Informatics in Medicine Unlocked 18 (2020) 100282

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: http://www.elsevier.com/locate/imu

Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images

Mohammad Ali Kadampur *, Sulaiman Al Riyaee

Department of Information Management, College of Computer & Information Sciences, Al-Imam Muhammed Ibn Saud Islamic University, Riyadh, India

ARTICLE INFO

Keywords:
Deep learning
AI
Model driven architecture
Deep cognition studio
CNN
Cancer
Image classification

ABSTRACT

Background: Skin cancer is a common form of cancer, and early detection increases the survival rate.
Objective: To build deep learning models to classify dermal cell images and detect skin cancer.
Methods: A model-driven architecture in the cloud, that uses deep learning algorithms in its core implementations, is used to construct models that assist in predicting skin cancer with improved accuracy. The study illustrates the method of building models and applying them to classify dermal cell images.
Results: The deep learning models built here are tested on standard datasets, and the metric area under the curve of 99.77% was observed.
Conclusions: A practitioner can use the model-driven architecture and quickly build the deep learning models to predict skin cancer.

1. Introduction

Skin cancer is a dangerous and widespread disease [1] Each year there are approximately 5.4 million new cases of skin cancer are recorded in USA alone [2,4].The global statistics are equally alarming [3][5]. Recent reports show that from 2008 to 2018, there has been a 53% increase in new melanoma cases diagnosed annually [1,4]. The mortality rate of this disease is expected to rise in the next decade. The survival rate is less than 14% [8–10] if diagnosed in later stages. However, if the skin cancer is detected at early stages then the survival rate is nearly 97% [3]. This demands the early detection of skin cancer. This paper addresses the issue of early diagnosis, with improved accuracy.

It is found that [18,22,23] a skilled dermatologist usually follows a series of steps, starting with naked eye observation of suspected lesions, then dermoscopy (magnifying lesions microscopically) and followed by biopsy. This would consume time and the patient may advance to later stages. Moreover accurate diagnosis is subjective, depending on the skill of the clinician. It is found that the best dermatologist has an accuracy of less than 80% in correctly diagnosing the skin cancer [6]. Adding to these difficulties, there are not many skilled dermatologists available globally in public healthcare.

In order to diagnose skin cancer speedily at the earliest stage and solve some of the aforementioned problems, there has been extensive research solutions by developing computer image analysis algorithms [[26]]. The majority of these algorithmic solutions were parametric, meaning that they required data to be normally distributed. As the nature of data cannot be controlled, these methods would be insufficient to accurately diagnose the disease. However, non-parametric solutions do not rely on the constraint that the data is in normal distribution form.

In this paper, an augmented assistance to the dermatologist is provided using deep learning. The essence of the approach is that a computer is trained to determine the problem by analyzing the skin cancer images. The novelty of the presentation is that the computer model can be developed without having any programming knowledge. The average accuracy of diagnosis using this model is found to be approximately 98.89% and the best is 100%. The machine assisted diagnosis presented here overcomes the problem of delay, accuracy, and scarcity of dermatologists in public health.

Studies suggest that in the area of skin cancer detection and image classification, there exists a plethora of research papers. A detailed survey of these methods is available in Refs. [1,7,8]. Each of these papers [1,7,8] used the then available state-of-art methods and claim performance improvements. The popular methods used for image classifications vary from application of decision tree algorithms [9,10] Bayesian classifiers [11–13], support vector machines [8,14], to a variety of Artificial Intelligence based approaches [15,16]. However a

* Corresponding author.
E-mail addresses: mkadampur@imamu.edu.sa (M.A. Kadampur), saalriyaee@imamu.edu.sa (S. Al Riyaee).
URL: http://www.imamu.edu.sa (M.A. Kadampur).

https://doi.org/10.1016/j.imu.2019.100282
Received 5 August 2019; Received in revised form 30 November 2019; Accepted 5 December 2019
Available online 15 December 2019

common theme in all these papers is that they are made to appear as a job of specialists in the domain of computers and software engineering. An essential level of programming expertise in computer languages like Java, R, and Python is indeed required to build any of these diagnostic models [17–19]. In this paper, we address methods of developing deep learning based image classification models for identification of skin cancer, without having prior programming knowledge. The main objectives of this paper are:

1 To enable researchers and practitioners to develop deep learning models by simple plug and play art.
2 To classify the cell images and identify Cancer with an improved degree of accuracy using deep learning.

2. Related works

The research of skin cancer detection based on image analysis has advanced significantly over the years. Many different techniques have been tried [1]. The International Skin Imaging Collaboration (ISIC) event of 2018 has become a de facto benchmark in skin cancer detection by hosting a challenge contest. It is also reported that a mobile app can be used to detect skin cancer. In all these efforts researchers have tried to improve the accuracy of diagnosis by employing different classification algorithms and techniques. Image classification took to new bounds when convolutional neural network (CNN) structure was introduced by Fukushima (1988) and later Le-Cunn (1990). They used CNNs for image classification. CNNsbasically mimic the human visual cognition system and are considered to be the best state-of-the-art methods for image classification. Though there is a plethora of literature available on image classification, we limit our review of the literature to deep learning methods for skin cancer images.

