

Analysis of Electrochemical Impedance Data: Use of Deep Neural Networks

Dulyawat Doonyapisut, Padmanathan-Karthick Kannan, Byeongkyu Kim, Jung Kyu Kim, Eunseok Lee, and Chan-Hwa Chung*

Technology advancements in energy storage, photocatalysis, and sensors have generated enormous impedimetric data. Electrochemical impedance spectroscopy (EIS) results play an essential role in analyzing the interfacial properties of materials. Nonetheless, in many situations, the data is misinterpreted due to the complexity of the electrochemical system or the compromise between the experimental result and the theoretical model, resulting in partiality in the interpretation process, especially for the impedimetric results. Typically, the experimenter interprets impedimetric results using a searching approach based on a theoretical model until the best-fitting model is obtained, which is a time-consuming process, and errors can occur. To reduce misinterpretation by the experimenter, herein, the machine-learning strategy is demonstrated for the classification of an EIS circuit model and parameter prediction using a deep neural network (DNN). The DNN model shows a highly accurate classifier for the commonly used EIS circuit with an average area under the receiver operating characteristic curve of more than 0.95. Additionally, the model demonstrates high accuracy in the prediction of EIS parameters on a complex EIS system, with a maximum R^2 of 0.999. These reveal that the machine-learning strategy may open a new room for studying electrochemical systems.

1. Introduction


Electrochemical impedance spectroscopy (EIS) is a powerful technique for studying electrode materials and is also frequently employed to understand the complicated phenomena on the electrode surface. EIS analysis is necessary for novel research in various electrochemical systems such as CO₂ reduction,^[1–3] batteries,^[4–6] and sensors.^[7–10] The interpretation of impedance and frequency responses in the EIS technique allows for understanding the chemical and physical information of the interaction between electrode and electrolyte.^[8,11] The interpretation process can be done using a searching approach based on the theoretical basis that involves the following sequences: starting with the examination of the EIS spectrum shape or patterns, followed by the selection of the well-matched patterns with the equivalent circuit, and then the fitting of experimental data with the theoretical model using the combinations of several parameters that closely match the experiment. This conventional process is repeated until the theoretical model and experiment data reach an acceptable agreement in the level of confidence, the goodness of fit, and the theoretical condition. In the case of theoretical conditions and the fitted model failing to achieve an acceptable agreement, the simplified theoretical condition may be selected as an alternate model. However, the mentioned approach has several significant drawbacks, such as being time-consuming, lacking a statistical background for evaluation of fitting, leading to significant confusion regarding the conclusion, and the interpretation result depends on the interpreter. To appropriately interpret EIS data, the interpreter must have a thorough understanding of the technique and theoretical background.

In the history of electrochemistry, there have been multiple significant failures caused by highly experienced researchers misinterpreting data. For example, even though the actual experiment involved a complex system, the Randles circuit was widely cited as an equivalent circuit in the literature. Given the drawbacks of conventional data interpretation, analyzing EIS data without humans involved based on the theoretical and statistical background must be an important goal in novel electrochemistry. In some cases, high-end equipment with very high computational power may be able to execute a massive number of

D. Doonyapisut, B. Kim, J. K. Kim, C.-H. Chung
School of Chemical Engineering
Sungkyunkwan University
Suwon 16419, Republic of Korea
E-mail: chchung@skku.edu

P.-K. Kannan
Department of Chemistry
Sungkyunkwan University
Suwon 16419, Republic of Korea

E. Lee
College of Computing and Informatics
Sungkyunkwan University
Suwon 16419, Republic of Korea

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/aisy.202300085>.

© 2023 The Authors. Advanced Intelligent Systems published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: 10.1002/aisy.202300085

simulations with many free parameters for interpreting experimental data. However, it is still computationally intensive and expensive. These drawbacks motivated the development of a new approach to narrow down the search range of the model choices and the parameters. The ideal approach should be able to analyze the raw data without human assistance, the interpretation agreement should be reached with a high level of confidence, and the outcome should be done automatically.

In recent years, technology has developed rapidly. Deep neural network (DNN) and various architectures of machine learning (ML) have been successfully used in many scientific areas such as physics,^[12,13] astronomy,^[14–16] chemistry,^[17–20] biology,^[21–24] and electrochemistry.^[25–28] For instance, Gareth et al. demonstrated identifying reaction mechanisms in the electrochemical system using DNN based on images of cyclic voltammograms and showed high performance in classification tasks.^[28] This study indicates that DNN is very useful not only in common image recognition but it is also able to recognize the spectrum in the scientific area with a high degree of confidence. However, even though ML approaches have been adopted in those research areas, there are still relatively few reports on ML in electrochemical systems compared to the other research fields. In this work, we present an automatic EIS equivalent circuit classification using ML with DNN, which assesses the confidence in each circuit model. By studying the essential aspects of the EIS-spectrum data and identifying the equivalent circuit based on the simulation dataset, the strategy followed here is to emulate what an experienced electrochemist performs. Here, the EIS spectrum is unsuitable to present as an image since both pattern and value reflect the circuit model and the EIS parameters, making it hard to normalize and convert into an image. The DNN in the time series to frequency domain approach is well suited to this task. So far, a few activities underlying ML have been reported in the electrochemical research work. However, a comprehensive review of the literature reveals a notable lack of activity in impedimetric research, which may be due to the complexity of the EIS experimental data. A study in 2019 reported various ML models, including support vector machine, neural network, AdaBoost, and Gaussian process for EIS classification on commercial

batteries.^[29] However, the accuracy score was relatively low, especially for the neural network. It was lower than 0.40, which is hardly used for practical application. In 2022, some studies showed improvement in the EIS circuit prediction, which utilizes AdaBoost, random forest, and multilayer perceptron (MLP).^[30,31] Intriguingly, MLP shows a high accuracy of 0.75–0.95, while AdaBoost and random forest provide an accuracy of only 0.45 and 0.58, respectively. Here, the recent report shows that a neural network has a high potential for the EIS interpretation application.

This work demonstrates what can be accomplished by utilizing DNN and ML in an impedimetric interpretation. We explore five models of widely used EIS equivalent circuits to demonstrate how DNN can evaluate the classification and prediction of EIS parameters in the circuit models under the account and provide interpretation results automatically. Our goal is to demonstrate the DNN model with fully repeatable and straightforward integration of raw experimental data with a ranked list of possible circuit models and predicted EIS parameters. In addition, the impact of the signal-to-noise ratio (SNR) has also been explored to imitate EIS experimental data.

2. Results and Discussion

2.1. EIS Dataset

The EIS spectrum was simulated for five commonly used EIS equivalent circuit models, as shown in **Figure 1** (denoted as C1–C5), to cover various electrochemical applications. Circuit C1 is the simplest circuit, called the simplified Randles circuit (cf. Figure 1a).

