Nuclear Engineering and Technology 54 (2022) 3027–3033

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Comparison and optimization of deep learning-based radiosensitivity prediction models using gene expression profiling in National Cancer Institute-60 cancer cell line

Euidam Kim [a], Yoonsun Chung [a, *]

[a] Department of Nuclear Engineering, Hanyang University, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 25 October 2021
Received in revised form
21 December 2021
Accepted 14 March 2022
Available online 17 March 2022

Keywords:
Radiosensitivity
Prediction
Deep learning
Model comparison
Gene expression
Survival fraction at 2Gy

ABSTRACT

Background: In this study, various types of deep-learning models for predicting *in vitro* radiosensitivity from gene-expression profiling were compared.
Methods: The clonogenic surviving fractions at 2 Gy from previous publications and microarray gene-expression data from the National Cancer Institute-60 cell lines were used to measure the radiosensitivity. Seven different prediction models including three distinct multi-layered perceptrons (MLP), four different convolutional neural networks (CNN) were compared. Folded cross-validation was applied to train and evaluate model performance. The criteria for correct prediction were absolute error < 0.02 or relative error < 10%. The models were compared in terms of prediction accuracy, training time per epoch, training fluctuations, and required calculation resources.
Results: The strength of MLP-based models was their fast initial convergence and short training time per epoch. They represented significantly different prediction accuracy depending on the model configuration. The CNN-based models showed relatively high prediction accuracy, low training fluctuations, and a relatively small increase in the memory requirement as the model deepens.
Conclusion: Our findings suggest that a CNN-based model with moderate depth would be appropriate when the prediction accuracy is important, and a shallow MLP-based model can be recommended when either the training resources or time are limited.

1. Introduction

In the post-genomic era, the development of a robust *in vitro* radiosensitivity prediction assay in cancer cell lines based on a plethora of genomic data is promising for application in various fields such as radiotherapy and radiological protection [1–3]. Prediction of radiation response in cancer versus normal tissue based on their genetic data is crucial to ensure patient safety and treatment outcome in the field of radiotherapy. Moreover, it is also important to consider the possible difference in radiosensitivity among the members in a single protection group to increase the effectiveness of radiological protection.

To date, various studies using statistical or machine learning-based methods such as principal component analysis (PCA), partial least squares (PLS), multi-genomic fused PLS (MGPLS), and support vector machine-based regression analysis have been conducted to predict radiosensitivity from genomic data [4–6]. Additionally, with the recent advance of deep learning and artificial intelligence technology, our previous study established deep-learning-based radiosensitivity prediction methodology, which presented the feasibility of a convolutional neural network (CNN)-based radiosensitivity prediction model based on the gene expression of National Cancer Institute-60 (NCI-60) cancer cell lines [7].

Deep-learning contains different types of model architectures based on various concepts, such as a basic straightforward multi-layered perceptron (MLP), a CNN for spatially coherent information processing, and a recurrent neural network (RNN) for time series data processing [8–10]. Deep-learning users can employ these pre-built reference models such as GoogleNet or VGGnet, or build up and modify their own models applying these concepts, based on the data to be processed [11,12]. Because of these characteristics, in addition to the specific model presented in our previous publication, there might be some other effective model

* Corresponding author. Department of Nuclear Engineering, Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul, 04763, Republic of Korea.
E-mail address: ychung@hanyang.ac.kr (Y. Chung).

https://doi.org/10.1016/j.net.2022.03.019

structures with different combinations of hidden layers considering each user's situation, such as available time and computational resources for model training, or the model's target accuracy [7].

Therefore, herein, we evaluated various types of models for predicting *in vitro* radiosensitivity from gene expression, aimed at deriving the most appropriate model types according to the user situation by comparing the model training time, calculation resources, and prediction performance.

2. Materials and methods

2.1. Intrinsic radiosensitivity index

The clonogenic cell survival fraction at a radiation dose of 2 Gy (SF2) was selected as a measurement of *in vitro* radiosensitivity of cancer cell line samples due to its wide usage and simplicity. The SF2 measured *in vitro* has been known to provide a good prediction of *in vivo* irradiation and therefore is considered as a standard of the *in vitro* radiosensitivity index with clinical evidence [4,13,14]. The measured (true) SF2 values that the deep-learning model aims to predict were obtained from previous publications [6,15].

2.2. Gene expression profiling

In the Gene Expression Omnibus (GEO; available at https://www.ncbi.nlm.nih.gov/sites/GDSbrowser; series accession number GSE32474 [16]) database, a microarray-based gene expression profiling of NCI-60 cancer cell lines was obtained (Affymetrix Human Genome U133 Plus 2.0; 54,675 probe sets). Duplicated or triplicated 174 samples from 59 cell lines of NCI-60 cancer cell lines except for MDA-N (not available on NCI-60), and all 54,675 probe sets were used as an input for the *in vitro* radiosensitivity prediction model.

