this study. The previous dataset of Edalatifar et al. [28] is accessible at: <https://data.mendeley.com/datasets/jp96vj3frz> [29]. Each sample of the base dataset has four images of 128×128 as output data consisting of

x and y direction velocity (x , y), distribution of temperature (θ), and pressure field (P). In order to generate a new dataset, only temperature distribution (θ) was added to the current dataset as input. The dataset of Edalatifar et al. [28] has four inputs in each sample, including the Rayleigh number (Ra), Stefan number (Ste), non-dimensional phase change temperature (TTF), and volume fraction of nanoparticles (Fi). The inputs were quantized from zero to four in five levels and added to the new dataset as outputs. The inputs were resized between 0 and 5 linearly in the quantization process and then rounded to lower integer numbers.

In summary, a sample of the new dataset has one normalized temperature image between 0 and 1 as input and four outputs consisting of Fi, TTF, Ra, and Ste. Here, a normalized function is utilized to scale the input images in the range of zero to one as:

$$\bar{x}_0 = \frac{x_0 - \min(x_0)}{\max(x_0) - \min(x_0)} \quad (1)$$

where x_0 is the input image and \bar{x}_0 is the normalized image. The range of non-normalized values of the original (base) dataset [28] and the range of data in the new dataset are represented in Table 1. It should be noted that in [28], the model parameters were available; and then, a DNN was used to estimate the solution. Thus, a direct approach was used to estimate the physical solution. However, in the present study, a few physical solution aspects were assumed available, and then, the DNN estimates the model

Table 1 a Range of non-normalized values of the original dataset Edalatifar et al. [28] and b the range of data in the new dataset

(a)			
	Variable	Min	Max
Output	V	− 74.39	74.39
	U	− 40.35	40.35
	θ	0.00	1.00
	P	− 366,527.83	336.08
Input	Fi	0.00	0.05
	Ste	1.00	5.00
	TTF	0.05	0.95
	Ra	1,000	100,000
(b)			
Output (natural number)	Fi	1	4
	Ste	1	4
	TTF	1	4
	Ra	1	4
Input	θ	0.00	1.00

parameters. This is an important step in physical problems, known as inverse problems, where the physical signs and phenome are available, but the original set of parameters and setting for such a physical observation is not available.

As a part of the present manuscript, the generated new dataset is published and is available at the following address: <https://data.mendeley.com/datasets/jp96vj3frz/draft?a=52cc5952-de3c-4106-acb8-c03df9459dc7> [30]. The data are available in DAT (MATLAB) and NPZ (NumPy of Python) format; each file consists of 70% training data, 15% testing data, and 15% validation data.

2.2 Deep neural network (DNN) structure

As mentioned above, a sample of the new dataset has a distribution of temperature image (θ) as input and four numeric parameters (Fi, TTF, Ste, Ra) as output. Each output separately predicts with a distinct DNN to have an accurate prediction. Figure 1 depicts the structure of the DNN. As shown in Fig. 1, the DNN is constructed in three parts. The first part consists of 12 convolutional layers. A Batch Normalization (BN) [31] layer and a Rectified

Linear Unit (ReLU) [32] were added after each convolutional layer. Deep convolutional neural networks with ReLU active functions could be trained several times faster than their equivalents with *tanh* units. Krizhevsky et al. [33] stated that ReLUs have the desirable property of not requiring input normalization to prevent them from saturating. Following the impressive results of the ImageNet [33], the structure of the present deep neural network used the same principal structure as ImageNet. Thus, after each convolution and dense layer of ImageNet a ReLU active function was added.

A convolutional layer mathematically is a collection of small matrixes, i.e., filters that convolve with the layer's input data. The filters' size and stride are two main characteristics of the BN layers. The common size of 3×3 or 5×5 was used in filters, and the stride was the size of movement of filters in the process of convolution as well. For all convolutional layers in this study, the size of filters was set on 3×3 , as given in Fig. 1. In addition; the stride was set to 1 when outputs and inputs of a layer must have the same size; otherwise, it was set on 2 to reduce the output size to half.

