IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 June 2023, accepted 6 July 2023, date of publication 13 July 2023, date of current version 20 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3294976

RESEARCH ARTICLE

Task Transfer Learning for Prediction of Transient Nitrogen Oxides, Soot, and Total Hydrocarbon Emissions of a Diesel Engine

SEUNGHYUP SHIN[1], MINJAE KIM[2], JIHWAN PARK[3], SANGYUL LEE[4], AND KYOUNGDOUG MIN[3], (Member, IEEE)

[1]Department of Artificial Intelligence, Sejong University, Seoul 05006, Republic of Korea
[2]Department of Mechanical Engineering, Myongji University, Yongin-si 17058, Republic of Korea
[3]Department of Mechanical Engineering, Seoul National University, Seoul 08826, Republic of Korea
[4]Division of Mechanical and Electronics Engineering, Hansung University, Seoul 02876, Republic of Korea

Corresponding authors: Sangyul Lee (sangyul.lee@hansung.ac.kr) and Kyoungdoug Min (kdmin@snu.ac.kr)

This work was supported in part by the Faculty Research Fund of Sejong University, in 2023; in part by the Institute of Advanced Machinery and Design (IAMD) of Seoul National University; and in part by the Research Project under Grant 2021R1F1A1063188.

ABSTRACT According to previous studies on internal combustion engine applications using deep learning, deep learning model should be individually optimized and trained to predict different phenomena. This study introduces task transfer learning to predict transient nitrogen oxides (NO_x), soot, and total hydrocarbon (THC) emissions, which are the major emissions from diesel engines. Using the concept of task transfer learning, when there is a pretrained model relevant to the target task, the model can be transferred to predict another phenomenon by training only the last two layers with hyperparameters of the pretrained model. This concept omits the need for optimizing and training separate models that can save computational time and cost. The results of task transfer learning were evaluated using Worldwide Harmonized Light Vehicles Test Procedure (WLTP) cycle data, which are representative transient cycles of the internal combustion engine, and all possible transfer cases with NO_x, soot, and THC emissions were investigated. The R^2 values of pretrained NO_x, soot, and THC models were 0.9780, 0.9215, and 0.9390, respectively. The R^2 gaps between the pretrained and transferred models were within 0.012, with a value of 0.0015 for the NO_x emission, 0.011 for the soot emission, and 0.0115 for the THC emission. The relative mean absolute errors (MAEs) to the maximum emission values were approximately 0.57-0.82% for NO_x emissions, 0.69-2.02% for soot emissions, and 1.52-2.42% for THC emissions. These accuracy results were comparable to the accuracy of the emission measurement device, which was better than that of the sensors for practical use in vehicles. The results indicated that task transfer learning was valid for predicting emissions of an internal combustion engine, and it achieved efficient organization of prediction models using a pretrained model.

INDEX TERMS Deep learning, diesel engines, task transfer learning, transient emissions.

I. INTRODUCTION

As public attention on air quality has recently increased, emission regulations are becoming increasingly stringent to reduce harmful emissions from internal combustion engines. EURO 6, validated in 2014, defines emission standards including those for nitrogen oxides (NO_x), particulate matter (PM), and total hydrocarbon (THC) for gasoline and diesel vehicles [1]. The test cycle for the regulation changed from the New European Driving Cycle to the Worldwide Harmonized Light Vehicles Test Procedure (WLTP), which involves a more complex driving profile to reflect real driving

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

conditions. Therefore, it is difficult to predict emissions under WLTP cycles.

A. LIMITATIONS OF DEEP LEARNING STUDIES FOR PREDICTION OF INTERNAL COMBUSTION ENGINE

Before rising of deep learning, many statistical methodologies were applied to predict various phenomena of internal combustion engines. Response surface methodology was used to predict and optimize engine performance and emissions [2]. Other methodologies such as analysis of variance technique [3] and least-squares support vector machine [4] were also utilized for modeling for performance and emissions of diesel engines.

Recently, deep learning has achieved high accuracy in the prediction of images and natural language processing [5]. It has been actively applied to predict engine phenomena, and previous studies can be divided into single-task and multi-task problems. For a single-task problem, an individual model is used to predict only one phenomenon, and the model has one output. On the other hand, the multi-task problem is defined as a model that is organized to predict multiple phenomena simultaneously with multiple outputs.

The single-task approach is a basic concept for deep learning models to predict specific phenomena. Deep learning models recently present higher accuracy compared to conventional equation-based models in many research areas. However, considerable computational cost and time are required to predict multiple phenomena because multiple models need to be optimized and trained.

Single-task models are typically employed to predict the performance or emissions of engines. Steady-state NO_x emissions were predicted using deep neural networks (DNN) and genetic algorithms [6], where some hyperparameters of the DNN model, such as the learning rate and epoch size, were optimized using a genetic algorithm to achieve the highest accuracy of the model. Cold start emissions of diesel vehicles, such as carbon dioxide (CO_2), NO_x, and THC, were predicted by artificial neural network (ANN) models [7]. Separated ANN models were trained to predict four emissions respectively with engine coolant temperature, vehicle velocity, vehicle specific power, engine speed, and engine torque as input variables. Using the trained models, three engine coolant temperature scenarios were investigated to evaluate cold start effect on the emissions. The accuracies of NO_x emissions from diesel engines were compared for the neural network and nonlinear regression models [8]. The data were clustered into several groups, and NO_x emissions were predicted for these groups. The accuracies of both models were similar, but the neural network model was better for dealing with data, not from a specific group. Long short-term memory (LSTM) was employed to predict the transient NO_x emissions of a diesel engine [9]. The LSTM model was evaluated using other machine learning algorithms such as random forest and support vector regressors. Another study was investigated the accuracies of DNN and LSTM models for NO_x emissions under transient conditions [10]. The accuracy of the LSTM model was slightly higher than that of the DNN model, and time-series data pre-processing was proposed to increase the accuracy of the DNN model to a level similar to that of the LSTM model. Models using a convolutional neural network (CNN) and LSTM were built to predict fuel flow, NO_x, and soot for a nonroad transient cycle [11]. The accuracy of the LSTM model was better than that of the CNN model to predict the transient cycle. Among the phenomena, the soot prediction had a lower accuracy that the NO_x and fuel flow prediction.

The multi-task approach is more efficient than the single-task approach because the optimization and training processes are not necessary to repeat for each output. However, the accuracy of some outputs from the multi-task model deteriorates as multiple phenomena are predicted using a single model. In addition, the multi-task model is vulnerable to sensor error when it is used for practical use. If there are errors in sensor data regarding a specific output, all outputs of the multi-task model cannot be reliable. However, the single-task model can be fixed by replacing a model relating to the unreliable output.

Multi-task models have mainly been applied for performance and emission prediction. The effects of biodiesel on exhaust emissions have been studied using an ANN [12]. The type of biodiesel fuel and concentration of magnesium oxide nanoparticles were the independent variables affecting CO, THC, and NO_x emissions. A 2-layer ANN model was organized with multiple input and output structures to predict the performance, emissions, and vibrations of a compressed-ignition engine with biodiesel fuels [13]. The layer configuration was tuned to achieve higher accuracy of the output variables, including torque, power, CO, CO_2, THC, NO_x, etc. The distribution of accuracy results was between an R^2 value of 0.88-0.98, according to the outputs. Some outputs had limited accuracy because they were predicted simultaneously with other variables by a model. A deep learning procedure was investigated to predict the performance, combustion, and emissions of a gasoline engine [14]. The study introduced the pipeline modeling concept, in which the abnormal combustion cases are filtered out before the main prediction model for the outputs.

B. TASK TRANSFER LEARNING

To compensate for the drawbacks of single- and multi-task approaches, task transfer learning was introduced in this study. Task transfer learning was rarely applied to predict phenomena of the internal combustion engine, but it is widely studied in image and natural language processing areas.

Task transfer learning is a methodology in which a trained model for predicting of a specific phenomenon, that is, a pretrained model, is transformed into a model for predicting another phenomenon by fine-tuning.

For image processing, the model for ImageNet 1000-class classification was transferred to classify the Cifar 100-class

dataset [15]. The authors introduced information theory for task transfer learning and defined the H-score to quantify the feature transferability among tasks. Computational taxonomic maps have been studied for task transfer learning [16]. 26 tasks in computer vision tasks were formed in a task dictionary to cover common themes for image processing. Then, the relationships between the training and target tasks were investigated by evaluating the transfer-learning dependency across the dictionary. A representation similarity analysis method was developed for task transfer learning of image classification [17]. Similarity scores among tasks were computed using the defined correlation between the models trained on different tasks. This task transfer learning for image processing has been applied to various industries, including medicine, beauty, and construction, to solve practical problems. Breast cancer in mammograms was diagnosed using a combination of CNN and multi-task transfer learning [18]. The knowledge of the model learned from nonmedical images was translated into medical diagnostic tasks. This task transfer learning increases the generalization capability of the model. Surgical task segmentation has also been performed for medical applications [19]. The features and segmentation points from manually labeled data were used to learn the segmentation policies. The policies were employed to segment new tasks through transfer learning. The classification task of construcion material images was conducted by transfer learning of CNN architectures [20]. AlexNet and GoogleNet were evaluated as pretrained CNN architectures for transfer learning. This study followed the basic transfer learning process using a fixed feature extractor and fine-tuning schemes.