2.3. Deep-learning-based radiosensitivity prediction models

In deep learning, MLP and CNN are the most widely used basic type of model architecture with very different characteristics. The MLP consists of a system of simple interconnected nodes based on matrix multiplication, whereas the CNN is based on the locally connecting kernel convolution operation [8,17]. The MLP represents good performance owing to the characteristic of matrix multiplication, which takes into account every node in the layer. However, because the number of parameters increases significantly as the layer deepens in MLP, it is considered very resource-intensive and suitable for relatively shallow, simple models [18].

Conversely, CNN-based models are likely to have their strengths when 1) dealing with data with intrinsic spatial information, and 2) reducing the number of trainable parameters through parameter sharing because it only connects the nodes within a certain range [19]. CNN-based models are classified into two different types: those with small and large kernels. Most widely known CNN-based image processing models, such as VGGNet, use large numbers of small, narrow kernels such as 2×2 or 3×3 convolution to extract the locally coherent features in general images such as curves (low-level features) or objects (high-level features) [11]. However, owing to the characteristic of one-dimensional gene expression in which the information contained in the gene is spread out arbitrarily, extracting relevant features or details using small kernels is challenging although a large amount of data is available [20]. Nevertheless, with the large kernel-based CNN, which has characteristics of both MLP (matrix multiplication of the entire nodes) and CNN (parameter sharing), relevant feature extraction due to its obtuseness to local data distribution may be achieved to deal with gene expression profiles.

Therefore, we compared the MLP-based models and CNN-based models with large and small kernels and attempted to find out what could be more effective for processing one-dimensional gene expression profiling data. First, based on the concept of MLP, the model with 54675 (input vector size)-1000-128-1 layers (called 'MLP-1'), the model with 54675-1000-32-1 layers (called 'MLP-2'), and that with 54675-8192-4096-2048-1024-512-300-128-32-1 layers (called 'MLP-3') were included in the comparing group. Likewise, as CNN-based models, the model with 3 convolutional layers with large-sized kernel and 5 fully connected (FC) layers (called '3-5 CNN'), the model with 5 convolutional layers with large-sized kernel and 3 FC layers (called '5-3 CNN'), that with 5 convolutional layers with large-sized kernel and 5 FC layers (called '5-5 CNN'), and finally, based on the concept of CNN and residual connection, previously established, well-known 34-layered residual networks with small-sized kernel and 6 FC layers (called 'Resnet-34') were also included in the comparison group [21]. The structures of the models are described in Table 1 in the form of (N x k), where N and k indicate that the number of the features and filter of the layer, respectively. Because gene expression profiling is a one-dimensional vector format, all convolutional layers are based on one-dimensional convolution, where the kernel strides in only one direction on the given gene expression vector [22].

2.4. Model training and validation

The hyper-parameters such as learning rate, batch size, and regularization rate were tuned through random searching, using the 1st fold of 6-fold cross-validation as a validation set. The cost function of each model was all the same as the mean squared error (mean squared deviation). To adequately train and validate each model, 10 rounds of 6-fold cross-validation were applied [23]. With folded cross-validation, predictions for all datasets could be obtained while adequately maintaining the model variance and bias. Data stratification of the folded cross-validation was identical to that in the previous study: each fold should not include one sample from a particular cell line to prevent overfitting the data for a certain cancer cell line [7]. The number of the samples from each tissue of origin in the 6-fold cross-validation is shown in Table 2. The average predicted SF2 values in 10 rounds of cross-validation were considered to be the final predicted SF2 value of a model to stabilize the prediction and minimize the predicted value's deviation. The model was evaluated and trained using the NVIDIA TITAN RTX and TensorFlow 1.14.0 framework based on Python version 3.6.8.

2.5. Model efficiency evaluation metrics

Comparison metrics are required to compare the prediction performance among the models. First, following the previous study, the absolute prediction error and the relative prediction error were defined as the absolute deviation between the predicted SF2 and measured SF2 and the division of absolute prediction error with the measured SF2 value, respectively, as shown below [7].

$$Absolute\ error_{sample} = |Predicted\ SF2_{sample} - Measured\ SF2_{sample}|$$

$$Relative\ error_{sample} = \frac{Absolute\ error_{sample}}{Measured\ SF2_{sample}}$$

The criteria for 'correct prediction' were defined as a prediction with an absolute prediction error within 0.02 (2% in terms of survival fraction) or the relative prediction error within 10% due to the

Table 1
Structures of the radiosensitivity prediction models.