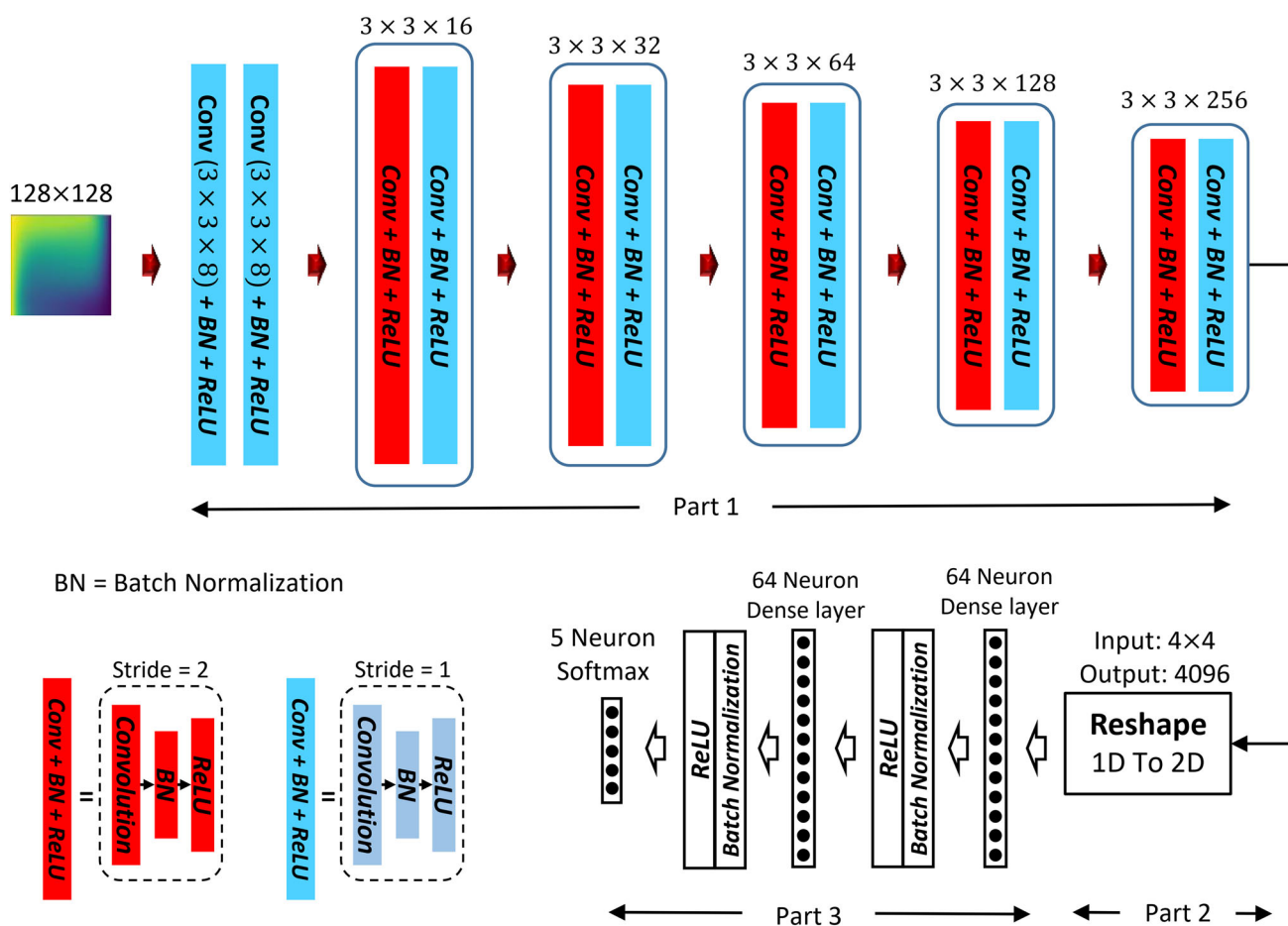


Fig. 1 DNN architect and block parts utilized in the current research

The features are extracted from the input data in Part 1. Since the inputs are images, the 2D convolutional layers require. This kind of layer results in 2D features. Part 2 prepares the output of Part 1 for entrance to Part 3 by reshaping the 2D features into a 1D array. Part 3 is a classifier, which gets the extracted features from Part1 and determines a data's class. It makes the two dense layers with 64 neurons, followed by a 5 neurons dense layer with a softmax activation function. Similar to Part 1, a BN and ReLU were added after the first and second dense layer of Part 3.

The dense layers, i.e., fully connected layers, mathematically are a matrix containing free parameters. The following equation computes the output of a dense layer:

$$Y_{l,k} = \sum_{i=1}^m W_{l,i,k} \times X_{l,i} + b_{l,k} \quad (2)$$

where W and b are free parameters. Also, X_l and Y_l are the input and output of layer l , respectively. The equation of ReLU is as below:

$$Y = \max(0, X) \quad (3)$$

The relationship between X and Y is shown in Fig. 2. From Eq. 2, infer that Y is X when the minus values of X are replaced with zero.

3 Results and discussion

This paper generates a new dataset with one image of the temperature distribution (θ) as input and four non-dimensional parameters of Fi , TTF , Ste , and Ra as outputs. Afterward, a new DNN is used to predict the output parameters according to the input image (Fig. 1). Each parameter is predicted with a distinct DNN for better accuracy. Therefore, four DNNs with similar structures are trained so that each DNN predicts one of the outputs.

DNNs have free parameters, which should be adapted to predict accurately. It is done in a procedure called the training process. The training process starts with the random initialization of free parameters, but when two