For natural language processing, conditionally adaptive multi-task learning was studied to improve transfer learning in natural language processing [21]. Generally, the best performance is achieved by organizing a separate model for each task. The authors pointed out that previous approaches caused overfitting to low-resource tasks, catastrophic forgetting, negative task transfer, and learning interference. Therefore, task-conditioned modules that facilitate weight sharing were suggested by keeping half of the weights of a pretrained model for efficient parameter sharing and mitigating forgetting. Cross-task transfer learning has been applied to deep-speech enhancement models [22]. The aim of this study was to improve the listening quality of speech and boost the noise robustness of speech-recognition systems. A multi-condition senone classifier trained by noisy speech features and a clean-condition senone classifier trained by enhanced speech features were combined with a deep speech enhancement model with robustness to unseen background noise.

C. OBJECTIVES

The objectives of this study were to apply task transfer learning to predict the emissions of diesel engines and verify their viability. Task transfer learning introduced from other research fields could be an effective solution for the limitation of single-task and multi-task approaches that previous deep learning studies for internal combustion engines had. The method for task transfer learning of internal combustion engines was suggested using existing weights during retraining with hyperparameters of the pretrained model. This method provided efficient organization of models for different emissions.

The targe emissions were transient NO_x, soot, and THC, the significant emissions of internal combustion engines, obtained under WLTP cycles.

Fig. 1 shows the cases investigated in this study. The accuracy results of task transfer learning were compared to models individually optimized (Model 1 of each emission) and trained for target emissions (Model 2 and 3 of each emission transferred from the pretrained models of the other emissions). For example, the NO_x model that was optimized from scratch was built as a reference model (NO_x model 1) for comparison with the transfer learning results. Subsequently, the transferred NO_x models (NO_x model 2 and NO_x model 3) were trained using task transfer learning from pretrained soot (Soot Model 1) and THC models (THC Model 1), respectively. The accuracies of the three models were compared to evaluate the applicability of transfer learning to emission prediction. The accuracies of the soot and THC predictions were similar to those of the NO_x case. All possible transfer cases with the three emissions were investigated in this study.

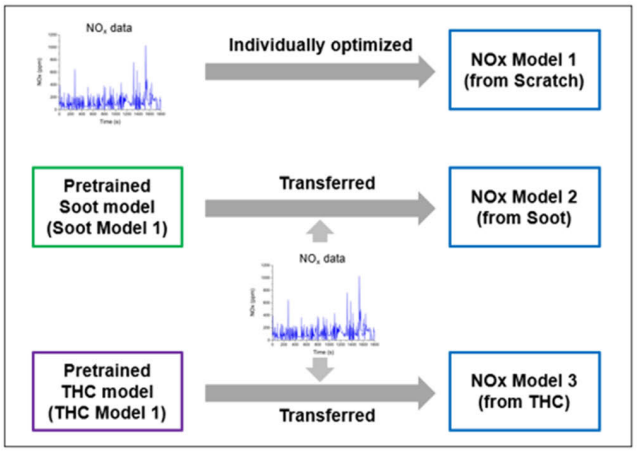
The main contributions of this study are as follows:

- This study applied task transfer learning to predict emissions of the internal combustion engine. This approach for predicting the internal combustion engine was rarely studied by previous research.
- This study showed that the accuracy results from task transfer learning were similar to those of the trained models for each emission. This proves that task transfer learning is valid for predicting internal combustion engine phenomena.
- This study organized the process of task transfer learning for emissions in an internal combustion engine. For task transfer, the last two hidden layers of the pretrained model should be trained for the target emission, whereas the other layers are frozen during training. The hyperparameters of the pretrained model do not need to be changed for task transfer learning.
- This study suggests a methodology for compensation of both single- and multi-task approaches using task transfer learning. Efficient organization of prediction models is achieved if there is a pretrained model, using existing weights, and without the need of an optimization process.

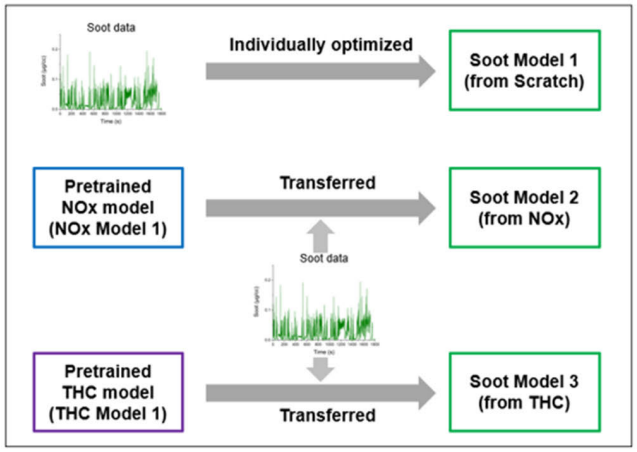
II. METHODOLOGY

A. METHOD OF TASK TRANSFER LEARNING

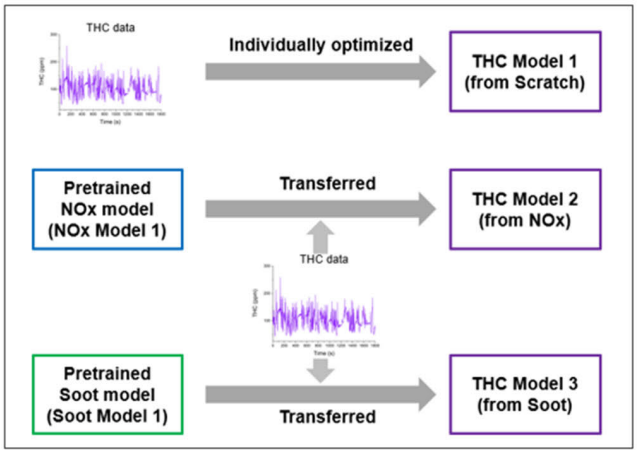
Transfer learning is classified by the relationships of the domain and task between the source and target as shown in



(a)



(b)



(c)

FIGURE 1. Investigation cases for task transfer learning in this study: (a) NO_x case, (b) Soot case, and (c) THC case.

Table 1 [23]. The relationships of domains and tasks between source and target are used to define the traditional machine learning and transfer learning.

The definition of inductive transfer learning is that the learning of the target prediction function $f_T(\cdot)$ in D_T is

TABLE 1. Relationship between traditional machine learning and various transfer learning settings [23].

Learning Settings		Source and Target Domains	Source and Target Tasks
Traditional Machine Learning		the same	the same
Transfer Learning	Inductive Transfer Learning / Unsupervised Transfer Learning	the same	different but related
		different but related	different but related
	Transductive Transfer Learning	different but related	the same

performed using knowledge in D_S and T_s, where $T_S \neq T_T$. Here, D_S is the source domain, T_S is the learning task, D_T is the target domain, and T_T is the target task. In this study, labeled data were available in the source domain.

The problem of multi-task learning was proposed for support vector machines [24], and it was modified for inductive transfer learning [23]. In inductive transfer learning,

$$w_s = w_0 + v_s \quad and \quad w_T = w_0 + v_T \tag{1}$$

where w_S is the parameter for the source task, and w_T is the parameter for the target task. v_S and v_T are specific parameters for each source task and target task, respectively, while w_0 is a common parameter. The transfer learning of support vector machines can be formulated as follows [23].

$$\begin{aligned} &\min_{w_0, v_t, \xi_{t_i}} J(w_0, v_t, \xi_{t_i}) \\ &\quad = \sum_{t \in S,T} \sum_{i=1}^{n_t} \xi_{t_i} + \frac{\lambda_1}{2} \sum_{t \in S,T} \| v_t \|^2 + \lambda_2 \| w_0 \|^2 \\ &s.t.\ y_{t_i} (w_0 + v_t) \cdot x_{t_i} \geq 1 - \xi_{t_i} \\ &\quad \xi_{t_i} \geq 0, \quad i \in \{1, 2, \cdots, n_t\} \quad and \quad t \in \{S, T\} \end{aligned} \tag{2}$$

Here, λ_1 and λ_2 are positive regularization parameters; ξ_{ti} are slack variables measuring the error that each of the final model w_t makes on the data; $J(\cdot)$ is the cost function; and $||\cdot||$ is the Euclidean distance. S and T are tasks in target and source domain, respectively. x_{ti} and y_{ti} indicate the i^{th} term of input and output vectors included in S and T. By applying task transfer learning, v_T is trained for the target task with a pretrained parameter, w_0.

For the deep learning model, task transfer learning for image classification has been derived [16], and can be modified for the regression applied in this study.

$$D_{s \to t} := arg \min_{\theta} \mathbb{E}_{R \in D}[L_t(D_\theta(E_s(R)), f_t(R))] \tag{3}$$

where R is the regression problem, $f_t(R)$ is the ground truth of t for the regression problem, L_t is the loss function, D_θ is the parameterized function, and $E_S(R)$ is the pretrained model for the source task.