Category	Models	Model structure
MLP-based	MLP-1	54675-1000-128-1
	MLP-2	54675-1000-32-1
	MLP-3	54675-8192-4096-2048-1024-512-300-128-32-1
CNN-based	3-5 CNN	(54675×1)-(4799×10)-(689×20)-(54×40)-800-300-100-32-1
	5-3 CNN	(54675×1)-(11621×10)-(2394×20)-(471×40)-(86×80)-(14×160)-500-21-1
	5-5 CNN	(54675×1)-(11621×10)-(2394×20)-(471×40)-(86×80)-(14×160)-800-300-100-32-1
	Resnet-34	34 layers of residual network (output size: 27392)-4000-1000-300-100-32-1

Acronyms: MLP, Multi-Layered Perceptron; CNN, Convolutional Neural Network.

Table 2
Numbers of the samples from each tissue of origin in the 6-fold cross-validation.

Tissue of Origin	Number of samples						
	1st fold	2nd fold	3rd fold	4th fold	5th fold	6th fold	Total
Leukemia	3	3	3	3	3	3	18
NSCLC	5	5	4	4	4	4	26
Colon	3	3	4	4	4	3	21
CNS	3	3	3	3	3	3	18
Melanoma	5	4	4	4	4	5	26
Ovarian	3	4	4	4	3	3	21
Renal	4	4	4	3	4	4	23
Prostate	1	1	1	1	1	1	6
Breast	2	2	2	3	3	3	15
Total	29	29	29	29	29	29	174

Acronyms: NSCLC, Non-Small Cell Lung Cancer; CNS, Central Nervous System.

known variability of clonogenic cell survival assay reported by Peters et al. [24]. Using these criteria for 'correct prediction,' the prediction accuracy was calculated as a percentage of correctly predicted samples from the entire sample. Each model's overall prediction accuracies were determined as the average accuracy in 10 rounds of cross-validation.

Concerning calculation efficiency, the model training time, amount of video random access memory (VRAM) consumption, and the model fluctuations were considered as the comparison metrics. Each model's training time was measured as the total training time divided by the number of training epochs. The amount of VRAM occupied under the same training condition in each model was determined as the model's VRAM consumption. The fluctuations of each model were quantified as the standard deviation of model prediction accuracy after 50,000 epochs where every model converges to a certain level.

2.6. Average of absolute prediction error and statistical comparison analysis

The average and standard deviation of the absolute prediction errors in each model were calculated. Then, the difference in the average value of the absolute prediction error between the models and a p-value acquired by multiple comparisons in one-way analysis of variance (ANOVA) using absolute prediction error were used to compare the models' prediction accuracy. All statistical analyses were performed using GraphPad Prism version 7.03 for Windows, GraphPad Software, San Diego, California USA, www.graphpad.com.

3. Results

3.1. Model evaluation: epoch-accuracy curves

Figs. 1 and 2 show the epoch-accuracy curves of each model in total epochs and the combined plot in low epochs, respectively. As depicted in Fig. 1, the fluctuations and prediction accuracy of the MLP-3 model (4.035, 93.10%, respectively) were both larger than that of MLP-2 (2.186, 86.78%, respectively), which implied that the fluctuations and prediction accuracy both were increased as the model became deeper for MLP-based models. In the case of the CNN-based models, A 3-5 CNN model showed high fluctuations (2.816) with high prediction accuracy (97.13%) while 5-3 CNN had low fluctuations (1.732) with low prediction accuracy (91.95%), and the 5-5 CNN featured the lowest fluctuations (1.162) and high prediction accuracy (96.55). The Resnet-34 model showed low fluctuations (1.54), but prediction accuracy was low (77.01%).

In Fig. 2, the MLP-based models converged in early epochs, CNN-based models converged slightly later, while Resnet-34 shows relatively late convergence, as also shown in the trend line drawn based on the Fourier series curve fitting. More specifically, in the early 150 epochs, MLP-2, MLP-3, 3-5 CNN, 5-3 CNN, 5-5 CNN, and Resnet-34 roughly approached accuracy of 66, 54, 33, 22, 29, and 13%, respectively, while their accuracies were increased to 77, 83, 90, 82, 91, and 66% in the later 2000 epochs, respectively.