datasets have interdependency, transfer learning [34] enhances the estimation accuracy. Transfer learning refers to the process of taking knowledge or skills learned in one domain and applying them to a related but different domain. It is a simple but effective technique for transferring knowledge from a well-trained DNN to another DNN in a similar domain. The process of transfer learning consists of the initial adjustment of all or some free parameters of a new DNN according to an excellent trained DNN instead of the random initialization. All or part of the first layers can be used to provide better training for DNNs.

In this study, two scenarios are considered for training DNNs. First, the free parameters of each DNN are randomly initialized with the Glorot uniform technique [35]. In the second scenario, the transfer learning technique is examined since the outputs of the dataset are interdependent. Therefore, the second scenario starts with training a DNN to predict Ra with initial free parameters using the Glorot uniform technique. Then their weights are transferred into three new DNNs before starting the training processes. Finally, those DNNs are trained to predict TTF , Fi , and Ste .

All data in this study are generated with Python and the DNNs built-in Keras [36] as well. The settings for training are illustrated in Table 2. All results reported in this section are extracted from an epoch with maximum accuracy of validation data. Therefore, it guarantees that those results are not associated with an overfit epoch.

Figures 3 and 4 show the variation of accuracy and loss error of the validation data throughout the training process when it does not use transfer learning (TL). These plots show while Ra , TTF , and Fi rapidly reach the almost final accuracy and less loss value, the accuracy of Ste slowly moves to the ultimate value. However, Ste and TTF could never reach the accuracy and loss of Ra and Fi . These phenomena reveal that the prediction of Ste is harder than others, and input information is not enough to accurately predict Ste and TTF . Moreover, Ra is the most predictable parameter, and DNN could predict it with high accuracy of about 100%. Similar results can be obtained from Fig. 4.