Based on the fundamentals described in (3), task transfer learning was performed as the schematic shown Fig. 2. The

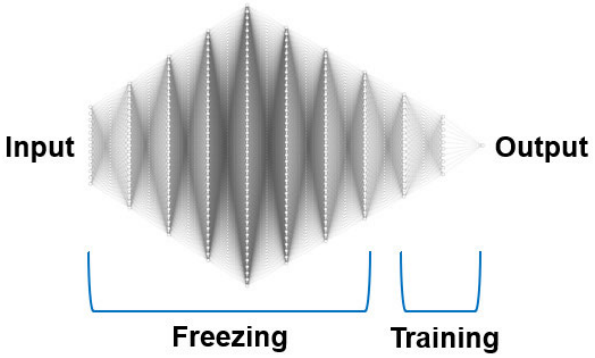


FIGURE 2. Frozen and trained layers for task transfer learning.

weights in the last two hidden layers of the pretrained model f_S were trained again using the target data T whereas the weights in the other hidden layer were frozen during training. In this process, the hyperparameters of the transferred model, including the configuration of the hidden layer, learning rate, and batch size, were the same as those of the pretrained model.

B. HYPERPARAMETERS

In this study, DNN models were utilized to predict emissions using transfer learning. The DNN structure has several hyperparameters that need to be determined. Some of the hyperparameters were optimized by Bayesian optimization and hidden-node determination logic [25], whereas the other hyperparameters were set at specific values.

TABLE 2. Minimum and maximum limits of hyperparameters for the optimization.

	Minimum value	Maximum value
Learning rate	10^{-7}	10^{-2}
Learning rate decay	10^{-9}	10^{-5}
Number of hidden layers	2	10
Number of the 1st hidden nodes	5	25
Batch size	20	72004

Table 2 presents the minimum and maximum limits of the hyperparameters for the optimization. The target hyperparameters for optimization were the learning rate, learning rate decay, number of hidden layers, number of 1st hidden nodes, and batch size. The hidden node arrangement was defined using the number of hidden layers and the number of 1st hidden nodes based on hidden-node determination logic [25]. This is because the number of iterations for the optimization can be exponentially increased if the number of hidden nodes is optimized using Bayesian optimization. Therefore, some logical equations were proposed to organize the structures of hidden layers, and the number of nodes in each hidden layer was determined by the number of hidden layers and the number of 1st hidden nodes. The maximum batch size was equal to the total number of data points consisting of four WLTP cycles, and a WLTP cycle included 18001 data points, which measured 0.1 s intervals for 1800 s. The data configuration is presented in the following section. The number of iterations for the Bayesian optimization was 300, and the epochs for each iteration were varied by early stopping callbacks [26]. The early stopping callback determines the epochs for the iteration to achieve the best accuracy of the model, that is, minimum validation loss. The training process was continued for 1000 more epochs (patience number) after the model achieved the best accuracy, and the early stopping callback restored the best model for the iteration.

Other hyperparameters such as the activation function, training optimizer, and batch normalization were fixed at specific values. A detailed description of these hyperparameters is provided in a previous study by the authors [10].

The exponential linear unit (ELU) function is introduced as an activation function as shown in (3) [27]. The ELU function is a modification of the rectified linear unit (RELU) function. The RELU function has a dying RELU problem in that the outputs of the function are all zero when the input data are below 0. This can interrupt the weight update of the node under certain conditions.

$$f(x) = \begin{cases} x & if\ x > 0 \\ \alpha\,(\exp(x) - 1) & if\ x \le 0 (\alpha < 0) \end{cases} \tag{4}$$

Here, α is a hyperparameter of the ELU function, which was set to 1 in this study.

Batch normalization [28] was applied to the DNN model. During the training process, gradient vanishing or gradient explosion can be caused by an internal covariant shift. The internal covariant shift inside the model was reduced using batch normalization. As shown in (5) and (6), the mean (μ_B) and variance (σ_B) of the mini-batch are calculated for each feature with size m. Equation (7) presents the normalization for feature using the mean and variance of the mini-batch from (5) and (6). Finally, batch normalization ($BN_{\gamma,\beta}(x_i)$) is performed using a linear transform with a scale factor (γ) and shift factor (β) to adjust the sample distributions of the mini-batch.

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{5}$$

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_B)^2 \tag{6}$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{7}$$

$$y_i = \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i) \tag{8}$$

where x_i and y_i are ith term of input and output vectors, and ϵ is a constant to avoid zero in the denominator for numerical stability.

The Adam optimizer [29] was used to train the DNN model, which includes the concepts of stochastic gradient

and moment to perform a stable training process. In (9), the gradient at time step t (g_t) is obtained and applied to the first moment (m_t) by an exponential moving average of the gradient, as shown in (10). As presented in (11), the second raw moment (v_t) is derived from the squared gradient (g_t^2) and the moment of the previous epoch (m_{t-1}). Then, in (12), the parameter (θ_t) is updated using the parameter of the previous time step (θ_{t-1}), the first moment, and the second moment from (10) and (11).

$$g_t = \nabla_\theta f_t\left(\theta_{t-1}\right) \tag{9}$$

$$m_t = \beta_1 \cdot m_{t-1} + (1-\beta_1) \cdot g_t \tag{10}$$

$$v_t = \beta_2 \cdot m_{t-1} + (1-\beta_2) \cdot g_t^2 \tag{11}$$

$$\theta_t = \theta_{t-1} - \alpha \cdot m_t / (\sqrt{v_t + \epsilon}) \tag{12}$$

where g_t^2 denotes the elementwise square operation ($g_t \odot g_t$), $f(\theta)$ is a stochastic objective function, β_1 and β_2 are 0.9 and 0.999, respectively, which represent the exponential decay rates, α is the step size of one epoch that is 0.001, and ϵ is a model constant, 10^{-8}.

III. EXPERIMENTAL SETUP

The experimental setup described in this section is based on previous research [30].

The displacement volume of the engine used in this study was 2.151 L with a compression ratio of 16.0. Four WLTP cycles were performed, with the temperature variations listed in Table 3. Detailed information on the engine specifications and WLTP cycles can be found in previous publications [30].

TABLE 3. Temperature conditions of WLTP cycles.

	Minimum value	Maximum value
WLTP 1	35	85
WLTP 2	45	85
WLTP 3	20	85
WLTP 4	35	40

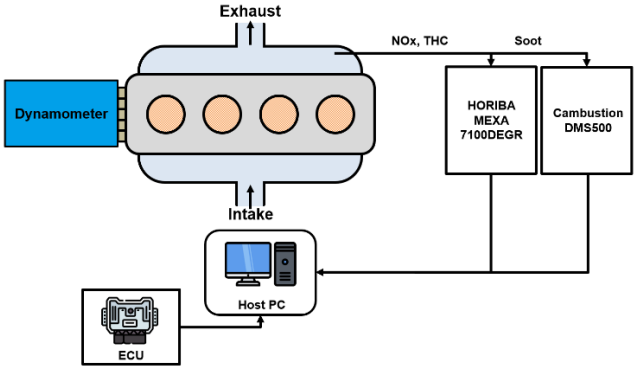


FIGURE 3. Experimental setup.

Fig. 3 presents the experimental setup. The engine was controlled by an engine control unit (ECU) connected to the host PC. The engine-out NO_x and THC were measured using a HORIBA MEXA 7100DEGR, and soot was measured in real-time by a Cambustion DMS500 during transient cycles.

A 340 kW alternating current dynamometer (AVL, Austria) was utilized to operate the engine system under transient conditions.

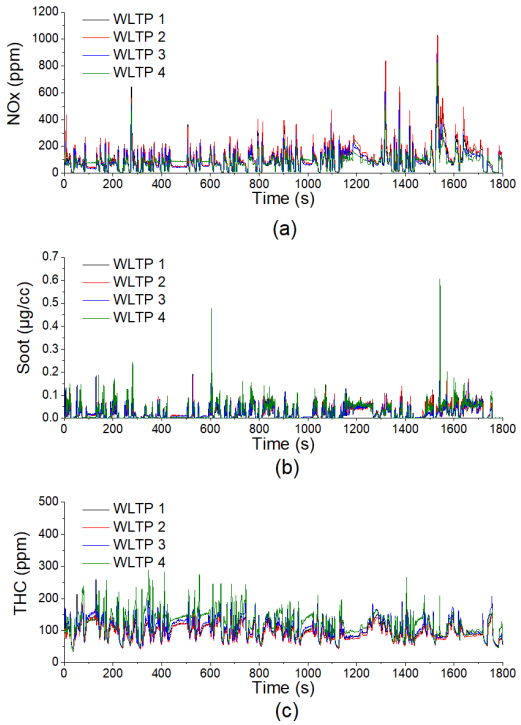


FIGURE 4. Measured emissions for 4 WLTP cycles: (a) NO_x, (b) soot, and (c) THC emissions.

TABLE 4. Mean and standard deviation of NO_x, Soot, and THC emissions for WLTP cycles.