3.2. Model evaluation: accuracy, training time, and VRAM consumption

Table 3 shows the overall prediction accuracy, training time per epoch, VRAM consumption, and fluctuations for each model. First, in terms of prediction accuracy, the models showed high prediction accuracy in the order of 3-5 CNN (97.13%), 5-5 CNN (96.55%), MLP-3 (93.10%), 5-3 CNN (91.95%), MLP-2 (86.78%), and Resnet-34 (77.01%). CNN-based models (3-5, 5-3, and 5-5 CNN) showed relatively high prediction accuracy with the same initialization, mini-batch size, and epochs, whereas the relatively shallow MLPs (MLP-1, MLP-2) and CNN-based model with small kernels (Resnet-34) had lower prediction accuracy. MLP-1 with 54675-1000-128-1 layers failed to converge, while the MLP-2 model with 54675-1000-32-1 layers, which is very similar to the MLP-1 model, successfully converged. Among the CNN-based models, the model with relatively long FC layers (five layers) showed higher prediction accuracy than the model with relatively short FC layers (three layers) did.

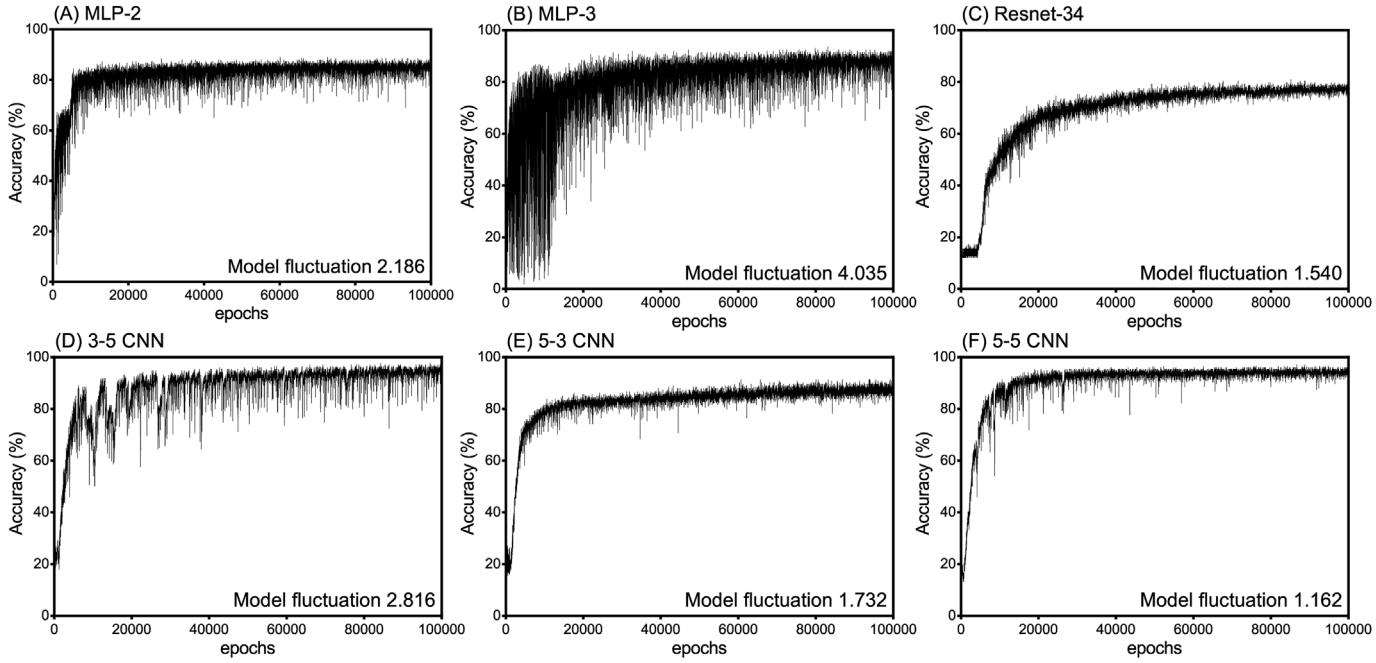

Fig. 1. Epoch-accuracy curves for each model in a total of 100,000 epochs. The fluctuations of each model were quantified as the standard deviation of model prediction accuracy after 50,000 epochs. (A) MLP-2 and (B) MLP-3 showed increasing fluctuations and prediction accuracy as the model became deeper. (D) 3-5 CNN model showed high fluctuations with high prediction accuracy while (E) 5-3 CNN had low fluctuations with low prediction accuracy, and the (F) 5-5 CNN featured low fluctuations and high prediction accuracy. The (C) Resnet-34 model showed low fluctuations and low prediction accuracy. ***Acronyms:*** MLP, Multi-Layered Perceptron; CNN, Convolutional Neural Network.