Tables 3 and 4 represent the accuracy and loss of DNNs when the processes of training are started with TL and randomly initialized weights (No TL). These results show that the accuracy of testing data enhances by using TL at about 3.65%, 0.4%, and 16.6% for TTF , Fi , and Ste , respectively. Therefore, the lack of information for training DNNs is compensated with transfer knowledge of training Ra to other DNNs using the transfer learning technique. Moreover, transfer learning considerably influences the prediction of TTF and especially Ste , but it cannot significantly change the accuracy of Fi . Table 3 shows that represented DNNs have acceptable accuracy in predicting

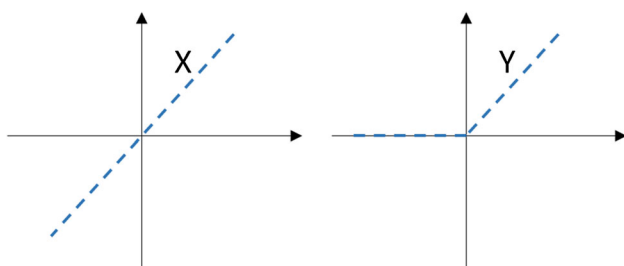
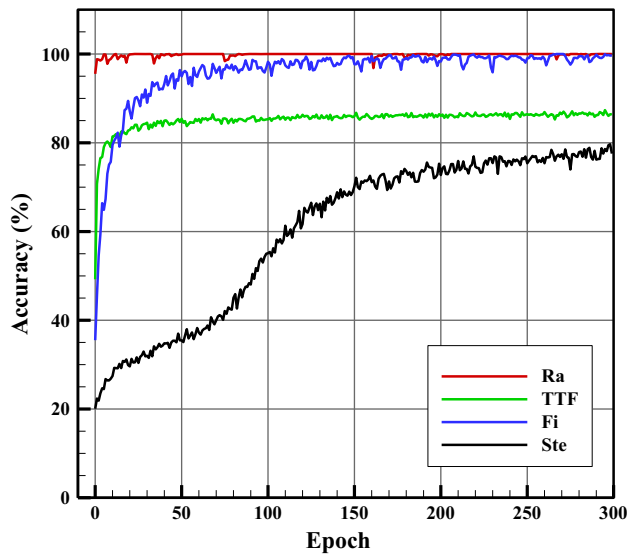
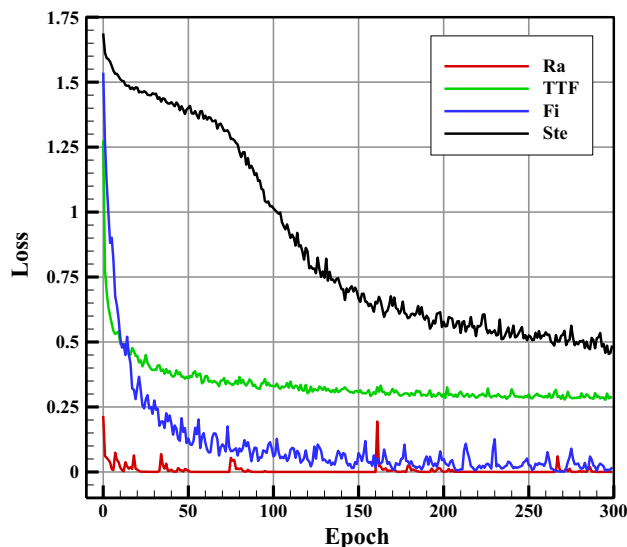


Fig. 2 Performance of ReLU activation function. X is input, and Y is output

Table 2 Training settings

Optimizer		Other settings	
Name	Adam	Batch	32
Learning rate	0.001	Epoch	300
β_1	0.9	Initializer	Glorot uniform technique [35] and transfer weights
β_2	0.999	Loss function	Categorical cross-entropy

**Fig. 3** Accuracy of validation data during training procedures while the training processes start without TL**Fig. 4** Validation data loss during the training procedures while the training processes start without TL

Ra and *Fi*, but *Ste* and *TTF* predict with low accuracy, especially when the training process starts without TL. As a result, more input information is needed to train DNNs of *TTF* and *Ste*. Note that the value of the training data in Table 3 for *Ste* and *TTF* are very lower than their values in