		WLTP 1	WLTP 2	WLTP 3	WLTP 4
NO_x [ppm]	Mean	107.3	113.9	92.6	85.8
	STD	95.8	102.3	77.9	65.0
Soot [μg/cc]	Mean	0.0240	0.0239	0.0225	0.0263
	STD	0.0255	0.0254	0.0246	0.0344
THC [ppm]	Mean	100.0	96.4	106.0	121.4
	STD	27.2	27.0	30.9	37.8

Fig. 4 presents measured NO_x, soot, and THC emissions for WLTP 1-4, and Table 4 lists the mean and standard deviation (STD) of these emissions for the WLTP cycles. Mean and STD values of NO_x and THC emissions were relatively similar compared to those of soot emission. The level of the mean values of NO_x and THC emissions was of the order of 10^2, and the STD was of the order of 10^1. However, soot emission had a mean and STD of the order of 10^{-2}. These differences affected the similarity of weights

inside neural networks while the task transfer learning was performed, as described in 'Results and Discussion' section.

The duration of the WLTP cycles was 1800 s, and data were obtained every 0.1 s for the cycles. Therefore, each cycle consisted of 18001 data points, including the data at the time step when the measurement started (t = 0). Because the experiments were conducted for four WLTP cycles, the total number of data points was 72004. These data were randomly distributed to the training (60%, 43204), validation (20%, 14400), and test (20% 14400) sets to organize the deep-learning models and evaluate the effects of transfer learning.

Table 5 lists the input variables in this study. The input variables comprised 13 statuses of the engine measured by the ECU. The NO_x, soot, and THC emissions that were target outputs were respectively combined with these input variables as datasets.

TABLE 5. Input variables.

Intake air mass [mg]	EGR rate [%]
Engine speed [rpm]	Fuel quantity [mg]
Main injection quantity [mg]	Pilot injection quantity [mg]
Post injection quantity [mg]	Main injection timing [CA]
Injection pressure [bar]	Intake pressure [bar]
Lambda	Intake temperature [°C]
Coolant temperature [°C]	

The computing environment consisted of an Intel® Xeon® Gold 6230 @ 2.10 GHz central processing unit, 256 GB RAM, NVIDIA Geforce RTX 2080 Ti with 12 GB of a graphical processing unit, and Windows 10 OS. Python 3.7 was used as a programming language; Keras v.2.3.1 with the TensorFlow backend was used as the deep learning library in this study.

IV. RESULTS AND DISCUSSION

A. RESULTS OF INDIVIDUALLY TRAINED MODELS

As a conventional training process for the prediction of target emissions, that is, NO_x, soot, and THC, three individual models were optimized and trained in this study. Each NO_x, soot, and THC model was the reference model to evaluate the results of task transfer learning on the emission prediction, as well as the pretrained model of task transfer learning for other emissions.

Table 6 presents the optimized hyperparameters of the NO_x, soot, and THC models and their optimization times. The node arrangement was derived using the hidden-node determination logic [25] with the number of hidden layers and number of 1st hidden nodes. The first 13 nodes indicate the input dimension of the data, and the last one is the output dimension of the model. When applying task transfer learning, the transferred model adopts the hyperparameters of the pretrained model. For example, the soot model

TABLE 6. Optimized hyperparameters of NO_x, Soot, and THC models and their optimization time.

Hyperparameter	NOx Model 1	Soot Model 1	THC Model 1
Learning rate	$10^{-2.406}$	$10^{-4.458}$	$10^{-2.467}$
Learning rate decay	$10^{-5.913}$	$10^{-7.240}$	$10^{-6.146}$
Batch size	50901	55102	44063
Number of hidden layers	9	9	9
Number of the 1st hidden nodes	21	24	15
Node arrangement	13-21-29-37-45-38-31-24-17-10-1	13-24-35-46-57-48-39-30-21-12-1	13-15-17-19-21-18-15-12-9-6-1
Optimization time [s]	748183.9	881598.2	738262.5

TABLE 7. Accuracy results of NO_x, Soot, and THC models.

		Training	Validation	Test
NO_x Model 1	R^2	0.9920	0.9802	0.9780
	RMSE [ppm]	7.90	12.6	12.6
Soot Model 1	R^2	0.9575	0.9193	0.9215
	RMSE [μg/cc]	0.00574	0.00805	0.00765
THC Model 1	R^2	0.9743	0.9445	0.9390
	RMSE [ppm]	5.17	7.59	8.05

transferred from the pretrained NO_x model, Soot Model 2, utilizes the hyperparameters of NO_x Model 1 in Table 6. Specifically, NOx Model 1 (pretrained model), Soot Model 2, and THC Model 2 (transferred model) adopted the same hyperparameters. The transferred models from Soot Model 1 (NO_x Model 2 and THC Model 3) utilized the same set of hyperparameters as their pretrained model. NO_x Model 3, and Soot Model 3 had the same hyperparameters as THC Model 1 because they originated from THC Model 1.

The accuracy results of NO_x Model 1, Soot Model 1, and THC Model 1 are listed in Table 7. According to the test set accuracies, NO_x emissions were predicted more accurately than the other emissions. Overall, the R^2 values for the test set, which are the representative indices indicating the model's accuracy, were over 0.92. These levels could be recognized as accurate under transient conditions. A detailed analysis of the models is presented next section with the results of the transferred models.

B. RESULTS OF TASK TRANSFER LEARNING

Using the pretrained models provided in the previous section, task transfer learning was performed for all cases of NO_x, soot, and THC emissions. The last two hidden layers of the pretrained models were trained again to transfer to other emission models with the hyperparameters of the pretrained

models. Task transfer learning reduces computational cost and time by eliminating repetitive optimization processes for the organization of models.

TABLE 8. Transfer learning results for NO_x, Soot, and THC emissions.

		Training	Validation	Test
NO_x Model 1	R^2	0.9920	0.9802	0.9780
	RMSE [ppm]	7.9	12.6	12.6
	Time [s]	3483.4	0.245	0.256
NO_x Model 2	R^2	0.9876	0.9781	0.9757
	RMSE [ppm]	9.7	13.2	13.1
	Time [s]	82319.6	0.304	0.362
NO_x Model 3	R^2	0.9871	0.9780	0.9765
	RMSE [ppm]	10.0	13.2	12.9
	Time [s]	5064.3	0.271	0.281
Soot Model 1	R^2	0.9575	0.9193	0.9215
	RMSE [µg/cc]	0.00574	0.00805	0.00765
	Time [s]	152589.3	0.250	0.250
Soot Model 2	R^2	0.9686	0.9212	0.9195
	RMSE [µg/cc]	0.00499	0.00797	0.00780
	Time [s]	41077.5	0.280	0.270
Soot Model 3	R^2	0.9556	0.9109	0.9105
	RMSE [µg/cc]	0.00589	0.00850	0.00821
	Time [s]	17295.3	0.284	0.291
THC Model 1	R^2	0.9743	0.9445	0.9390
	RMSE [ppm]	5.17	7.59	8.05
	Time [s]	3921.1	0.261	0.263
THC Model 2	R^2	0.9484	0.9296	0.9288
	RMSE [ppm]	7.33	8.53	8.65
	Time [s]	4162.0	0.293	0.242
THC Model 3	R^2	0.9502	0.9305	0.9275
	RMSE [ppm]	7.20	8.47	8.74
	Time [s]	121924.0	0.280	0.270

Table 8 presents the accuracies of transfer learning for NO_x, soot, and THC emissions, respectively. By comparing the test set accuracies, the pretrained models were found to be more accurate than the transferred models. However, the R^2 gaps between the models were within 0.012, with a value of 0.0015 for the NO_x emission, 0.011 for the soot emission, and 0.0115 for the THC emission. In the case of the prediction of the transient emissions, the difference in the R^2 value of 0.0015 (0.15% compared to the R^2 value of NO_x Model 1) can be recognized as a similar accuracy level. In the cases of soot and THC emissions, the accuracy differences were acceptable considering the optimization and training times. The times for hyperparameter optimization applied to the Soot Model 1 and THC Model 1 were 881598.2 s (10.2 days) and 738262.5 s (8.5 days), respectively, as presented in Table 6, and they required additional training time to obtain results, as listed in Table 8. However, the transferred models did not require optimization processes because they utilized the hyperparameters of the pretrained model. They required only training time, with a minimum of 4162.0 s (1.2 hours) on THC Model 2 and a maximum of 121924.0 s (1.4 days) on THC Model 3. Therefore, with an accuracy reduction in R^2 values of only 0.011 and 0.015, they reduced the computation time by several days.

The accuracy of the pretrained model affected the accuracy of the transferred model. Considering the same task, the accuracy of the transferred model was relatively higher when the accuracy of the pretrained model was high. The accuracies of the pretrained models were in the order of NO_x Model > THC Model 1 > Soot Model 1. As an example of the transferred NO_x models, NO_x Model 3 was more accurate than NO_x Model 2 because of the accuracy of THC Model 1, which was pretrained model of NO_x Model 3, was higher than the accuracy of Soot Model 1, which was the pretrained model of NO_x Model 2. The transferred soot and THC models presented the same tendency as the NO_x case.