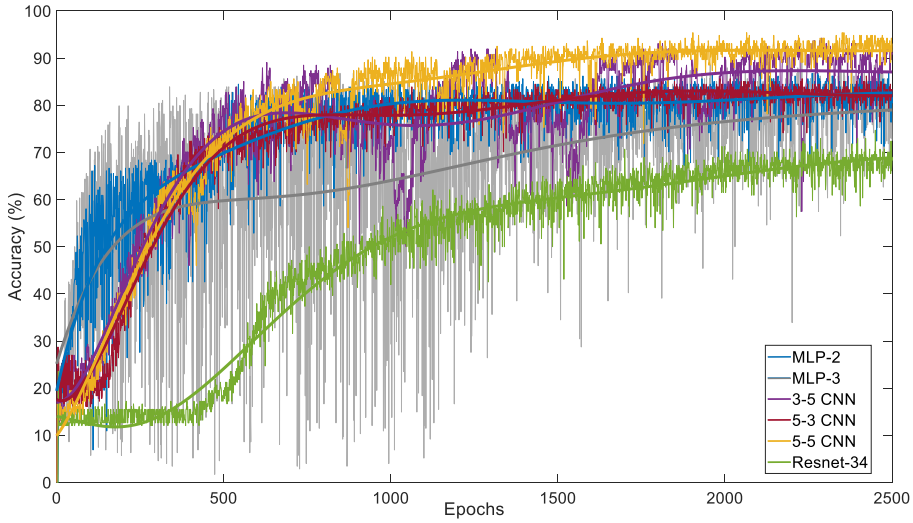

Fig. 2. Combined epoch-accuracy curves and their trend line in early epochs (2500 epochs). The MLP-based models (MLP-2, MLP-3) showed early converge while the CNN-based models showed late convergence. Three CNN models with large kernels (3-5, 5-3, and 5-5 CNN) represented a similar initial convergence state while compared to the CNN model with a small kernel (Resnet-34). The trend line was drawn using the Fourier series curve fitting of the MATLAB curve fitting tool. ***Acronyms:*** MLP, Multi-Layered Perceptron; CNN, Convolutional Neural Network.

Table 3
Overall prediction accuracy, training time per epoch, VRAM consumption, and fluctuations of the models.

Models	Prediction Accuracy (%)	Training time per epoch (sec)	VRAM consumption (MiB)	Model fluctuations (Standard deviation)
MLP-1	0 (0/174)	–	2913	–
MLP-2	86.78 (151/174)	0.106	2913	2.186
MLP-3	93.10 (162/174)	0.447	17419	4.035
3-5 CNN	97.13 (169/174)	1.154	6515	2.816
5-3 CNN	91.95 (160/174)	1.089	5363	1.732
5-5 CNN	96.55 (168/174)	1.056	5491	1.162
Resnet-34	77.01 (134/174)	0.704	9349	1.540

Acronyms: VRAM, video random access memory; MLP, Multi-Layered Perceptron; CNN, Convolutional Neural Network.

The Resnet-34 model showed a much lower prediction accuracy despite its relatively long FC layers (six FC layers).

Furthermore, the training time per epoch increased in the order of MLP-based models, Resnet, and CNN-based models. MLP-based shallow model (MLP 2) required a much shorter training time per epoch (about 0.1 s per epoch), while the MLP-based deep model (MLP-3) required four times longer training time (about 0.4 s per epoch). The Resnet-34 model was approximately 7 times longer than the MLP-based shallow model (approximately 0.7 s per epoch), and the CNN-based models required approximately 10 times longer than the MLP-based shallow model (approximately 1 s per epoch).

Lastly, the VRAM consumption was increased in the order of MLP-2, CNNs, Resnet, and MLP-3. The MLP-based deep model (MLP-3) required the largest VRAM to be trained (17,419 MiB). MLP-based shallow models (MLP-1 and 2) required about a sixth of VRAM (2913 MiB) than the MLP-3 model did. Between CNN-based models (3-5, 5-3, and 5-5 CNNs), 3-5 CNN showed a slightly higher (6515 MiB) but similar VRAM consumption to that of the other models (5363 and 5491 MiB, respectively), while Resnet-34 required nearly double (9349 MiB).