Here, the spectra of C1 were simulated using Equation (S6), Supporting Information. Circuit C1 commonly shows a single semicircle arc in the spectrum, as noticed in **Figure 2a**. Next, C2 spectra were simulated using Equation (S7), Supporting Information. Generally, C2 provides two semicircle arcs in the spectrum, as plotted in Figure 2b (orange line). In addition to that the blue line may be present in the case of $Q_1 \approx Q_2$ and $R_1 \approx R_2$, as shown in Figure 2b. In Figure 1c, spectra of circuit

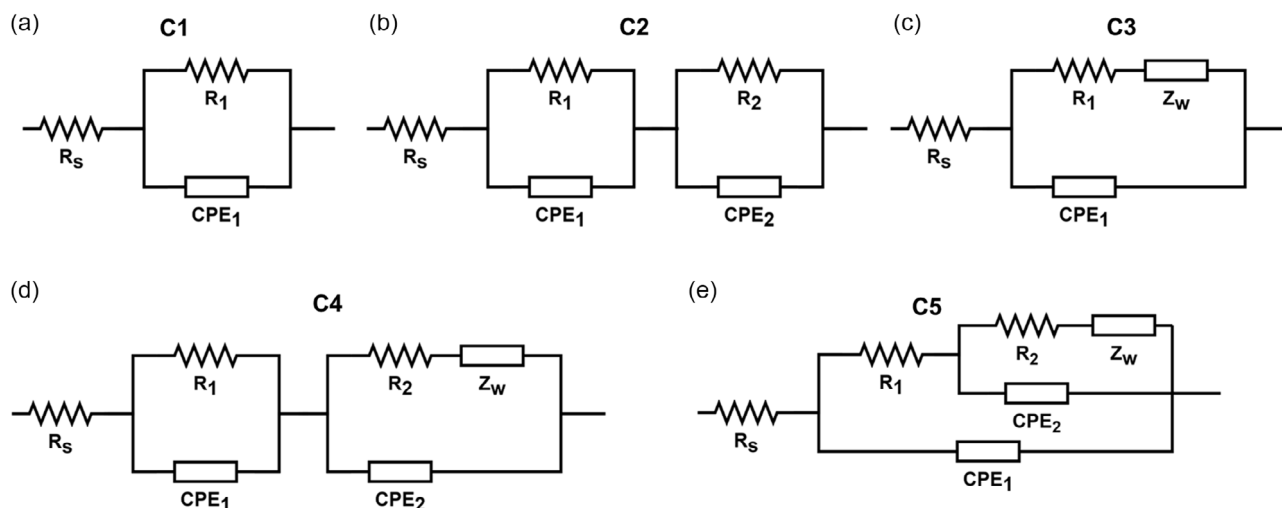


Figure 1. Five different equivalent circuit models, C1 to C5, are illustrated in a–e), respectively.

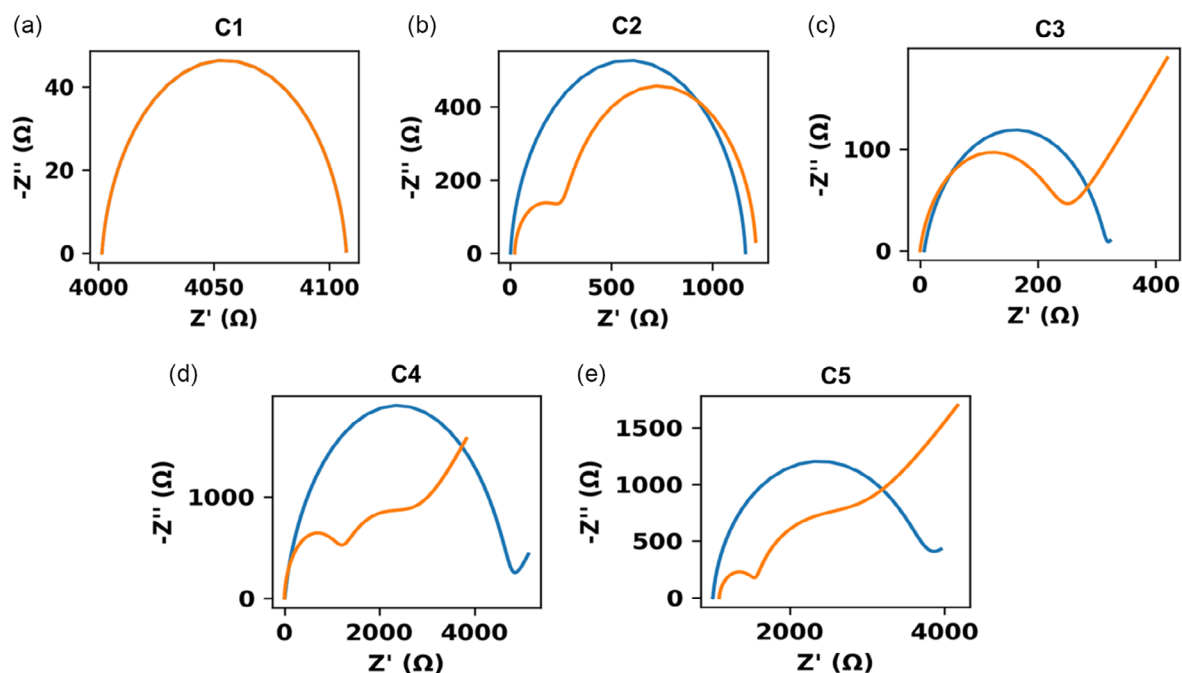


Figure 2. The examples of the electrochemical impedance spectroscopy (EIS) spectra from five equivalent circuits, C1 to C5, are plotted in a–e), respectively. The orange and blue lines present possible EIS spectrum patterns for the respective circuits.

C3 were generated by Equation (S8), Supporting Information. The Nyquist plot (one semicircle arc with a 45° straight line) for C3 is shown in Figure 2c (orange line). In contrast, the spectrum with only one semicircle arc may be observed (blue line), which in the case of R_1 is much greater than σ . Circuit C4 is the extent of C3, as described by Equation (S9), Supporting Information. As shown in Figure 2d, C4 exhibits the signature of two semicircle arcs with one 45° straight line. This model is usually found in the half-cell of a battery system.^[32,33]

Circuit C5 is a modification of C1, which replaces the R_1 with an infinite Warburg–Randles circuit, as shown in Equation (S10), Supporting Information. As a result, both signatures of C3 and C4 are presented in the EIS patterns of C5, as shown in Figure 2e. Because C4 and C5 were influenced by many parameters, the spectrum could be in one or two semicircle arcs, as shown by the blue and orange lines (Figure 2d,e). Since this work aims to demonstrate the capabilities of DNN for the classification of EIS equivalent circuits, the EIS data parameters are adjusted across a wide range to provide sufficient data for training to cover all conditions possibly found in the practical application of EIS experiments. The electrochemical and physical parameters and their ranges are given in Table 1.

2.2. DNN Construction for the EIS Model Classification

To extract meaningful information from the simulated EIS data, the EIS spectrums were split into six features corresponding to the imaginary part (Z''), phase angle (ϕ), magnitude ($|Z|$) of the impedance (Figures S1 and S2, Supporting Information), and their augmented flipping value (multiplied by -1), as shown in Figure S3, Supporting Information. For example, a complex-valued 100 points of EIS data (array size of 100) turn

Table 1. Electrochemical and physical parameters and range.

Parameter	Range
Datapoint	100 points
Frequency (F)	10^{-2} – 10^6 Hz
Resistance (R)	10^{-1} – 10^4 Ω
Ideality factor (α)	0.8–1
Nonideal capacitance (Q)	10^{-5} – 10^{-3} s Ω^{-1}
Warburg coefficient (σ)	1 – 10^3 Ω s ^{1/2}

into a real (100,6) array. Hence, the input layer is 100-D vectors with six channels for each EIS spectrum. Additionally, the extra augmented (flipping) features not only increase the variety of the features but also help in the prevention of overfitting problems and increase the generalization of the DNN model.^[34,35] The 1D convolutional neural network (CNN) layers (conv1D) were employed as a feature extraction layer.^[36] The first conv1D layer consists of 64 filters and a kernel size of 32. The kernel size determines the length of the convolution matrix (Figure S4, Supporting Information), and the number of filters represents the number of extracted features. In this case, a real (100,6) input array would turn into a real (100,64) array, which means six input spectra with 100 points each are convoluted into 64 extracted features with 100 points on each dataset, as shown in Figure S5, Supporting Information. The first conv1D layer is followed by a series of conv1D layers with 128, 256, 512, and 768 filters and kernel sizes of 16, 8, 4, and 2, respectively. The final extracted features were 768 features. Next, the spatial dropout 1D layer was introduced into the network, followed by the batch normalization layer, as shown in Figure 3a.