Table 3 Accuracy of DNNs using TL and not using TL (No TL)

Data type		<i>Ra</i> (%)	<i>TTF</i> (%)	<i>Fi</i> (%)	<i>Ste</i> (%)
Training	No TL	99.96	72.99	100.00	44.94
	TL	–	77.38	99.96	67.43
Validation	No TL	100.00	72.41	93.51	40.97
	TL	–	78.09	91.68	60.65
Testing	No TL	99.80	73.48	93.32	44.74
	TL	–	77.13	93.72	61.34

Fig. 3 since the value of Table 3 is reported for an epoch with maximum validation data accuracy.

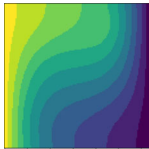
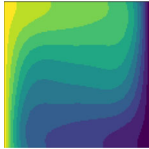
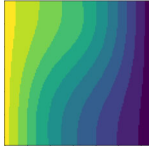
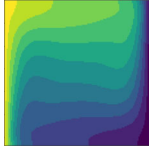
Figure 5 gives a better view of the transfer learning influences on the training. Figure 5 shows the variation of the training data for *TTF*, *Fi*, and *Ste* from epoch 10 to 300. This plot reveals that the information of DNN of *Ra* that transfers to other DNNs with the TL technique increases the rate of accuracy improvement for *Ste*, but this improvement is not significant for *TTF* and *Fi*. However, according to Table 3, the TL can increase ultimate results.

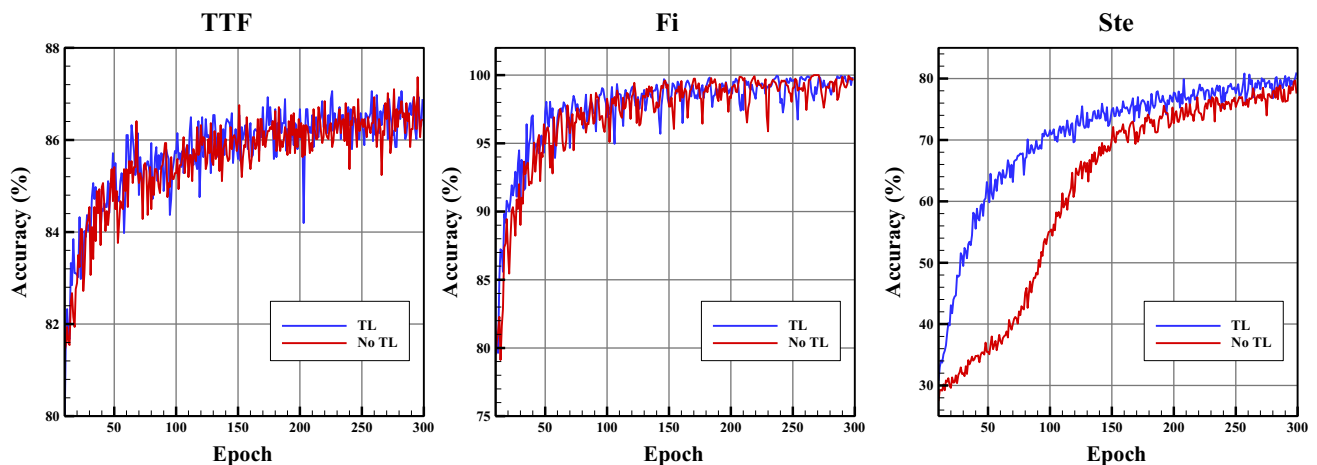
Table 4 illustrates some samples of test data randomly picked from the dataset. From the left, column 1 shows the input image to the DNN, and columns 2 to 4 show the real values of outputs, predicted values without TL, and with TL, respectively.