3.3. Model evaluation: comparison of the average of absolute prediction errors

The average and standard deviation of the absolute prediction errors in each model and the difference in the average absolute errors between the models are presented in Table 4. The average and standard deviation of the models' absolute prediction error (MLP-2, MLP-3, 3-5 CNN, 5-3 CNN, 5-5 CNN, and Resnet-34) were 0.0195±0.0263, 0.0219±0.0274, 0.0099±0.0266, 0.0176±0.0357, 0.0091±0.0254, and 0.0343±0.0376, respectively. The MLP-2 was found to have a statistically significantly better prediction result than Resnet-34 ($p < 0.0001$) but worse than the 3-5 CNN and 5-5 CNN ($p < 0.0001$). The MLP-3 also showed a statistically significantly better prediction result than Resnet-34 ($p < 0.001$) did, but worse than the 3-5 CNN and 5-5 CNN ($p < 0.0001$) did. The 3-5 CNN was shown to have a statistically significantly better prediction result than MLP-2, MLP-3, and Resnet-34 ($p < 0.0001$). Furthermore, the 5-3 CNN also had a statistically significantly better prediction result than the Resnet-34 ($p < 0.001$) did, but worse than the 5-5 CNN ($p < 0.05$) had. The 5-5 CNN provided a statistically significantly better prediction result than the MLP-2, MLP-3 ($p < 0.0001$), 5-3 CNN ($p < 0.05$), and Resnet-34 ($p < 0.0001$) did. Lastly, Resnet-34 had a statistically significantly worse prediction result compared to all the other models.

4. Discussion

Deep-learning combined with a large amount of genomic data is an emerging research field and is raising interest from several researchers [25]. However, radiosensitivity analysis based on gene expression using deep learning has not seemed to attract sufficient attention yet. Therefore, in this study, we compared three types of MLP-based models and four types of CNN-based models to investigate the optimal configuration of radiosensitivity prediction using gene expression profiling data.

Among the MLP-based models, the prediction accuracy, training time per epoch, and VRAM consumption were observed to increase as the model became more complicated. Although the training time per epoch did not increase significantly with the model depth, the VRAM consumption increased significantly owing to the MLP's characteristics, which were based on matrix multiplication. MLP-1 with a prediction accuracy of 0% had a relatively large number of nodes (128 nodes) in the last hidden layer compared to the MLP-2 model (32 nodes), which had succeeded in convergence. Therefore, it seems necessary to select an appropriate depth and number of nodes for the model considering the VRAM consumption in the MLP-based model. Among the MLP-based models, MLP-3 exhibited large fluctuations due to its increased depth without any compensation to prevent gradient vanishing problems. However, MLP-3 showed a prediction accuracy of 93.10%, which was lower than that in the 3-5 or 5-5 CNN models but higher than that in the 5-3 CNN model. Considering this, the MLP-based models were not necessarily inferior in performance to the CNN-based models and could have good prediction performance with a considerably deep structure, but seem difficult to use in general clinical applications owing to their resource-dependent characteristics.

Meanwhile, three different types of CNN models with large kernels (3-5, 5-3, 5-5 CNNs) and one CNN-based model with small kernels (Resnet-34) were compared. Since the epoch-accuracy curves exhibited lower fluctuations in most CNN-based models, the convolutional layer in CNN models seemed to contribute to the radiosensitivity prediction model's stability. The fact that the 3-5 CNN model showed high fluctuations in the epoch-accuracy curve due to its relatively shallow convolutional layers also supports this notion. Among the CNN-based models, models with sufficient FC layer length (3-5, 5-5 CNN) showed higher prediction accuracy compared that in the model with a short FC layer (5-3 CNN), even though the convolutional layer of 5-3 CNN was identical to that in 5-5 CNN or deeper than that in 3–5 CNN. Thus, not only the convolutional layer's depth but also the FC layer's depth could be considered important for radiosensitivity prediction model performance. A 3-5 CNN model required a slightly longer training time and more VRAM capacity because of its larger convolution kernel than other CNN-based models. Resnet-34 had a lower prediction accuracy, shorter training time, and higher VRAM consumption with a deep convolutional and FC layer (34 and 6 layers, respectively) compared to the other CNN models, supporting the idea that the CNN-based model with large kernels is more appropriate for gene expression information processing than the CNN-based model with small kernels, considering the characteristics of the different-sized CNN kernels mentioned in the methods section. Therefore, the large kernel CNN-based model with a moderate amount of

Table 4
Average and standard deviation of the absolute error, and difference of average absolute error between each model.

Models	Average ± SD of absolute error	Difference of average absolute error					
		MLP-2	MLP-3	3-5 CNN	5-3 CNN	5-5 CNN	Resnet-34
MLP-2	0.0195 ± 0.0263	–	0.0236^{ns}	0.0096***	0.0019^{ns}	0.0104***	0.0148***
MLP-3	0.0219 ± 0.0274		–	0.0119***	0.0042^{ns}	0.0128***	0.0124**
3-5 CNN	0.0099 ± 0.0266			–	0.0077^{ns}	0.0009^{ns}	0.0243***
5-3 CNN	0.0176 ± 0.0357				–	0.0086*	0.0167**
5-5 CNN	0.0091 ± 0.0254					–	0.0252***
Resnet-34	0.0343 ± 0.0376						–

Acronyms: MLP, Multi-Layered Perceptron; CNN, Convolutional Neural Network; SD, standard deviation.
^{ns}, non-significant. ***p<0.0001; **p<0.001; *p<0.05 from one-way ANOVA multiple comparison.

convolutional and FC layers would be appropriate for radiosensitivity prediction when high accuracy is required but that would also need longer training time.