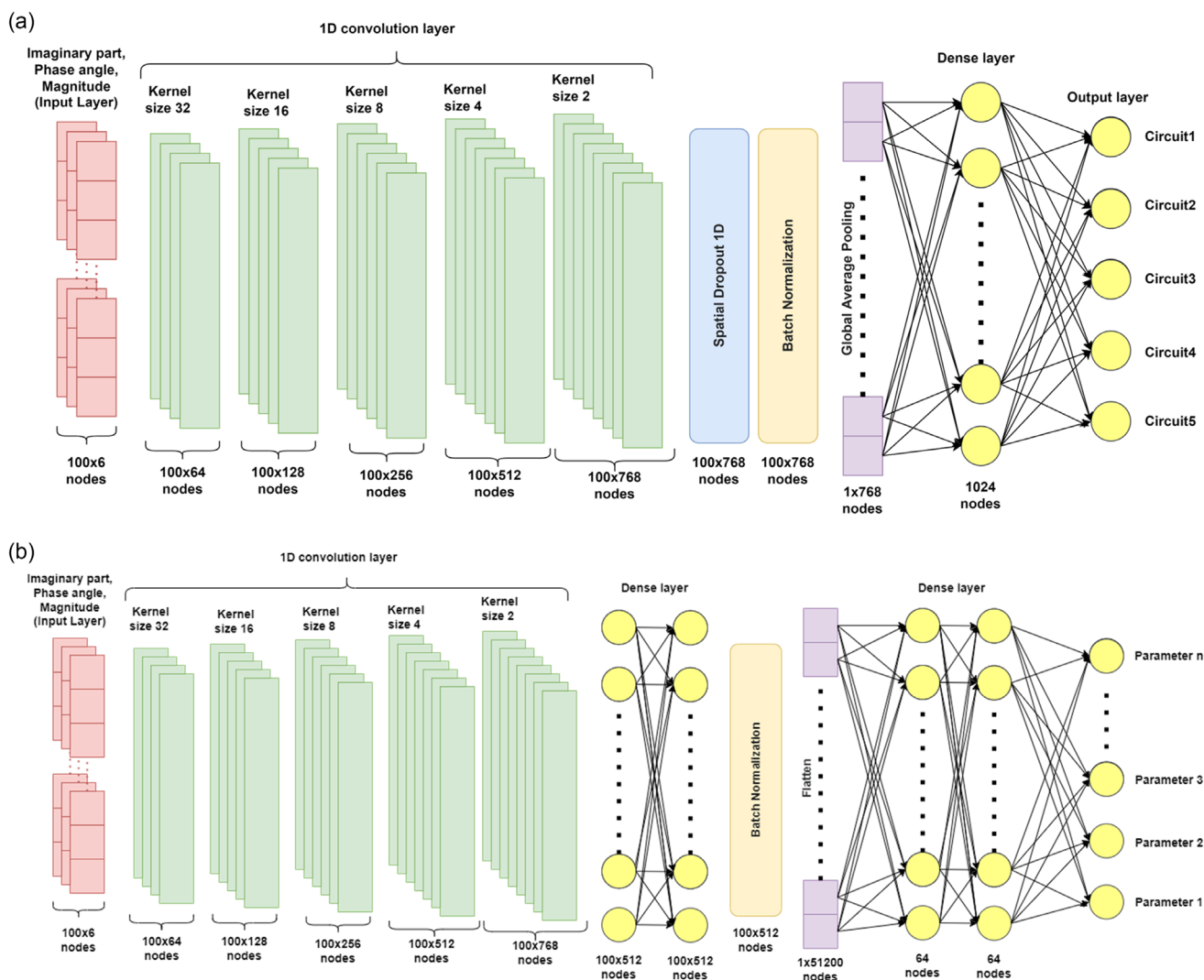


Figure 3. The scheme of the neural networks used for a) the EIS circuit classification and b) parameters regression.

For the classification part, the global average pooling 1D layer was employed to map a 100 by 768D feature into a single 768D vector.^[37] After that, there is another dense (fully connected) layer with 1024 nodes and a five-node output layer. The Softmax function was applied to the output layer, transforming the output value into a probability distribution for each circuit model. The rectified linear unit (ReLU) has also been used for conv1D and dense layer activation functions. Here, the series of conv1D layers operate as a feature extraction module, starting with a large kernel size of 32.

The convoluted feature represented the big picture of the entire data with a kernel size of 32 (Figure S4, Supporting Information) and narrowed down to the kernel size of 2 for the small details, potentially allowing the network to extract more detailed features from the data. Therefore, increasing the number of filters (from 64 to 768) results in sufficient features for the network to classify the circuit model. The spatial dropout 1D layer allows the convolution network to drop a node unit (along with connections) to prevent the overfitting problem as well as the batch normalization layer.^[38,39] In addition to that, the effect of kernel size was observed on the series of kernel sizes

that begins with 16, 32, 64, and 128, as shown in Figure S6, Supporting Information. The DNN model, whose kernel size begins with 16 and 32, shows high accuracy without a sign of overfitting. In contrast, the model whose kernel size begins with 64 and 128 shows significantly lower accuracy and also shows a signal of overfitting. Here, Figure S7, Supporting Information, shows a minor increase in accuracy, but the correct prediction for circuit C5 is significantly increased with a kernel size of 32. As a result of the trade-off between improved accuracy and prediction results on C5, the kernel size starting at 32 is more suitable for this DNN model.

2.3. Deep Neural Network Construction for the EIS Parameters Regression

The DNN model for EIS parameters regression is a challenging problem since the number of parameters on each circuit model is different. For example, C1 has three parameters (R_s , R_1 , Q_1), whereas C3 has four parameters (R_s , R_1 , Q_1 , σ). To archive a

high-performance DNN model, the regression models were constructed by keeping the same input layer and the feature extraction module from the classification model, as shown in Figure 3b. After that, the regression module was added, starting by mapping the 100 by 768-D features into 100 by 512 nodes with a series of two dense layers, and then the batch normalization layer was added for the robustness of the model. Here, the 100 by 512 nodes were mapped into a single 51 200D vector by flattened layer for further regression process. At this point, the usual ReLU activation function was replaced by the linear activation function for the series of two dense with 64 nodes on each layer. The final output layer calculates each parameter value that corresponds to the circuit model.

2.4. Training and Evaluation of the EIS Model Classification

For training the DNN, the simulated datasets were split into the training set and the validation set with a ratio of 80:20, respectively. Adaptive moment estimation (Adam) was used as an optimizer with the categorical cross-entropy (CCE) loss function. The CCE loss function was defined in Equation (1) where N is the number of data, C is the number of classes, y_{ij} is the true class, and \hat{y}_{ij} is the prediction.

$$L(y_{ij}, \hat{y}_{ij}) = -\frac{1}{N} \sum_{j=1}^N \sum_{i=1}^C y_{ij} \log(\hat{y}_{ij}) \quad (1)$$

Table 2. List of hyperparameters with the best-fitted values.