There was a noticeable tendency in the training times of the transferred models. Compared to the models transferred from NO_x or THC pretrained models, it took more time to train NO_x Model 2 and THC Model 3, which were transferred using the pretrained soot model. Because the learning rate of the soot pretrained model (Soot Model 1) was much smaller than those of NO_x Model 1 and THC Model 1, more time was required to train the transferred models from the pretrained soot model. In terms of training time, it is recommended to utilize NO_x or THC emissions instead of soot emissions to organize the pretrained model if the option is available.

Table 9 provides the statistical results for the NO_x, soot, and THC models for WLTP cycles, and figures from Fig. 9 to Fig. 10 in Appendix present the emission profiles under WLTP cycles by comparing the measured data and model results. The accuracies for WLTP cycles calculated using the mean absolute error (MAE) showed similar levels in Models 1 – 3 for all emissions. In particular, the transferred models (Models 2 and 3) exhibited similar accuracy.

The MAEs relative to the maximum emission values were approximately 0.57-0.82% for NO_x emissions, 0.69-2.02% for soot emissions, and 1.52-2.42% for THC emissions. The linearity of the HORIBA MEXA 7100 DEGR, which is a measurement device for NO_x and THC emissions, was 1% of the maximum measurement range (5000 ppm) or 2% of the reading scale. The NO_x emission results were better than the linearity of the measurement device. The error in the soot and THC emissions were similar to those of the device. The measurement devices used in this study were accurate equipment used in the laboratory, and they were much more accurate than sensors for practical use, especially in vehicles. Therefore, the results of all models in Table 9 are comparable to those of the measurement devices and better than the accuracies of the commercial sensors for the vehicles.

TABLE 9. Statistical results of NO_x, Soot, THC models for WLTP cycles.

NO_x		WLTP 1	WLTP 2	WLTP 3	WLTP 4
Maximum NO_x of measured data [ppm]		1028.9	1025.3	891.9	828.9
MAE [ppm] (% of maximum value)	NO_x Model 1	6.90 (0.67)	7.21 (0.70)	5.90 (0.66)	4.69 (0.57)
	NO_x Model 2	8.06 (0.78)	8.33 (0.81)	6.75 (0.76)	5.38 (0.65)
	NO_x Model 3	8.03 (0.78)	8.38 (0.82)	6.75 (0.76)	5.30 (0.64)
Soot		WLTP 1	WLTP 2	WLTP 3	WLTP 4
Maximum Soot of measured data [μg/cc]		0.195	0.186	0.190	0.606
MAE [μg/cc] (% of maximum value)	Soot Model 1	0.00332 (1.70)	0.00354 (1.90)	0.00349 (1.84)	0.00427 (0.70)
	Soot Model 2	0.00350 (1.79)	0.00366 (1.97)	0.00344 (1.81)	0.00423 (0.70)
	Soot Model 3	0.00354 (1.82)	0.00376 (2.02)	0.00368 (1.94)	0.00420 (0.69)
THC		WLTP 1	WLTP 2	WLTP 3	WLTP 4
Maximum THC of measured data [ppm]		258.8	208.8	258.9	289.8
MAE [ppm] (% of maximum value)	THC Model 1	3.93 (1.52)	3.95 (1.89)	3.99 (1.54)	5.14 (1.77)
	THC Model 2	5.38 (2.08)	5.05 (2.42)	5.24 (2.02)	6.51 (2.25)
	THC Model 3	5.12 (1.98)	4.98 (2.39)	5.06 (1.95)	6.53 (2.25)

Task transfer learning in this study was performed by retraining the last two hidden layers of the pretrained model while the weights of the other hidden layers were maintained. Representatively, weights in the last two layers of NO_x Model 1, Soot Model 2, and THC Model 2 were analyzed. Soot Model 2 and THC Model 2 were transferred from NO_x Model 1, and they had the same configuration of hidden layers and nodes.

Before the analysis, notations were defined to indicate the location of the weights inside the model. As shown in Table 7, NO_x Model 1 had nine hidden layers; therefore, the nodes of the 8^{th} and 9^{th} layers were trained again to transfer the model to Soot Model 2 and THC Model 2. Fig. 5 presents the configuration of the last part of NO_x Model 1 for transfer learning. Only the connections from node 1 of the 8^{th} layer are shown in the figure, and other connections are omitted for clarity. Weights from the 8^{th} layer to the 9^{th} layer are marked as w_{i-j}, where i indicates the node of the 8^{th} layer and j is the node of the 9^{th} layer. Likewise, weights from the 9^{th} layer to the output layer are denoted as w_k because there is only one hidden node in the output layer.

Fig. 6 presents weights inside the models from the 8^{th} layer to the 9^{th} layer. The patterns and colors of NO_x Model 1 and THC Model 2 are similar, which implies that the value ranges of weights are not significantly different. However, the colors of Soot Model 2 are more vivid than those of NO_x Model 1 and THC Model 2. The weights of the NO_x Model 1 were,

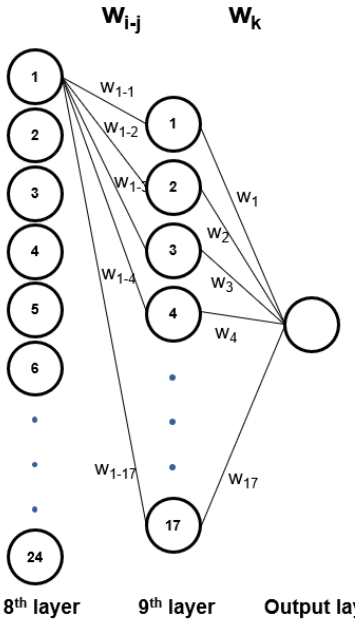


FIGURE 5. Weight Notations in the 8^{th} layer, the 9^{th} layer, and output layers of NO_x Model 1, Soot Model 2, and THC Model 2.

thus, remarkably changed during the retraining process to Soot Model 2 compared to transferring to THC Model 2. This tendency can also be observed in the weights of the last layer, from the 9^{th} layer to the output layer, in Fig. 7. The weight values in the last layer of NO_x Model 1 and THC Model 2 were slightly different while the weights of Soot Model 2 were much smaller than those of the other models.

From Fig. 6 and Fig. 7, the retraining process for the task transfer learning from NO_x Model 1 to THC Model 2 can be considered as fine-tuning to assign a new task to the pretrained model. However, the transfer from NO_x Model 1 to Soot Model 2 involved a larger change in weight values. This difference was caused by the data distribution of NO_x, soot, and THC emissions. The statistics for the emissions presented in Table 4 show that the mean values and STD values for both the NO_x and THC emissions were of the order of 10^2 and 10^1, respectively. However, for the soot emissions, the values for the mean and STD were of the order of 10^{-2}. Therefore, the weights of NO_x Model 1 should be more drastically changed to transfer the model to Soot Model 2 to reflect the data distribution.

Weight similarities between the models were also observed in cases of transferred models from the pretrained soot and THC models, as presented in Fig. 8. The numbers of last hidden nodes of Soot Model 1 and THC Model 1 were 12 and 6, respectively; therefore, the number of k nodes indicated on the x-axis of the figures was different according to the pretrained model. As shown in Fig. 8 (a), the pretrained soot model (Soot Model 1) was transferred to NO_x Model 2 and THC Model 3, and the weights of the models exhibited similar tendencies according to the data distribution. The pretrained THC model (THC Model 1) was fine-tuned to