4 Conclusion

This study provides insights into the capability of DNNs in estimating the characteristics and parameters of the natural convection heat transfer of NEPCM suspension. A dataset containing 2,390 samples was generated. Each sample of the dataset had one temperature distribution (θ) as input, and four non-dimensional parameters consisted of the Rayleigh number (*Ra*), Stefan number (*Ste*), non-dimensional phase change temperature (*TTF*), and volume fraction of nanoparticles (*Fi*) as the outputs. Since the goal of this paper was classification, the outputs were quantized into five levels, and a new DNN was utilized to predict these classes by using the temperature distribution image. For better accuracy, each output was predicted with a DNN. Therefore, four DNNs were used to predict outputs.

Table 4 Predicted samples of the dataset with DNNs

Input (θ)	Real output	No TL predict output	TL predict output
	Ra = 0 TTF = 3 Fi = 3 Ste = 3	Ra = 0 TTF = 3 Fi = 3 Ste = 3	Ra = 0 TTF = 3 Fi = 3 Ste = 3
	Ra = 1 TTF = 3 Fi = 0 Ste = 1	Ra = 1 TTF = 3 Fi = 0 Ste = 0	Ra = 1 TTF = 0 Fi = 0 Ste = 1
	Ra = 0 TTF = 3 Fi = 4 Ste = 3	Ra = 0 TTF = 3 Fi = 4 Ste = 0	Ra = 0 TTF = 3 Fi = 4 Ste = 3
	Ra = 4 TTF = 0 Fi = 1 Ste = 4	Ra = 4 TTF = 0 Fi = 1 Ste = 2	Ra = 4 TTF = 0 Fi = 1 Ste = 4

**Fig. 5** Variation of training data accuracy while using TL and not using TL (No TL)

The training processes were done in the form of two scenarios. The first scenario randomly started training with initial free parameters (DNNs weights). Since outputs had interdependency, free parameters were initialized with the transfer learning technique in the second scenario. The main finding of the current study can be listed as follows:

1. The present DNN predicts Ra and Fi testing data with high accuracy, about 99.8, and 93.32%, respectively. For TTF, the accuracy was not very high, at 73.48%. However, this DDN is not able to predict Ste with acceptable accuracy.
2. The transfer learning improved accuracy by about 3.65, 0.4, and 16.6% for TTF, Fi, and Ste, respectively. Therefore, the lack of information for trained DNNs

was compensated by transferring the knowledge of training Ra to other DNNs using the transfer learning technique.

3. The results reveal that DNNs need more input information to enhance the accuracy of TTF and Ste as well.

The present study assumes the availability of only a few physical solution aspects, and the DNN is used to estimate the model parameters. This step is crucial in solving inverse problems, which deal with physical observations where the physical signs and phenomena are known, but the original parameters and setting for such observations are not available. The current research shows a DNNs can be utilized for physical parameter classification using input flow and heat transfer patterns. The flow and heat transfer

patterns are nonlinearly coupled to the physical parameters; and hence, their classification is a difficult task without any prior knowledge about their governing differential equations. The low accuracy of the DNN for classifying Stefan number was improved by using the transfer learning approach. Thus, it is evident that a dataset with a higher number of images could be the subject of future studies.

Moreover, the DNN classification could be done on a desktop computer in a few seconds. Thus, the current approach could be promising to be used in monitoring systems applications to detect and control the physical characteristics parameters. In the present study, a ReLU activation function was used. However, other types of activation functions, such as *tanh*, can be employed in the DNN design. The type of activation function can impact the DNN speed and performance. Thus, this subject can be investigated comprehensively in future studies.

Acknowledgements This research of Mohammad Ghalambaz and Mikhail Sheremet was supported by the Tomsk State University Development Program (Priority-2030).

Data availability The datasets generated during and/or analyzed during the current study are available in the Mendeley repository (<https://doi.org/10.17632/jp96vj3frz.1>), <https://data.mendeley.com/datasets/jp96vj3frz>.