Hyperparameters	Value
Batch size	64, 256, 1024
Dropout rate	0.5, 0.6, 0.7, 0.8
Number of data in each circuit (data size)	1024, 2048, 4096, 8192, 16 384, 32 768

With the Keras API,^[40] the learning rate was set to be reduced with a condition of 50% reduced when a loss value stopped decreasing after 20 epochs. Here, there are several hyperparameters to be tuned, such as the batch size, dropout rate, and the number of data (i.e., data size) in each circuit. The hyperparameter values for the DNN in this work are given in Table 2. The tuning process has been done by matching the batch size and dropout rate with the 4096 trained dataset for each circuit with 50 epochs.

The accuracy and the loss are shown in Figure 4. Figure 4b shows the maximum accuracy of 58.72% at a batch size of 1024 and a dropout rate of 0.7, while Figure 4c shows the lowest loss value of 0.86 at the same batch size and dropout rate, showing that a batch size of 1024 and a dropout rate of 0.7 are the optimum parameters for the model. Since increasing batch size may not result in better accuracy, the independent impact of batch size on accuracy may be unclear. However, at a dropout rate of 0.7, it is clearly seen that the increase in batch size results in increasing accuracy and lowering the loss value. In contrast, the loss values for the dropout rates of 0.5 and 0.8 are relatively high, indicating that the networks may drop too few features at the 0.5 rate and too many features at the 0.8 rate, resulting in low accuracy. Figure 4a depicts the expected exponential curve for the effect of data size on the accuracy, with the best accuracy of 78.92% at a data size of 32 768. The accuracy increased significantly from a data size of 0 to 10 000 and then slowed dramatically. In addition to that, the effect of augmented (flipping) features has also been observed, as shown in Figure S8, Supporting Information. The fact that the generalization gap is getting smaller suggests that the model is becoming generalized, which helps to prevent the overfitting problem. Since the hyperparameters were optimized, the final DNN model would be trained under the optimum condition: a batch size of 1024, a dropout rate of 0.7, and a data size of 32 768.

To train the final optimized model, EIS data of 32 768 spectra for each circuit were simulated. The obtained datasets of 163 840 spectra were split into 131 072 spectra for the training set and

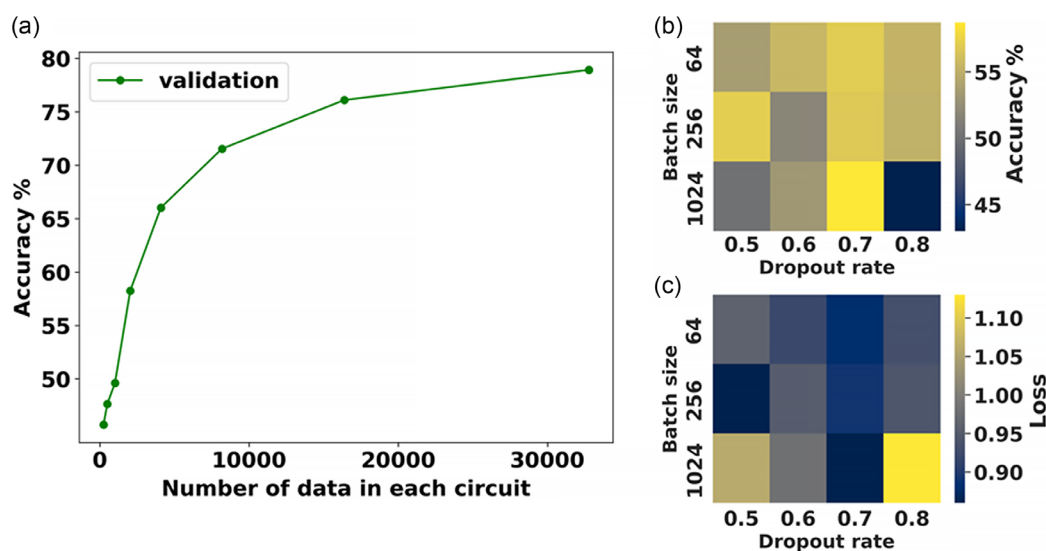


Figure 4. The effect of hyperparameters: a) effect of the number of data on the accuracy, b) effect of batch size and dropout rate on the accuracy, and c) effect of batch size and dropout rate on loss value.

32 768 spectra for the validation set, respectively. The training result of the optimized model is shown in Figure S9, Supporting Information, along with a confusion matrix. The confusion matrix presents the true circuit model on the left-hand side and the predicted circuit using the DNN classification model on the bottom. The value on the left-to-right diagonal line represents the correct prediction. The classification result of the validation set with the optimized model shows a high accuracy of 78.92%, which indicates that most cases are properly identified, especially on the first four circuits. However, there are some mispredictions on C5. For example, when the total number of the spectrum on C5 was 6637, only 2704 were correctly predicted.

At this point, the post-training optimized model further performs the classification on the test dataset: freshly simulated EIS spectra (never involved in the training process), together with the receiver operating characteristic (ROC) curve, to determine the classification performance of the DNN model. The freshly simulated dataset consists of 6500 EIS spectra on each circuit (32 500 spectra in total). These spectra were used to evaluate the model. The confusion matrix in Figure 5a shows the evaluation result on the test dataset, which has an accuracy of 78.92%.

However, determining the classification performance of C5 from the confusion matrix is still difficult. Hence, the ROC curve has been used to determine the trade-off between true-positive rate (TPR) and false-positive rate (FPR),^[41] as shown in Figure 5b. The dashed diagonal line represents the TPR is equal to the FPR, which means that the portion of correct classification of the true model is equal to the portion of incorrect classification of the false model. In other words, any point on the dashed diagonal line represents the random guess of the classifier. The area under the ROC curve (AUC) also helps to evaluate the performance of the classifier. The ideal classifier will occupy the perfect rectangle with an AUC of 1. In contrast, if the AUC is lower than 0.5, it might indicate a poor classifier. In Figure 5b, the DNN model shows a high-performance classifier with AUC values above 0.95 for circuits of C1, C2, C3, and C4 and AUC of 0.87 for C5, respectively. The average AUC for all the circuits was 0.95. Even though C5 is evaluated with an AUC value lower than those of other circuits, the ROC is still very far from the random guess diagonal line. These indicate that the ML strategy with the DNN model is a high-performance classifier for EIS circuit classification.

Here, Figure 6 shows an example of the classification result with a probability distribution. The EIS spectra of C3, C4, and C5 in Figure 6 have a very similar pattern in their spectra. As shown in the classification results of ML, the DNN model can still identify the true circuit for those spectra. As presented on the right-hand side in Figure 6, the bar charts represent the probability distribution over the different types of circuits. In contrast, Figure S10, Supporting Information, shows an example of the mispredictions from the DNN model. The result shows that the misprediction on C5 (Figure S10c, Supporting Information) may be due to the pattern of the spectrum being very close to the pattern of C3 (a single semicircle with a 45° straight line). Hence, mispredictions may exist due to the nearly identical spectrum among circuit models, and this mistake also occurs with experienced electrochemists. Here, the conditions that may lead to producing the nearly identical spectrum are shown in Figure S11, Supporting Information. To observe the