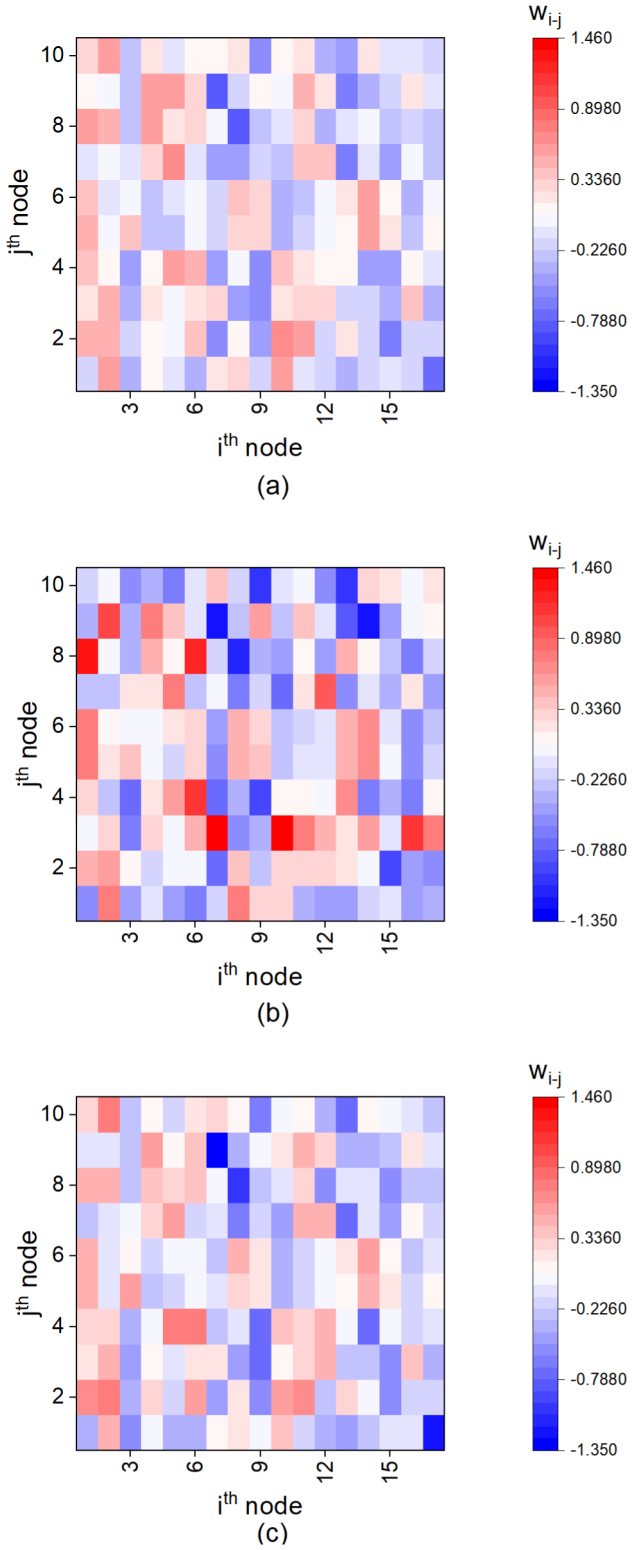


FIGURE 6. Weights from the 8^{th} layer to the 9^{th} layer: (a) NO_x Model 1, (b) Soot Model 2, and (c) THC Model 2.

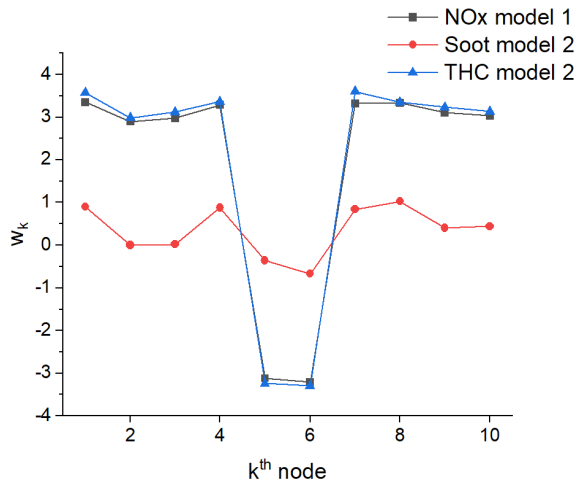


FIGURE 7. Weights of the last layer, from the 9^{th} layer to output layer.

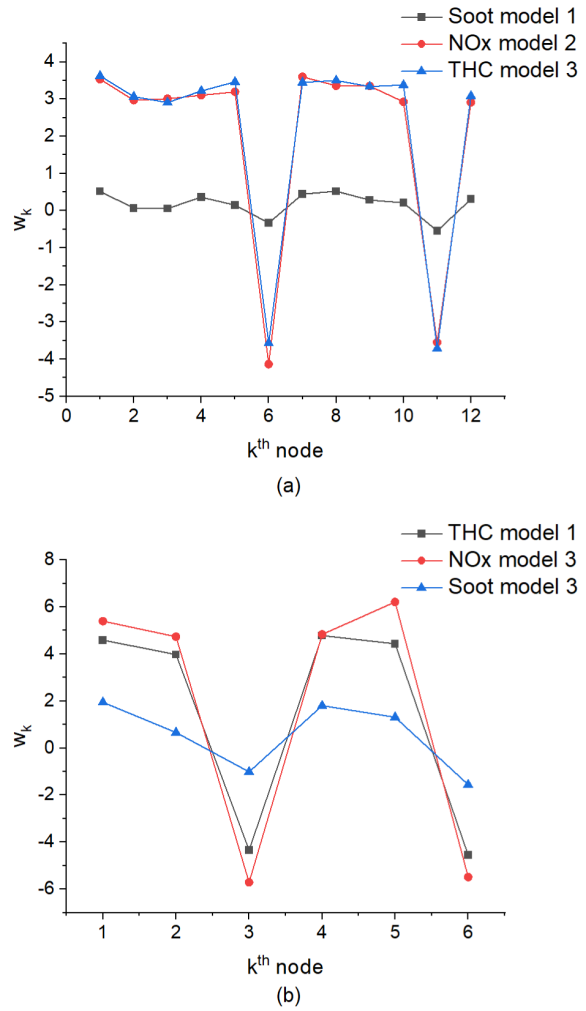


FIGURE 8. Weights of the last layer: (a) transferred models from Soot Model 1, and (b) transferred models from THC Model 1.

transfer to NO_x Model 3, and the weights of THC Model 1 were further changed to convert into Soot Model 3, as shown in Fig. 8 (b).

V. CONCLUSION

In this study, task transfer learning mainly utilized in image and natural language processing areas was introduced to

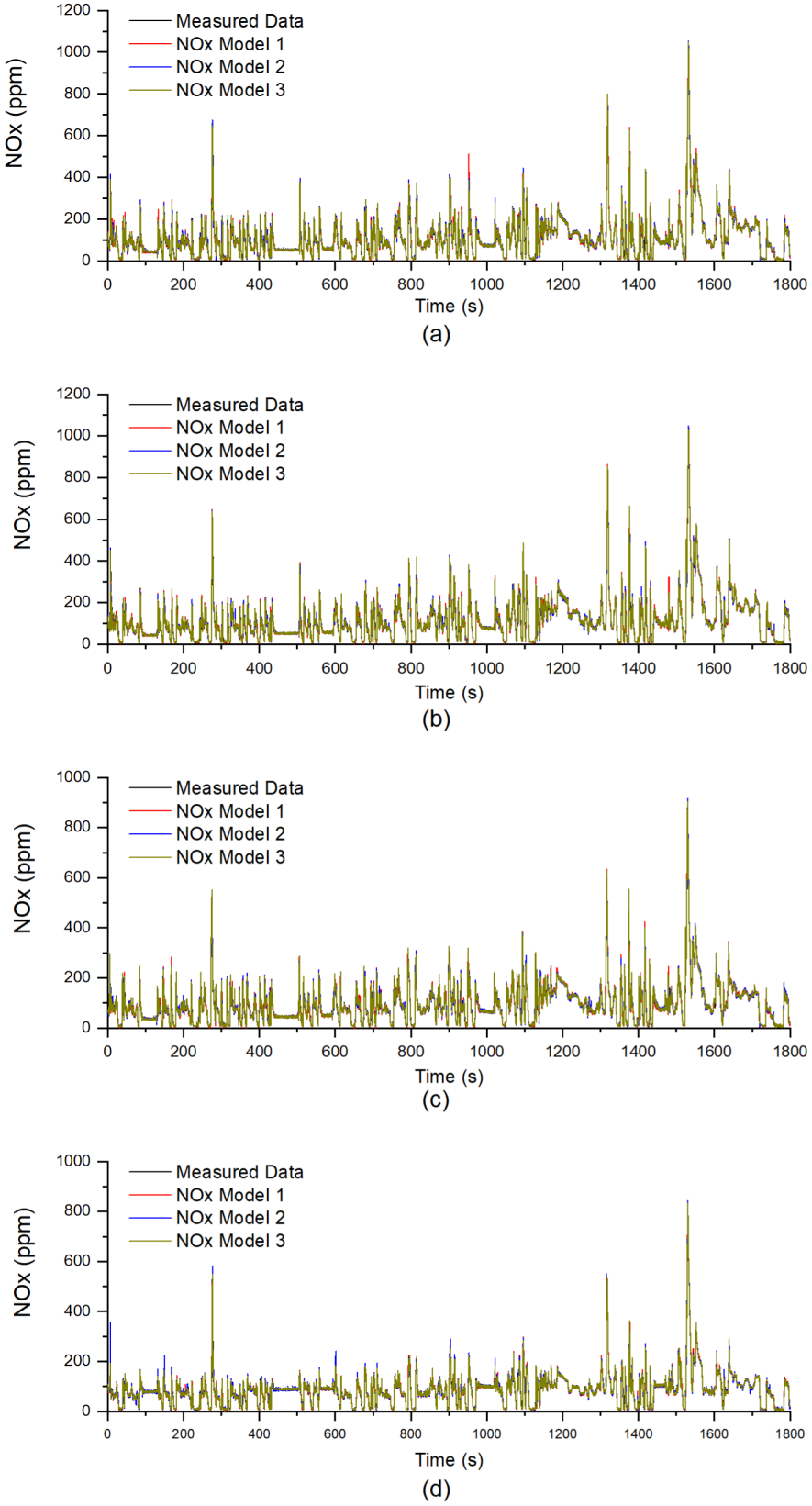


FIGURE 9. NO_x emission profiles of measured data, and NO_x Models 1-3: (a) WLTP 1, (b) WLTP 2, (c) WLTP 3, and (d) WLTP 4.