Declarations

Conflict of Interest The authors clarify that there is no conflict of interest to report.

References

- Hemmat Esfe M, Bahiraei M, Hajbarati H, Valadkhani M (2020) A comprehensive review on convective heat transfer of nanofluids in porous media: energy-related and thermohydraulic characteristics. *Appl Therm Eng* 178:115487
- Gürdal M, Arslan K, Gedik E, Minea AA (2022) Effects of using nanofluid, applying a magnetic field, and placing turbulators in channels on the convective heat transfer: a comprehensive review. *Renew Sustain Energy Rev* 162:112453
- Rekha MB, Sarris IE, Madhukesh JK, Raghunatha KR, Prasanakumara BC (2022) Impact of thermophoretic particle deposition on heat transfer and nanofluid flow through different geometries: an application to solar energy. *Chinese J Phys* 80:190–205
- Altunay FM, Pazarlıoğlu HK, Gürdal M, Tekir M, Arslan K, Gedik E (2022) Thermal performance of Fe_3O_4 /water nanofluid flow in a newly designed dimpled tube under the influence of non-uniform magnetic field. *Int J Therm Sci* 179:107651
- Nada SA, El-Zoheiry RM, Elsharnoby M, Osman OS (2022) Enhancing the thermal performance of different flow configuration minichannel heat sink using Al_2O_3 and CuO -water nanofluids for electronic cooling: an experimental assessment. *Int J Therm Sci* 181:107767
- Wang H, Chen X (2022) Numerical simulation of heat transfer and flow of Al_2O_3 -water nanofluid in microchannel heat sink with cantor fractal structure based on genetic algorithm. *Anal Chim Acta* 1221:339927
- Samiezadeh S, Khodaverdian R, Doranehgard MH, Chehrmonavari H, Xiong Q (2022) CFD simulation of thermal performance of hybrid oil- $\text{Cu-Al}_2\text{O}_3$ nanofluid flowing through the porous receiver tube inside a finned parabolic trough solar collector. *Sustain Energy Technol Assess* 50:101888
- Khodadadi M, Ali Farshad S, Ebrahimpour Z, Sheikholeslami M (2021) Thermal performance of nanofluid with employing of NEPCM in a PVT-LFR system. *Sustain Energy Technol Assess* 47:101340
- Yazdanifard F, Ameri M, Taylor R (2021) Parametric investigation of a nanofluid-NEPCM based spectral splitting photovoltaic/thermal system. *Energy Convers Manage* 240:114232
- Ghalambaz M, Chamkha AJ, Wen D (2019) Natural convective flow and heat transfer of nano-encapsulated phase change materials (NEPCMs) in a cavity. *Int J Heat Mass Transf* 138:738–749
- Hussain S, Molana M, Armaghani T, Rashad A, Nabwey HA (2022) Energy storage performance and irreversibility analysis of a water-based suspension containing nano-encapsulated phase change materials in a porous staggered cavity. *J Energy Storage* 53:104975
- Alhashash A, Saleh H (2022) Free convection flow of a heterogeneous mixture of water and nano-encapsulated phase change particle (NEPCP) in enclosure subject to rotation. *J Energy Storage* 51:104168
- Güllü H, Canakci H, Alhashemy A (2019) A ranking distance analysis for performance assessment of UCS versus SPT-N correlations. *Arab J Sci Eng* 44(5):4325–4337
- Güllü H, Canakci H, Alhashemy A (2018) Use of ranking measure for performance assessment of correlations for the compression index. *Eur J Environ Civ Eng* 22(5):578–595
- Matheswaran MM, Arjunan TV, Muthusamy S, Natrayan L, Panchal H, Subramaniam S, Khedkar NK, El-Shafay A, Sonawane C (2022) A case study on thermo-hydraulic performance of jet plate solar air heater using response surface methodology. *Case Stud Therm Eng* 34:101983
- Güllü H, Fedakar Hİ (2017) Response surface methodology for optimization of stabilizer dosage rates of marginal sand stabilized with sludge ash and fiber based on UCS performances. *KSCE J Civ Eng* 21(5):1717–1727
- Güllü H (2017) A new prediction method for the rheological behavior of grout with bottom ash for jet grouting columns. *Soils Found* 57(3):384–396
- Güllü H (2017) A novel approach to prediction of rheological characteristics of jet grout cement mixtures via genetic expression programming. *Neural Comput Appl* 28(1):407–420
- Gullu H (2017) On the prediction of unconfined compressive strength of silty soil stabilized with bottom ash, jute and steel fibers via artificial intelligence. *Geomech Eng* 12(3):441–464
- Legaard C, Schranz T, Schweiger G, Dragoña J, Falay B, Gomes C, Iosifidis A, Abkar M, Larsen P (2023) Constructing neural network based models for simulating dynamical systems. *ACM Comput Surv* 55(11):1–34
- Yang Q, Wang Z, Guo K, Cai C, Qu X (2023) Physics-driven synthetic data learning for biomedical magnetic resonance: the imaging physics-based data synthesis paradigm for artificial intelligence. *IEEE Signal Process Mag* 40(2):129–140
- Håstad J, Goldmann M (1991) On the power of small-depth threshold circuits. *Comput Complex* 1(2):113–129
- Ermis K, Ereğ A, Dincer I (2007) Heat transfer analysis of phase change process in a finned-tube thermal energy storage system using artificial neural network. *Int J Heat Mass Transf* 50(15):3163–3175
- Azwadi CSN, Zeinali M, Safdari A, Kazemi A (2013) Adaptive-network-based fuzzy inference system analysis to predict the