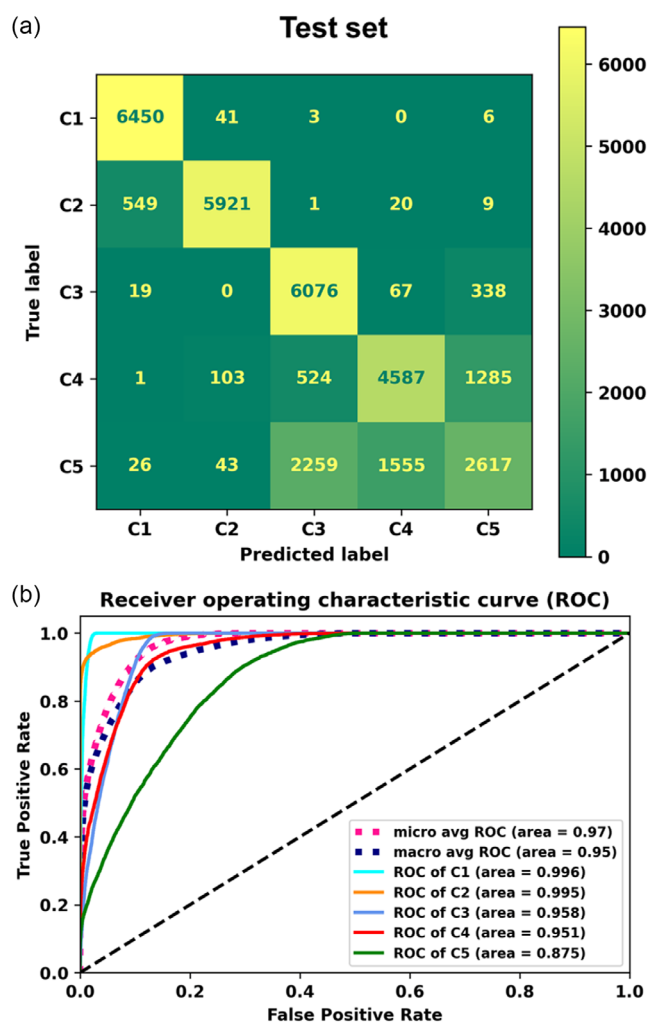


Figure 5. a) The confusion matrix and b) receiver operating characteristic (ROC) curve for the deep neural network (DNN) classification were evaluated on the test set, yielding an accuracy of 78.92%.

effect of the nearly identical spectrum, the optimized model was trained and evaluated on the new dataset with a low number of identical spectra, the simulation conditions provided in Table S1, Supporting Information. The result shows an improvement in accuracy from 78.92% to 91.76% (Figure S12a, Supporting Information). Additionally, the ROC shows an AUC value of 0.974 on C5, which is significantly improved from the original value of 0.875 (Figure S12b, Supporting Information).

Further study has been performed by introducing an extra equivalent circuit called C6 (Figure S13a, Supporting Information). At this point, the DNN model had been trained using 6 different circuit models. Circuit C6 has a signature pattern of 3 semicircle arcs with a 45° straight line. However, C6 also shows a similar spectrum to C3, C4, and C5, as shown in Figure S13b, Supporting Information. The evaluation result shows that the additional circuit decreased the accuracy by a value of 5.4% since the misprediction was increased due to a false negative on C4, as shown in Figure S13c, Supporting Information. In a similar way, the lowering in value of the

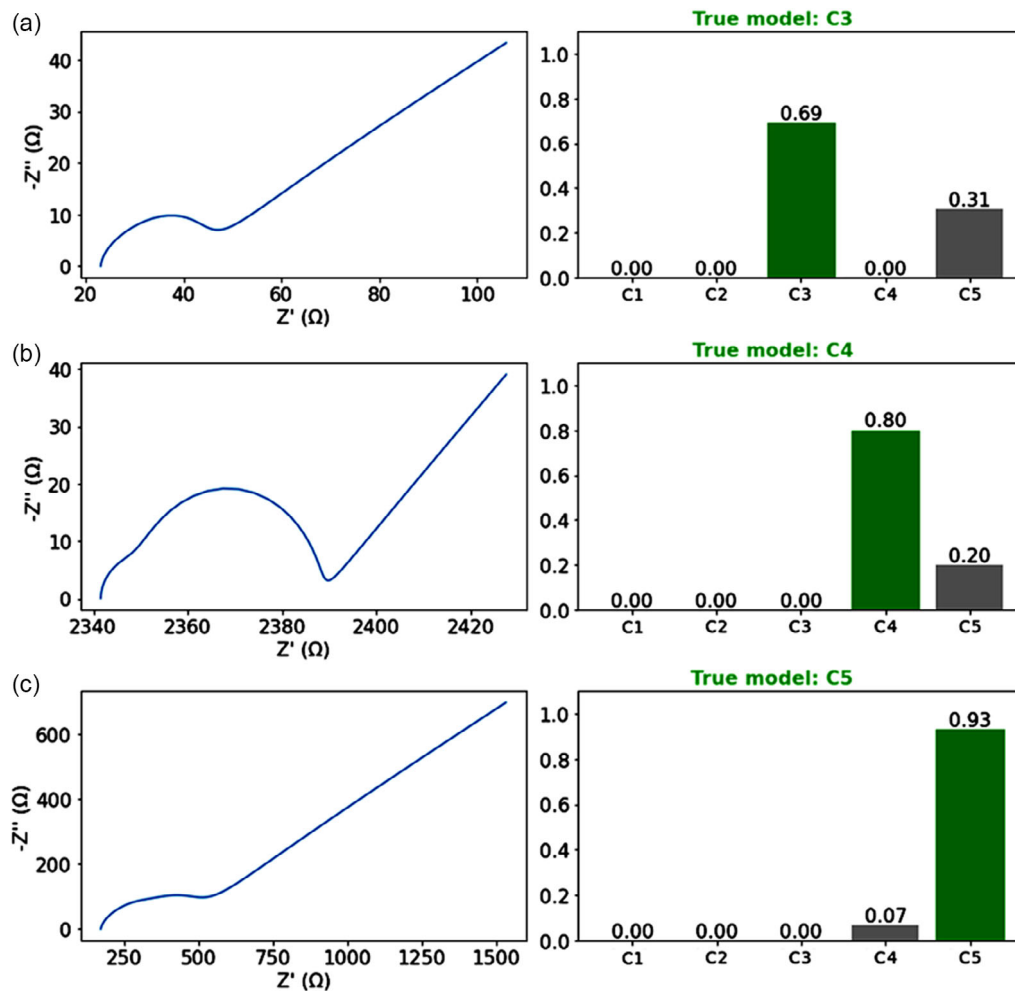


Figure 6. The example of correct classification results from the DNN model with the probability distribution on the right-hand side, in the case of a) C3, b) C4, and c) C5.

AUC of C4 indicated the confusion of the C4 classifier in the prediction of the C4 spectrum (Figure S13d, Supporting Information). Here, it is suggested that the identical spectrum significantly affects the accuracy and classification performance of the DNN model. Nevertheless, the given probability distribution is significantly helpful in narrowing down the choice of circuit model as a suggestion for the experimenter, and consequently, the DNN model is able to suggest high-potential models, which can save a lot of time.

In addition to that, the DNN model has been evaluated against the noisy data, as shown in Figure 7. To add noise to the original data, the SNR in the decibel (dB) unit was converted into an SNR value using Equation (2). The noise value was generated by a random normal distribution function with a mean value from the mean of the original spectrum and an SD value from Equation (3).

$$\text{SNR}_{\text{dB}} = 10 \log_{10}(\text{SNR}) \quad (2)$$

$$\text{SD} = \frac{\text{Mean of the original spectrum}}{\text{SNR}} \quad (3)$$

Then, the random normal distribution noise was added to the respective circuits (C1–C5) with a different SNR of 80, 40, and 20 dB. For instance, Figure 7a–c (top part) shows the comparison of the original data of C3 and the noisy data of C3. The bottom part of Figure 7a–c shows the confusion matrix of the DNN model evaluated on C1–C5. The obtained result shows that the DNN model achieved an accuracy of 78.8%, 70.4%, and 37.1%, even with the SNR of 80, 40, and 20 dB, respectively.