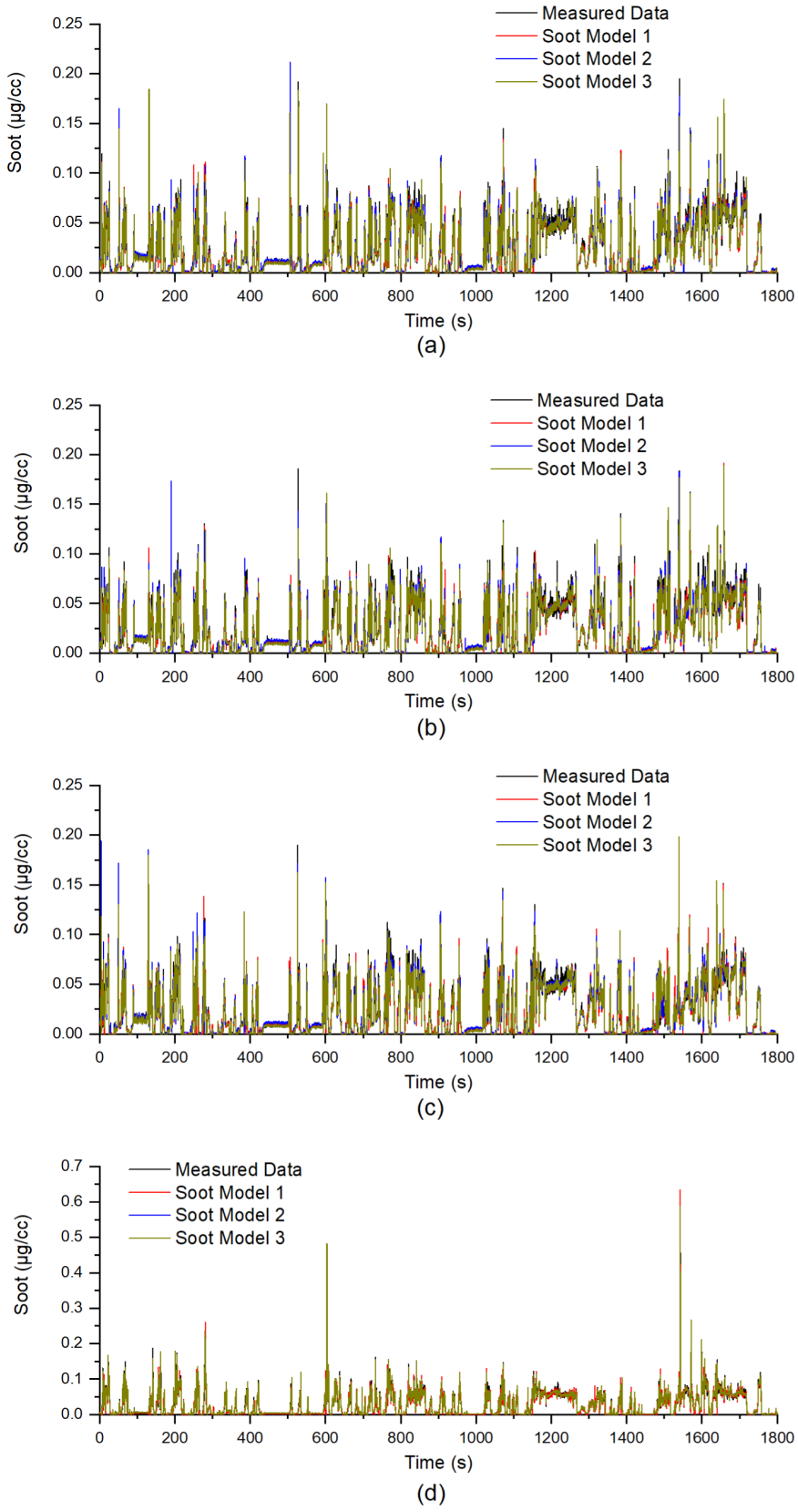


FIGURE 10. Soot emission profiles of measured data, and Soot Models 1-3: (a) WLTP 1, (b) WLTP 2, (c) WLTP 3, and (d) WLTP 4.

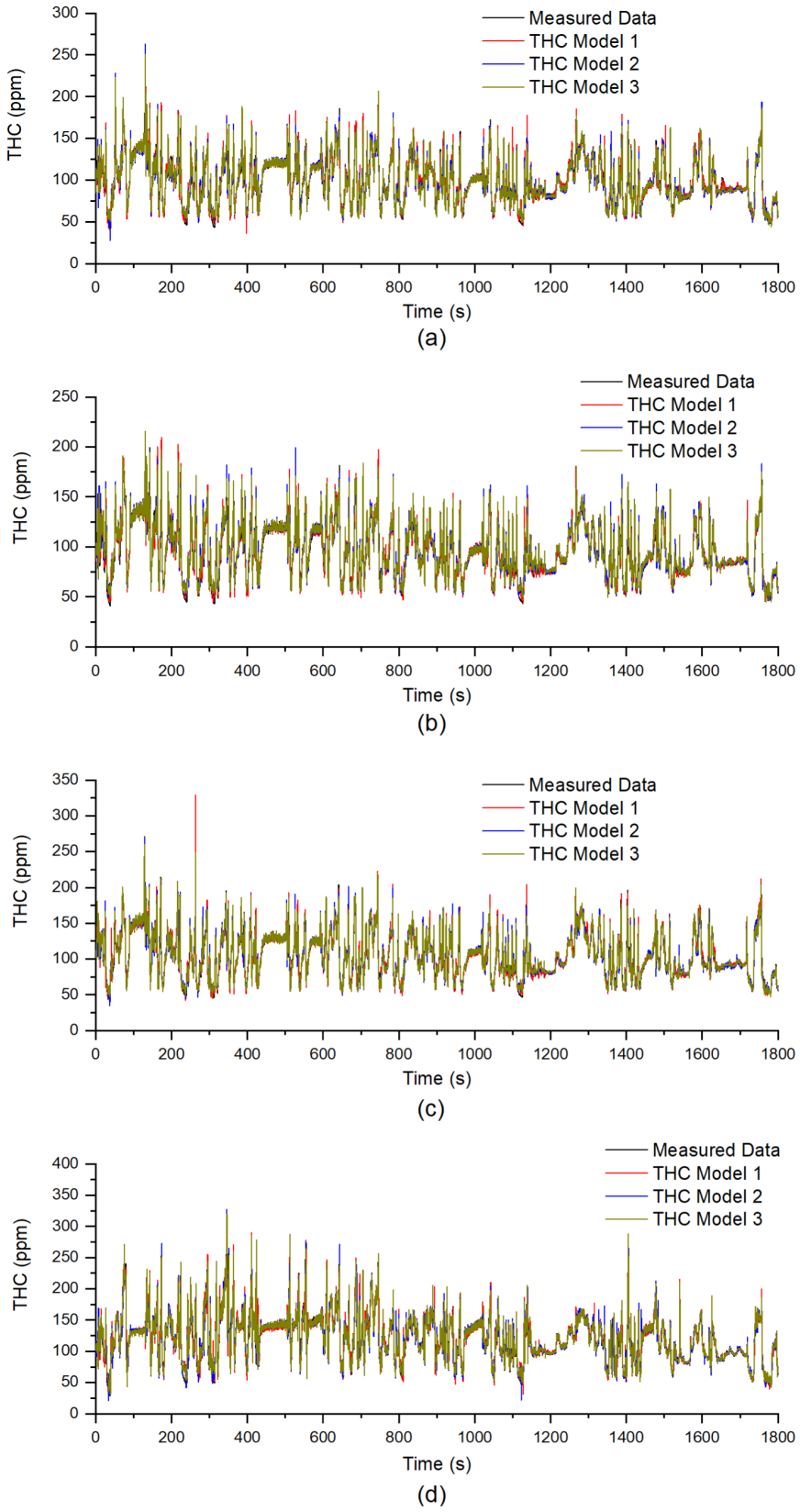


FIGURE 11. THC emission profiles of measured data, and THC Models 1-3: (a) WLTP 1, (b) WLTP 2, (c) WLTP 3, and (d) WLTP 4.

predict the emissions of internal combustion engines. The target emissions were the transient NO_x, soot, and THC emissions. All possible transfer cases between the three emissions were investigated. WLTP cycles were introduced to evaluate the accuracy of the model under transient conditions.

1) Task transfer learning was performed by training the last two hidden layers of the pretrained model for the target emission, while the other layers were frozen during training. The hyperparameters of the pretrained model were utilized for task transfer learning.
2) The accuracies of the individually trained and transferred models were compared to evaluate the validity of task transfer learning. The R^2 values were 0.9765-0.9780 for the NO_x models, 0.9105-0.9215 for the soot models, and 0.9275-0.9390 for the THC models. The R^2 gaps between pretrained and transferred models were 0.0015 for the NOx emission, 0.011 for the soot emission, and 0.0115 for the THC emission. These R^2 values below 0.012 could be considered as similar-level in accuracy. The relative MAEs of models applying WLTP cycles were comparable to those of the emission measurement device, which had better accuracy than commercial sensors for vehicle installation.
3) It took approximately 8-10 days to individually optimize the models for each emission. Using transfer learning, it was not necessary to repeat this optimization process to predict different emissions from the pretrained emission. From this point of view, the small accuracy reduction caused by task transfer learning can be considered acceptable because of the significantly higher efficiency achieved by the process of the model organization.