- temperature and flow fields in a lid-driven cavity. *Numeri Heat Transf, Part A Appl* 63(12):906–920
25. Akbari E, Karami A, Nazari S, Ashjaee M (2020) An intelligent integrated approach of Jaya optimization algorithm and neuro-fuzzy network to model the natural convection in an open round cavity. *Int J Model Simul* 40(2):87–103
 26. Selimefendigil F, Akbulut Y, Sengur A, Oztop HF (2020) MHD conjugate natural convection in a porous cavity involving a curved conductive partition and estimations by using long short-term memory networks. *J Therm Anal Calorim* 140(3):1457–1468
 27. Zhou S, Liu X, Du G, Liu C, Zhou Y (2019) Comparison study of CFD and artificial neural networks in predicting temperature fields induced by natural convection in a square enclosure. *Therm sci* 23:3481–3492
 28. Edalatifar M, Tavakoli MB, Setoudeh F (2022) A deep learning approach to predict the flow field and thermal patterns of nonencapsulated phase change materials suspensions in an enclosure. *J Appl Comput Mech* 8(4):1270–1278
 29. Edalatifar M, Ghalambaz M, Tavakoli MB, Setoudeh F (2023) deep learning approach to natural convection heat transfer in a cavity: a simulation dataset for nano-encapsulated phase change material suspensions. Mendeley Data. <https://doi.org/10.17632/j5f6r56jnb.1>
 30. Ghalambaz M, Edalatifar M, Moradimaryamnegari S, Sheremet M (2023) Nano-PCM intelligence classification dataset for energy storage and heat transfer analysis using deep neural networks. Mendeley Data. <https://doi.org/10.17632/jp96vj3frz.1>
 31. Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift, arXiv preprint [arXiv:1502.03167](https://arxiv.org/abs/1502.03167).
 32. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*.
 33. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. *Commun ACM* 60(6):84–90
 34. Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C (2018) A survey on deep transfer learning. In: Kůrková V, Manolopoulos Y, Hammer B, Iliadis L, Maglogiannis I (eds) *Artificial neural networks and machine learning–ICANN 2018*. Springer, Cham
 35. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*.
 36. Chollet, F., Others, *Keras*. 2015, <https://keras.io>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.