The lower SNR reveals a higher false positive on circuits C2 and C4, suggesting that the trained DNN model is sensitive to the curve shape on the spectrum since C2 and C4 presented multiple semicircles in the spectrum. However, the poor SNR (as in SNR = 20 dB) in Figure 7c is a rare event that has not been found often in practical experiments since the present equipment is good enough to increase the SNR by adjusting the scan rate or by averaging the collecting data, automatically. In contrast, the sensitivity to the curve shape of the DNN model is advantageous in classifying the spectrum with unclear semicircles in a noiseless manner. Moreover, the performance of DNN was assessed by employing the dataset with simulation parameters

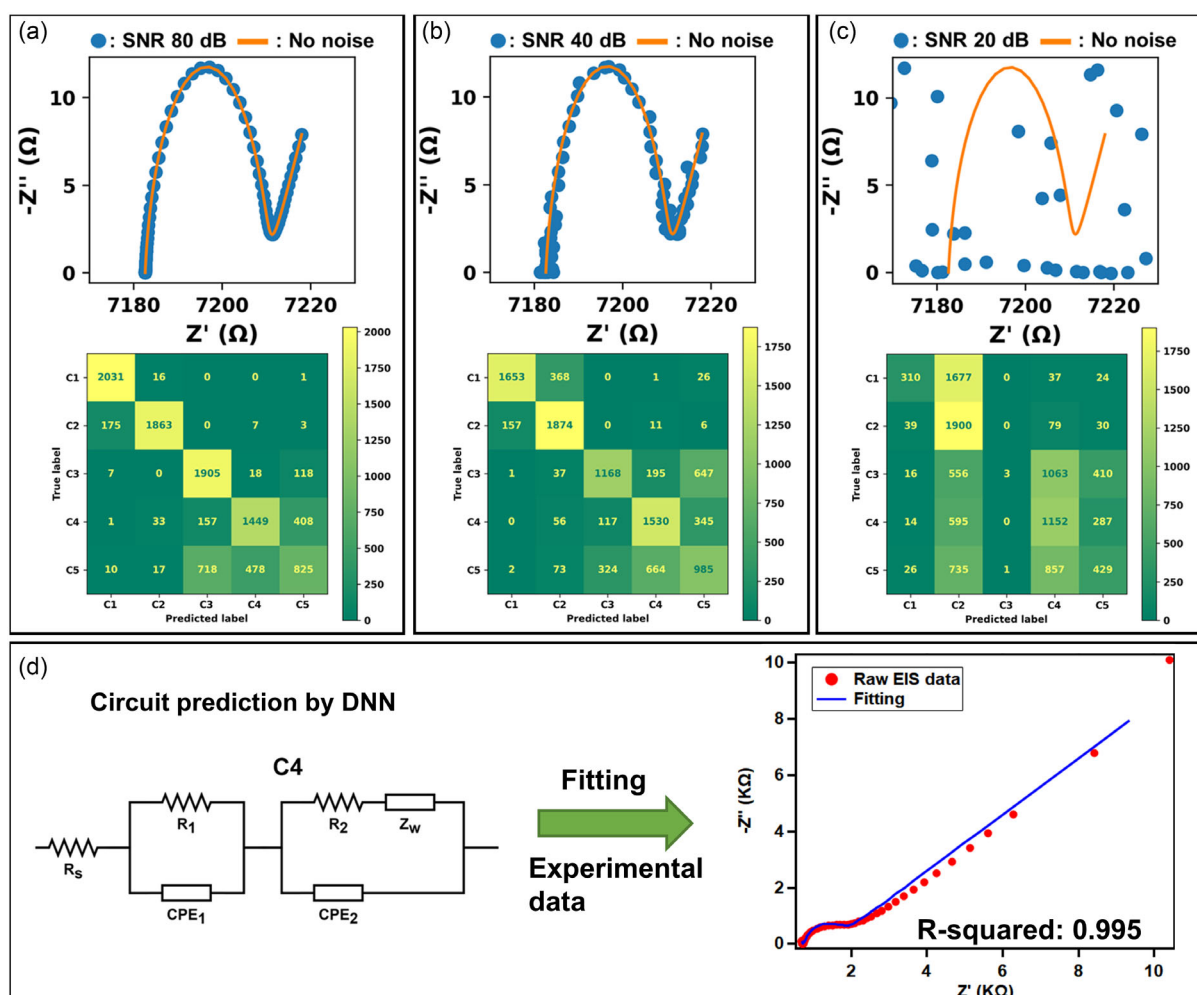


Figure 7. The noise dataset with various signal-to-noise ratios (SNR): a) SNR 80 dB, b) SNR 40 dB, and c) SNR 20 dB, and their confusion matrix (on the bottom part). d) The fitting of experimental EIS data with DNN predicted circuit (C4) using ZMAN software.

outside of the training set. The lower and upper boundaries of each parameter in the training set were increased by 10%, 20%, and 30%, resulting in a total of 6 different ranges for each parameter (Figure S14a, Supporting Information). Changes in resistance range, nonideal capacitance range, and Warburg coefficient range have small effects on the accuracy of the DNN model. In contrast, the change in frequency range and ideality factor range has a significant effect on the accuracy of DNN. However, the EIS spectrum in electrochemistry is usually measured in the frequency range from 10 mHz to 1 MHz, and the nonideality factor can range from 0 to 1, but is usually found between 0.8 and 1,^[11] which are both in the training range. In addition to that, the effect of changing the boundary for multiple parameters (R , Q , and σ) was observed by using the confusion matrix (Figure S14b, Supporting Information) under the condition where both the upper and lower boundaries were extended by 30%. As a result, the accuracy was found to be 76.76% with a high value of correct prediction, indicating that the DNN model is able to make a correct prediction within a range of 30% added to the boundary of the training range.

As a proof of concept, the practical EIS experiment on a commercial screen-printed carbon electrode was performed in the electrolyte, which contains 1 M KCl with 10 mM $K_3Fe(CN)_6$ and 10 mM $K_4Fe(CN)_6$. The DNN model runs very fast in classification. As shown in Figure S15, Supporting Information, the experimental spectrum was categorized into the C4 model in 2 s. Figure 7d shows a fitting result using ZMAN software of the C4 model on an experimental spectrum that is well matched with the DNN suggested circuit model. The R^2 value was found to be 0.995, suggesting an acceptable fitting result.

Additionally, the classification performance of various ML models was compared in Table S2, Supporting Information, indicating that the DNN model performed remarkably well in this study. Thus, the ML strategy with the DNN model is a high-potential approach to be used in the novel research.