This study proved the validity of task transfer learning for predicting phenomena of an internal combustion engine. Through task transfer learning, deep learning models were effectively transferred to predict another task. In this study, the major emissions of diesel engines under transient conditions were utilized to verify the task transfer learning.

Task transfer learning had been validated for outputs corresponding to similar categories of emission in this study. Subsequently, it would be meaningful for this study to progress by applying task transfer learning to phenomena related to different categories such as fuel efficiency, performance, and emissions.

APPENDIX

See Figs. 9, 10, and 11.

REFERENCES

[1] M. Williams and R. Minjares, *A Technical Summary of Euro 6/VI Vehicle Emission Standards*. Washington, DC, USA: International Council on Clean Transportation, Tech. Briefing, Jul. 2016.

[2] A. Atmanli, E. Ileri, and N. Yilmaz, "Optimization of diesel–butanol–vegetable oil blend ratios based on engine operating parameters," *Energy*, vol. 96, pp. 569–580, Feb. 2016.

[3] P. Dimitriou, Z. Peng, D. Lemon, B. Gao, and M. Soumelidis, "Diesel engine combustion optimization for bio-diesel blends using Taguchi and ANOVA statistical methods," SAE Tech. Paper, 2013-24-0011, Sep. 2013.

[4] N. Yilmaz, E. Ileri, A. Atmanli, A. D. Karaoglan, U. Okkan, and M. S. Kocak, "Predicting the engine performance and exhaust emissions of a diesel engine fueled with hazelnut oil methyl ester: The performance comparison of response surface methodology and LSSVM," *J. Energy Resour. Technol.*, vol. 138, no. 5, Sep. 2016, Art. no. 052206.

[5] E. Akleman, "Deep learning," *Computer*, vol. 53, no. 9, p. 17, Sep. 2020.

[6] J. Kim, J. Park, S. Shin, Y. Lee, K. Min, S. Lee, and M. Kim, "Prediction of engine NO_x for virtual sensor using deep neural network and genetic algorithm," *Oil Gas Sci. Technol. Revue d'IFP Energies Nouvelles*, vol. 76, p. 72, Nov. 2021.

[7] J. Seo, B. Yun, J. Kim, M. Shin, and S. Park, "Development of a cold-start emission model for diesel vehicles using an artificial neural network trained with real-world driving data," *Sci. Total Environ.*, vol. 806, Feb. 2022, Art. no. 151347.

[8] C. M. A. L. Cornec, N. Molden, M. van Reeuwijk, and M. E. J. Stettler, "Modelling of instantaneous emissions from diesel vehicles with a special focus on NO_x: Insights from machine learning techniques," *Sci. Total Environ.*, vol. 737, Oct. 2020, Art. no. 139625.

[9] Y. Wang, Y. Yu, and J. Li, "Predicting the transient NO_x emissions of the diesel vehicle based on LSTM neural networks," in *Proc. IEEE Conf. Telecommun., Opt. Comput. Sci. (TOCS)*, Dec. 2020, pp. 261–264.

[10] S. Shin, Y. Lee, J. Park, M. Kim, S. Lee, and K. Min, "Predicting transient diesel engine NO_x emissions using time-series data preprocessing with deep-learning models," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 235, no. 12, pp. 3170–3184, Oct. 2021.

[11] M. Bellone, E. Faghani, and Y. Karayiannidis, "Comparison of CNN and LSTM for modeling virtual sensors in an engine," *SAE Int. J. Adv. Current Practices Mobility*, vol. 2, no. 5, pp. 2632–2639, Apr. 2020.

[12] T. Van Hung, H. H. Alkhamis, A. F. Alrefaei, Y. Sohret, and K. Brindhadevi, "Prediction of emission characteristics of a diesel engine using experimental and artificial neural networks," *Appl. Nanosci.*, vol. 13, no. 1, pp. 433–442, Jan. 2023.

[13] S. H. Hosseini, A. Taghizadeh-Alisaraei, B. Ghobadian, and A. Abbaszadeh-Mayvan, "Artificial neural network modeling of performance, emission, and vibration of a CI engine using alumina nano-catalyst added to diesel–biodiesel blends," *Renew. Energy*, vol. 149, pp. 951–961, Apr. 2020.

[14] S. Shin, S. Lee, M. Kim, J. Park, and K. Min, "Deep learning procedure for knock, performance and emission prediction at steady-state condition of a gasoline engine," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 234, no. 14, pp. 3347–3361, Dec. 2020.

[15] Y. Bao, Y. Li, S. Huang, L. Zhang, L. Zheng, A. Zamir, and L. Guibas, "An information-theoretic approach to transferability in task transfer learning," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 2309–2313.

[16] A. R. Zamir, A. Sax, W. Shen, L. Guibas, J. Malik, and S. Savarese, "Taskonomy: Disentangling task transfer learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3712–3722.

[17] K. Dwivedi and G. Roig, "Representation similarity analysis for efficient task taxonomy & transfer learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12379–12388.

[18] R. K. Samala, H.-P. Chan, L. M. Hadjiiski, M. A. Helvie, K. H. Cha, and C. D. Richter, "Multi-task transfer learning deep convolutional neural network: Application to computer-aided diagnosis of breast cancer on mammograms," *Phys. Med. Biol.*, vol. 62, no. 23, pp. 8894–8908, Nov. 2017.

[19] Y. Tsai, B. Huang, Y. Guo, and G. Yang, "Transfer learning for surgical task segmentation," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9166–9172.

[20] S. Bunrit, N. Kerdprasop, and K. Kerdprasop, "Evaluating on the transfer learning of CNN architectures to a construction material image classification task," *Int. J. Mach. Learn. Comput.*, vol. 9, no. 2, pp. 201–207, Apr. 2019.

[21] J. Pilault, A. Elhattami, and C. Pal, "Conditionally adaptive multi-task learning: Improving transfer learning in NLP using fewer parameters & less data," Sep. 2020, *arXiv:2009.09139*.

[22] S. Wang, W. Li, S. M. Siniscalchi, and C. Lee, "A cross-task transfer learning approach to adapting deep speech enhancement models to unseen background noise using paired senone classifiers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6219–6223.

[23] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[24] T. Evgeniou and M. Pontil, "Regularized multi-task learning," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2004, pp. 109–117.
[25] S. Shin, Y. Lee, M. Kim, J. Park, S. Lee, and K. Min, "Deep neural network model with Bayesian hyperparameter optimization for prediction of NO_x at transient conditions in a diesel engine," *Eng. Appl. Artif. Intell.*, vol. 94, Sep. 2020, Art. no. 103761.
[26] Keras. (May 2, 2023). *EarlyStopping*. [Online]. Available: keras.io/api/callbacks/early_stopping
[27] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," Nov. 2015, *arXiv:1511.07289*.
[28] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, Lille, France, vol. 37, 2015, pp. 448–456.
[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Jan. 2017, *arXiv:1412.6980*.
[30] S. Shin, Y. Lee, Y. Lee, J. Park, M. Kim, S. Lee, and K. Min, "Designing a steady-state experimental dataset for predicting transient NO_x emissions of diesel engines via deep learning," *Expert Syst. Appl.*, vol. 198, Jul. 2022, Art. no. 116919.

SEUNGHYUP SHIN received the B.S., M.S., and Ph.D. degrees from the Department of Mechanical and Aerospace Engineering, Seoul National University, in 2009, 2011, and 2021, respectively.

From 2011 to 2015, he was a Research Assistant with Hyundai Doosan Infracore Company Ltd., Incheon, South Korea. After the Ph.D. degree, he was with the Department of Railroad Artificial Intelligence, Korea Railway Research Institute, Uiwang, South Korea, from 2022 to 2023. He is currently an Assistant Professor with the Department of Artificial Intelligence, Sejong University, Seoul, South Korea. His current research interest includes convergence research between artificial intelligence and mechanical engineering systems.

MINJAE KIM was born in South Korea. He received the B.S. and M.S. degrees from the Department of Electrical Engineering, Seoul National University, in 2008 and 2010, respectively, and the Ph.D. degree from Seoul National University, in 2014. He is currently a Professor with the School of Mechanical Engineering, Myongji University.

JIHWAN PARK was born in South Korea. He received the B.S. degree from the Department of Mechanical Engineering, Seoul National University, in 2018, where he is currently pursuing the Ph.D. degree, specializing in applying deep learning in the field of engines.

SANGYUL LEE was born in South Korea. He received the B.S. and M.S. degrees from the Department of Mechanical Engineering, Seoul National University, in 2006 and 2008, respectively, and the Ph.D. degree from Seoul National University, in 2013. He is currently a Professor with the Division of Mechanical and Electronics Engineering, Hansung University.

KYOUNGDOUG MIN (Member, IEEE) was born in South Korea. He received the B.S. and M.S. degrees from the Department of Mechanical Engineering, Seoul National University, in 1986 and 1988, respectively, and the Ph.D. degree from the Massachusetts Institute of Technology, in 1994. He is currently a Professor of mechanical engineering with Seoul National University.