From what we have discussed so far, 3-5 CNN and 5-5 CNN are considered to be effective based on their high prediction accuracy, relatively small computational resources, and relatively short training times. The 3-5 CNN had better prediction accuracy, which is counterintuitive since the 5-5 CNN had a deeper structure. However, when compared with the 5-3 CNN which showed a higher average absolute error than the two CNN models, there was no statistically significant difference between the 3-5 CNN and 5-3 CNN ($p = 0.2095$) while the 5-5 CNN showed a significantly better prediction result than the 5-3 CNN ($p = 0.0329$). Moreover, statistical analysis using ANOVA revealed no statistically significant difference in the average absolute error between the 3-5 CNN and 5-5 CNN. All of these results indicate that the 5-5 CNN performance could not be inferior to that of the 3-5 CNN. Therefore, combined with the fact that the 5-5 CNN is faster and occupies less VRAM than the 3-5 CNN does, 5-5 CNN is considered the most effective and efficient model in our comparison group.

In their analysis of disease classification in Yu et al. using high-throughput omics datasets containing RNA-sequencing and metabolomics data, they concluded that their MLP-based model outperformed the CNN-based model and classical machine learning models [19]. The CNN-based model discussed in their study was a CNN with a small kernel (kernel size = 3) which performed a similar convolutional operation to that of the Resnet-34 model in our study. The result of their study that MLP outperformed CNN is in accordance with our result that MLP-based models outperformed the Resnet-34 model. This further supports the outcomes of our study that the CNN-based model with large kernels is more appropriate for gene expression processing than a CNN-based model with small kernels.

A limitation of this study is that, besides the Resnet-34 model, various kinds of small kernel-based novel deep neural network architectures such as Densenet were not included in the comparison group [26]. However, we selected the Resnet-34 model as a representative small kernel-based novel deep neural network architecture. Had the prediction performance of Resnet-34 been significantly better than the other models or at least similar to that of the other MLP and CNN-based models, it would have been necessary to compare the various kinds of models with the small-sized kernel. However, the Resnet-34 model was found to be less suitable for predicting radiosensitivity using gene expression. Therefore, we did not extend this study to other types of small kernel-based CNN models.

Despite the limitation, this study had its own strengths as this was the first study comparing various types of deep-learning models predicting *in vitro* radiosensitivity using gene expression profiling. Herein, several factors such as model training time, VRAM occupation, and model performance according to the model architecture were compared with the same data under the same environment. Additionally, it was first suggested in this study that CNN-based models with large kernels had better performance than CNN-based models with small kernels when dealing with a one-dimensional gene expression vector, unlike that in the widely used image processing deep-learning models.

To sum up, the result of this study suggests that when predicting cellular radiosensitivity from gene expression profiling, the CNN with appropriate depth would be beneficial in terms of both computational efficiency and prediction accuracy. Since the appropriate model type would vary greatly depending on the characteristics and distribution of the data when it comes to the other types of input data rather than the gene expression, it cannot be said that the CNN would be generally appropriate for other types of data with the results of this study. However, it can be said that the data with similar characteristics to gene expression (locally incoherent, arbitrary distributed, one-dimensional long vector) would have a good result with the large kernel-based CNN-based models. Therefore, when readers need to build their own radiosensitivity prediction model, they can refer to the results of this study and the characteristics of their dataset to determine the appropriate type and depth of the model.

5. Conclusion

The performances of various types of deep-learning-based *in vitro* radiosensitivity prediction models were compared in this study. MLP-based models represented fast initial convergence, short training time per epoch, high training fluctuations, and a significant increase in the VRAM capacity requirement as the model deepened. However, CNN-based models showed relatively slow initial convergence, long training time per epoch, very low training fluctuations, and a relatively small increase in VRAM capacity requirement as the model deepened. Our results suggest that each type of deep-learning model has its own characteristics, and therefore, can be optimized by the users to predict *in vitro* radiosensitivity according to their own situation or goals. A CNN-based model with moderate depth such as 5-5 CNN would be appropriate when the prediction accuracy is important, and an MLP or shallow CNN-based model could be recommended when the training resources or time are limited.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (KRF) funded by the Ministry of Education (NRF-2018R1D1A1B07049228).