2.5. Training and Evaluation of the EIS Parameters Regression

The training process of the regression model was done by training the model on each circuit separately (5 ML models in total),

which employed EIS data of 131 072 spectra on each circuit. The train data to validation data ratio remains at 80:20. The applicable hyperparameter, such as batch size, was applied. However, the mean absolute error (MAE) loss function was used instead of the CCE loss function. Here, the regression model tries to minimize the MAE loss function value (shown in Equation (4), where y_{ij} is the true value, and \hat{y}_{ij} is the prediction, and N is the total of data points) by using the Adam optimizer.

$$L(y, \hat{y}) = \frac{1}{N} \left(\sum_{i=1}^N |y_i - \hat{y}_i| \right) \quad (4)$$

To evaluate the performance of the regression model, freshly simulated test datasets, which consist of 5243 spectra for each circuit model, were employed. **Figure 8** shows the parameters prediction of a trained model on a test dataset of C3. The prediction result shows a very well-matched predicted parameter with an R^2 of 0.999 on R_s , R_1 , σ , and 0.992 on Q_1 . The MAE value is relatively low compared to their parameter range value. They are suggesting the high potential of utilizing the regression model on a more complicated EIS circuit model. To determine the prediction performance of the regression model on the complicated circuit model, the prediction results for each circuit model were summarized in the comparison table (**Table 3**) with the R^2 value presented (the prediction results of C1, C2, C4, and C5 were shown in Figures S16–S19, Supporting Information, respectively).

It is clearly seen that the regression models are highly accurate in predicting R_s and σ values with the minimum R^2 of 0.998 on the applicable circuit model. Moreover, the circuit model without the series of R/Q (C1, C3, C5) is also able to achieve $R^2 \geq 0.950$ on

Table 3. Comparison of R^2 of the predicted EIS parameters on the different circuit models.

Circuit	Parameter number	R^2					
		R_s	R_1	R_2	Q_1	Q_2	σ
C1	3	0.999	0.999	–	0.951	–	–
C2	5	0.999	0.453	0.484	0.133	0.205	–
C3	4	0.999	0.999	–	0.992	–	0.999
C4	6	0.999	0.538	0.463	0.599	0.518	0.998
C5	6	0.999	0.955	0.959	0.982	0.568	0.999

every relevant parameter except Q_2 , indicating a highly accurate prediction performance of the model even though it was performed on the complicated circuit model. In contrast, the circuit model with the series of R/Q (C2, C4) shows a very low R^2 value, which may suggest that the input data and extracted features might not be informative enough to predict EIS parameters on the circuit with the series of R/Q .

3. Conclusion

We successfully constructed a DNN model and trained it on electrochemical impedance spectra for five commonly encountered equivalent circuit models, which are not only capable of classifying the circuit model with high accuracy but also allow the SNR of 40 dB. A few misclassifications may occur as a result of the similar spectrum shapes of the various models. Additionally, the DNN regression model is capable of accurately predicting

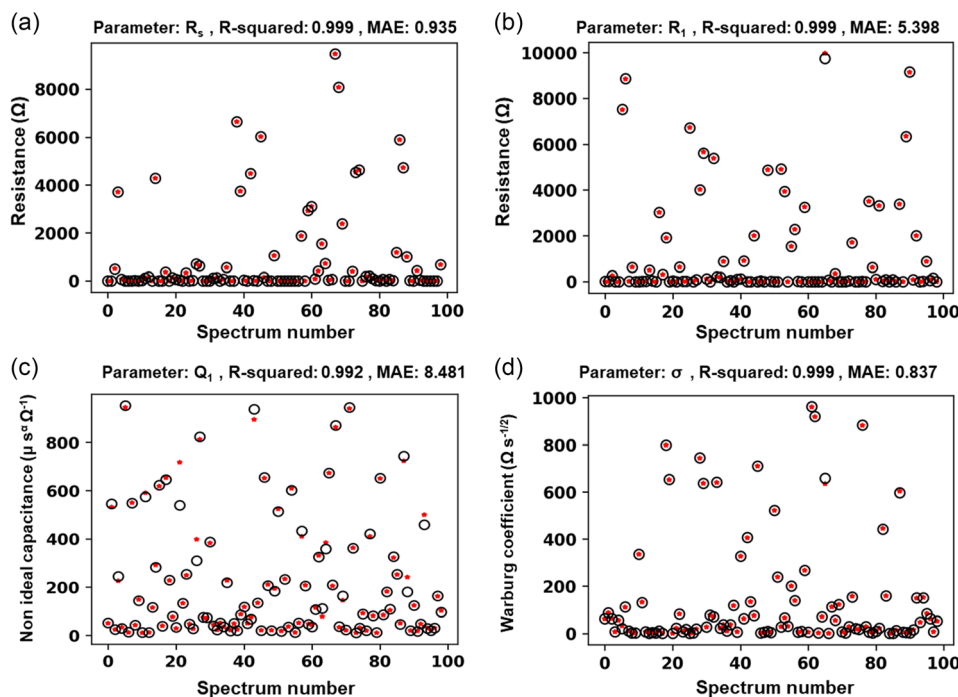


Figure 8. The prediction of C3 parameters for a) R_s , b) R_1 , c) Q_1 , and d) σ , presented by the first 100 spectra with R^2 and mean absolute error (MAE) values. The black circle denotes a prediction value, and the solid red circle denotes a real value.

EIS parameters, especially R_s and σ , with the minimum R^2 of 0.998. This approach has the potential to replace the conventional searching strategy, which is time-consuming and highly dependent on the experimenter. Conventional interpretation methods can take up to 20 min to interpret a single spectrum, depending on experience. However, our approach can recognize and predict an EIS parameter in a few seconds for multiple spectra and can suggest or narrow the circuit model selection without involving a human. It is clear that this approach has an advantage over the traditional approach with rapid analysis, unbiased results, a straightforward process, and is based on a statistical and theoretical basis. Thus, the ML strategy with the DNN model is a beginning point for the advancement of automatic analysis in electrochemical systems.

4. Experimental Section

EIS Data Acquisition: To simulate the EIS spectrum, the elements such as resistor (R), constant phase element (CPE), and finite Warburg diffusion element (W) were defined as essential elements to fit into the EIS equivalent circuit model as described in Section S1, Supporting Information. In addition to that, the complicated equation of the EIS equivalent circuit could be turned into a customizable simulation program with the help of the Python package, including NumPy and SciPy libraries, to simulate five EIS equivalent circuits, which were commonly used in electrochemical systems (Figure 1). The Python code for EIS simulation is available at https://github.com/DulyawatDoonyapisut/EIS_ML.

Deep Neural Network Building and Training: The DNN was built using a Python package that included the TensorFlow^[42] and Keras^[40,43] libraries. Five main operating layers, such as CNN layers, spatial dropout 1D layer, batch normalization layer, global average pooling 1D layer, and dense (fully connected) layer, were employed as the building blocks for the neural networks. To train a large number of datasets, the Colab environment was used with the NVIDIA Tesla P100 GPU and 16 GB of memory.

The simulated EIS data with 1024, 2048, 4096, 8192, 16 384, and 32 768 spectra on each circuit model were used as a database for the training process. The training data was separated into spectra data (input data) and circuit-labeled data (output data). The input data covered important features such as physical meaning and the morphology of the spectrum. In this model, the imaginary part (Z''), the phase angle (θ), and the magnitude ($|Z|$) of the impedance were employed as input data, as shown in Figure S1, Supporting Information. The imaginary part of impedance (Z'') provided information about the nonideal capacitance (Q) and the Warburg diffusion element (W), both which belonged to the imaginary part, as shown in Equations (S2) and (S3), Supporting Information. Changes in phase angle (θ) were very helpful in determining the complexity of the circuit model since the resistor, CPE, and Warburg diffusion element gave different phase angle changes, as shown in Figure S2, Supporting Information. The magnitude of the impedance ($|Z|$) provided information on the overview of the spectrum with the impedance value. The Python code for the DNN model was available at https://github.com/DulyawatDoonyapisut/EIS_ML.

Supporting Information

Supporting Information is available from the Wiley Online Library or from the author.