References

[1] S.D. Bouffler, Evidence for variation in human radiosensitivity and its potential impact on radiological protection, Ann. ICRP 45 (2016) 280–289.
[2] D.G. Hirst, T. Robson, Molecular biology: the key to personalised treatment in radiation oncology? Br. J. Radiol. 83 (2010) 723–728.
[3] H.S. Kim, S.C. Kim, S.J. Kim, C.H. Park, H.-C. Jeung, Y.B. Kim, et al., Identification of a radiosensitivity signature using integrative metaanalysis of published microarray data for NCI-60 cancer cells, BMC Genom. 13 (2012) 348.
[4] Q.E. He, Y.F. Tong, Z. Ye, L.X. Gao, Y.Z. Zhang, L. Wang, et al., A multiple genomic data fused SF2 prediction model, signature identification, and gene regulatory network inference for personalized radiotherapy, Technol. Cancer Res. Treat. 19 (2020), 1533033820909112.
[5] J.F. Torres-Roca, S. Eschrich, H. Zhao, G. Bloom, J. Sung, S. McCarthy, et al., Prediction of radiation sensitivity using a gene expression classifier, Cancer Res 65 (2005) 7169–7176.
[6] C. Zhang, L. Girard, A. Das, S. Chen, G. Zheng, K. Song, Nonlinear quantitative radiation sensitivity prediction model based on NCI-60 cancer cell lines, Sci. World J. 2014 (2014) 903602.
[7] E. Kim, Y. Chung, Feasibility study of deep learning based radiosensitivity prediction model of National Cancer Institute-60 cell lines using gene expression, Nucl. Eng. Technol. (2021).
[8] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012) 1097–1105.
[9] F. Murtagh, Multilayer perceptrons for classification and regression, Neurocomputing 2 (1991) 183–197.
[10] H. Sak, A.W. Senior, F. Beaufays, Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling, 2014.
[11] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale

Image Recognition, 2014 arXiv preprint arXiv:14091556.
[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, et al., Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[13] R.G. Bristow, P.A. Hardy, R.P. Hill, Comparison between in vitro radiosensitivity and in vivo radioresponse of murine tumor cell lines. I: parameters of in vitro radiosensitivity and endogenous cellular glutathione levels, Int. J. Radiat. Oncol. Biol. Phys. 18 (1990) 133–145.
[14] C.M. West, Invited review: intrinsic radiosensitivity as a predictor of patient response to radiotherapy, Br. J. Radiol. 68 (1995) 827–837.
[15] S. Eschrich, H. Zhang, H. Zhao, D. Boulware, J.-H. Lee, G. Bloom, et al., Systems biology modeling of the radiation sensitivity network: a biomarker discovery platform, Int. J. Radiat. Oncol. Biol. Phys. 75 (2009) 497–505.
[16] T.D. Pfister, W.C. Reinhold, K. Agama, S. Gupta, S.A. Khin, R.J. Kinders, et al., Topoisomerase I levels in the NCI-60 cancer cell line panel determined by validated ELISA and microarray analysis and correlation with indenoisoquinoline sensitivity, Mol. Cancer Therapeut. 8 (2009) 1878–1884.
[17] M.W. Gardner, S.R. Dorling, Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences, Atmos. Environ. 32 (1998) 2627–2636.
[18] J.M. Johnson, T.M. Khoshgoftaar, Survey on deep learning with class imbalance, J. Big Data 6 (2019) 27.
[19] H. Yu, D.C. Samuels, Y-y Zhao, Y. Guo, Architectures and accuracy of artificial neural network for disease classification from omics data, BMC Genom. 20 (2019) 167.
[20] A. Sharma, E. Vans, D. Shigemizu, K.A. Boroevich, T. Tsunoda, DeepInsight: a methodology to transform a non-image data to an image for convolution neural network architecture, Sci. Rep. 9 (2019) 11399.
[21] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[22] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D.J. Inman, 1D convolutional neural networks and applications: a survey, Mech. Syst. Signal Process. 151 (2021) 107398.
[23] T. Fushiki, Estimation of prediction error by using K-fold cross-validation, Stat. Comput. 21 (2011) 137–146.
[24] L.J. Peters, The ESTRO Regaud lecture. Inherent radiosensitivity of tumor and normal tissue cells as a predictor of human tumor response, Radiother. Oncol. 17 (1990) 177–190.
[25] S. Min, B. Lee, S. Yoon, Deep learning in bioinformatics, Briefings Bioinf. 18 (2016) 851–869.
[26] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.