Acknowledgements

This work was supported by the Basic Science Research Program and Basic Research Laboratory through the National Research Foundation of Korea (NRF), funded by grants from the Ministry of Science

(Grant nos. NRF-2022R1A2B5B01001764 and NRF-2021R1A4A1024129) and AI Convergence Research Fund, Sungkyunkwan University, 2022.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Keywords

deep learning, EIS analysis, EIS prediction, impedance machine learning

Received: February 10, 2023

Revised: March 28, 2023

Published online: April 21, 2023

- [1] B. N. Choi, J. Y. Seo, Z. An, P. J. Yoo, C.-H. Chung, *Chem. Eng. J.* **2022**, 430, 132807.
- [2] D. Yang, Q. Li, F. Shen, Q. Wang, L. Li, N. Song, Y. Dai, J. Shi, *Electrochim. Acta* **2016**, 189, 32.
- [3] S. Lee, M. Kim, K. T. Lee, J. T. S. Irvine, T. H. Shin, *Adv. Energy Mater.* **2021**, 11, 2100339.
- [4] B. N. Choi, J. Y. Seo, B. Kim, Y. S. Kim, C.-H. Chung, *Appl. Surf. Sci.* **2020**, 506, 144884.
- [5] K. Morimoto, T. Kusumoto, K. Nishioka, K. Kamiya, Y. Mukouyama, S. Nakanishi, *ACS Appl. Mater. Interfaces* **2020**, 12, 42803.
- [6] Y. Zheng, Z. Shi, D. Guo, H. Dai, X. Han, *J. Power Sources* **2021**, 489, 229505.
- [7] V. Balasubramani, S. Chandraleka, T. S. Rao, R. Sasikumar, M. R. Kuppusamy, T. M. Sridhar, *J. Electrochem. Soc.* **2020**, 167, 37572.
- [8] B.-A. Mei, J. Lau, T. Lin, S. H. Tolbert, B. S. Dunn, L. Pilon, *J. Phys. Chem. C* **2018**, 122, 24499.
- [9] A. Sacco, *Renewable Sustainable Energy Rev.* **2017**, 79, 814.
- [10] Y. Ding, J. Li, K. Yan, J. Zhang, *Sens. Actuators B Chem.* **2022**, 369, 132259.
- [11] A. R. C. Bredar, A. L. Chown, A. R. Burton, B. H. Farnum, *ACS Appl. Energy Mater.* **2020**, 3, 66.
- [12] J. Jeong, T. Kim, B. J. Lee, J. Lee, *Adv. Intell. Syst.* **2023**, 5, 2370003.
- [13] W. Shin, K. K. Min, J.-H. Bae, J. Kim, R.-H. Koo, D. Kwon, J.-J. Kim, D. Kwon, J.-H. Lee, *Adv. Intell. Syst.* **2023**, 2200377.
- [14] G. Martin, S. Kaviraj, A. Hocking, S. C. Read, J. E. Geach, *Mon. Not. R. Astron. Soc.* **2020**, 491, 1408.
- [15] J. Brehmer, S. Mishra-Sharma, J. Hermans, G. Louppe, K. Cranmer, *Astrophys. J.* **2019**, 886, 49.
- [16] M. B. Schäfer, F. Ohme, A. H. Nitz, *Phys. Rev. D* **2020**, 102, 63015.
- [17] C. Kim, R. Batra, L. Chen, H. Tran, R. Ramprasad, *Comput. Mater. Sci.* **2021**, 186, 110067.
- [18] M. Fujinami, J. Seino, H. Nakai, *Bull. Chem. Soc. Jpn.* **2020**, 93, 685.
- [19] I. Funes-Ardoiz, F. Schoenebeck, *Chem* **2020**, 6, 1904.
- [20] C. Yan, G. Li, *Adv. Intell. Syst.* **2023**, 2200243.
- [21] M. Scherer, F. Schmidt, O. Lazareva, J. Walter, J. Baumbach, M. H. Schulz, M. List, *Nat. Comput. Sci.* **2021**, 1, 183.
- [22] M. A. Konerman, L. A. Beste, T. Van, B. Liu, X. Zhang, J. Zhu, S. D. Saini, G. L. Su, B. K. Nallamothu, G. N. Ioannou, *PLoS One* **2019**, 14, 0208141.

- [23] F. Noé, G. De Fabritiis, C. Clementi, *Curr. Opin. Struct. Biol.* **2020**, 60, 77.
- [24] Y. Wu, B. Jiang, Q. Wang, Y. Liu, C. Wu, M. Wu, H. Song, *Adv. Intell. Syst.* **2023**, 2200352.
- [25] F. V. Lavrentev, I. S. Rumentsev, A. S. Ivanov, V. V. Shilovskikh, O. Y. Orlova, K. G. Nikolaev, D. V. Andreeva, E. V. Skorb, *ACS Appl. Mater. Interfaces* **2022**, 14, 7321.
- [26] V. C. Rodrigues, J. C. Soares, A. C. Soares, D. C. Braz, M. E. Melendez, L. C. Ribas, L. F. S. Scabini, O. M. Bruno, A. L. Carvalho, R. M. Reis, R. C. Sanfelice, O. N. Oliveira, *Talanta* **2021**, 222, 121444.
- [27] Q. Xiao, Z. Zhou, Z. Shen, J. Chen, C. Gu, L. Li, F. Chen, H. Liu, *Sens. Actuators B Chem.* **2023**, 375, 132922.
- [28] G. F. Kennedy, J. Zhang, A. M. Bond, *Anal. Chem.* **2019**, 91, 12220.
- [29] S. Zhu, X. Sun, X. Gao, J. Wang, N. Zhao, J. Sha, *J. Electroanal. Chem.* **2019**, 855, 113627.
- [30] Z. Zhao, Y. Zou, P. Liu, Z. Lai, L. Wen, Y. Jin, *Electrochim. Acta* **2022**, 418, 140350.
- [31] V. Bongiorno, S. Gibbon, E. Michailidou, M. Curioni, *Corros. Sci.* **2022**, 198, 110119.
- [32] W. Choi, H.-C. Shin, J. M. Kim, J.-Y. Choi, W.-S. Yoon, *J. Electrochem. Sci. Technol.* **2020**, 11, 14.
- [33] T. Momma, T. Yokoshima, H. Nara, Y. Gima, T. Osaka, *Electrochim. Acta* **2014**, 131, 195.
- [34] B. K. Iwana, S. Uchida, *PLoS One* **2021**, 16, 0254841.
- [35] A. Flores, H. Tito-Chura, H. Apaza-Alanoca, *Proceedings of the Future Technologies Conference (FTC)* (Ed.: K. Arai), Vol. 2, Springer International Publishing, Cham **2021**, pp. 492–506.
- [36] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D. J. Inman, *Mech. Syst. Signal Process.* **2021**, 151, 107398.
- [37] M. Lin, Q. Chen, S. Yan, (Preprint) arXiv1312.4400 **2013**.
- [38] S. Ioffe, C. Szegedy, *32nd Int. Conf. Mach. Learn. ICML 2015*, Vol. 1, Lille, France **2015**, p. 448.
- [39] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, C. Bregler, In *Proc. of the IEEE Conf. Computer Vision and Pattern Recognition*, IEEE, Piscataway, NJ **2015**, pp. 648–656.
- [40] F. Chollet, Keras, GitHub, **2015**.
- [41] D. Vining, G. W. Gladish, *Radiographics* **1992**, 12, 1147.
- [42] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, (Preprint) arXiv1603.04467 **2016**.
- [43] A. Gulli, S. Pal, *Deep Learning With Keras*, Packt Publishing Ltd, Birmingham UK **2017**.