The first breakthrough on skin cancer classification by a pre-trained GoogLeNet Inception V3 CNN model came from Esteva et al. [20]. They used 129,450 clinical skin cancer images including 3,374 dermatoscopic images. The reported accuracy of classification is 72.1 ± 0.9. In 2016, Yu et al. [21] developed a CNN with over 50 layers on the ISBI 2016 challenge dataset for the classification of malignant melanoma cancer. The best classification accuracy reported in this challenge was 85.5%. In 2018, Haenssle et al. [22] utilized a deep convolutional neural network to classify a binary diagnostic category of dermatoscopy melanocytic images, and reported 86.6% sensitivity and specificity for classification. A multiclass classification using ECOC SVM and deep learning CNN was developed by Dorj et al. in Ref. [23]. The approach was to use ECOC SVM with pre-trained AlexNet Deep Learning CNN and classify multiclass data. An average accuracy of 95.1% is reported in this work. In Ref. [24] Han et al. have used a deep convolutional neural network to classify the clinical images of 12 skin diseases. The reported best classification accuracy instance varies between 96.0% ± 1%. A detailed review of classifiers is not the scope of this paper; however a systematic review of deep learning classifiers is found in Ref. [25].

3. Models & algorithms

Using Deep Learning Studio (DLS), we can build intelligent data discovery models. We can simply drag and drop suitable components to build an appropriate model. The DLS comes with a desktop version and a cloud version. It supports multi GPU training up to 4 GPUs in its community edition and additional GPUs in its enterprise edition. Here in this work we have used the cloud version with a server GPU-XEON-E5-8 GB. DLS has UI components for project creation, data upload, model building, training the model, testing the model, and code generation in its architecture. Different deep learning algorithms can be selected just by dragging the relevant dashboard component. DLS enables easy building of deep learning models. An example model build is shown in Fig. 1.

The steps for constructing a deep learning model are listed below:

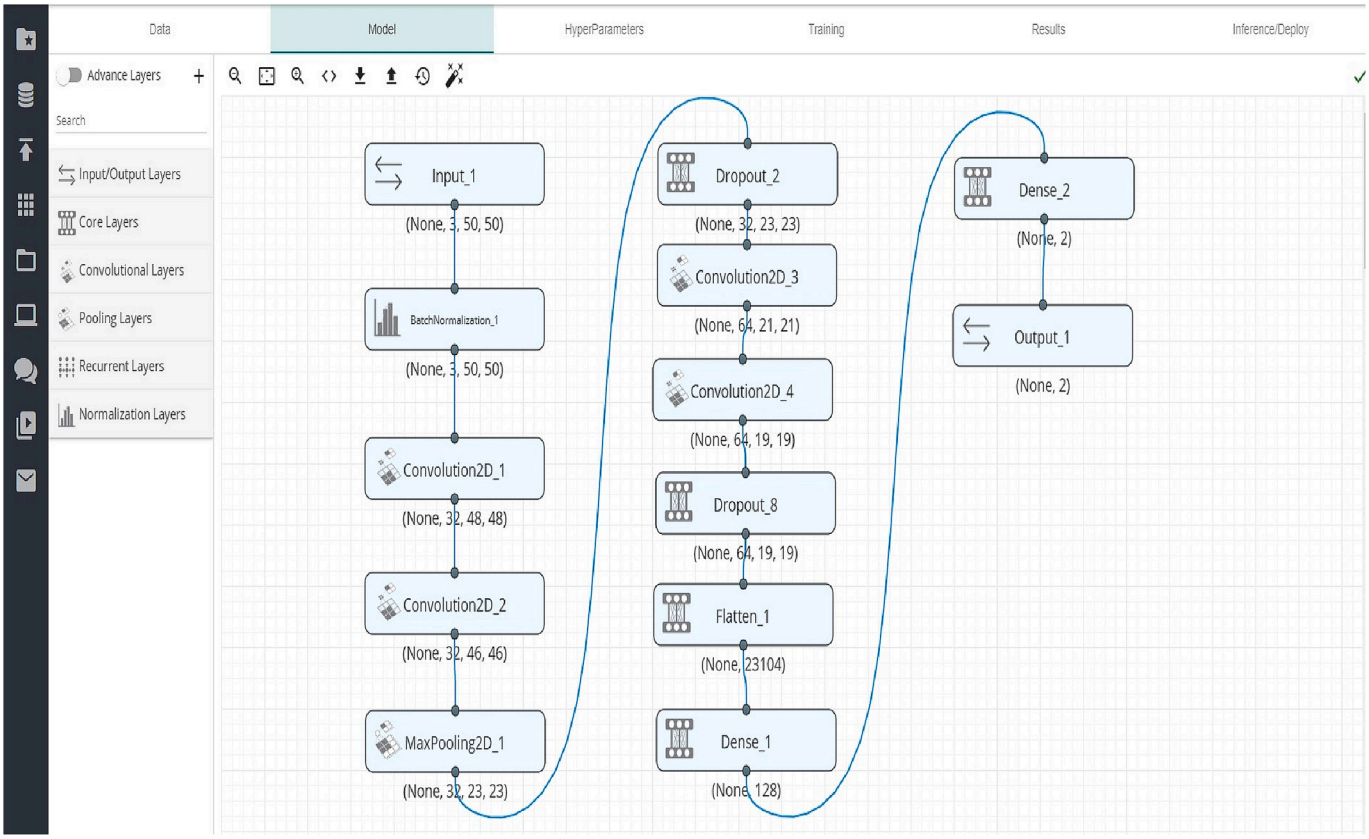


Fig. 1. An example deep learning model as built using DLS.

1. Select the input block, drag and drop it into the work space.
2. Perform some normalization (if required, drag & drop related blocks)
3. Apply Convolution blocks (Choose activation ReLU preferably)
4. Apply Pooling Blocks
5. Apply Dropout block
6. Repeat 3, 4, & 5 as many times as optimally needed
7. Use some flattening block
8. Use some core layers (Dense blocks preferably). Ensure here that final core layer before the output block has the proper output value parameter as per the data (output) categories.
9. Apply the output block.
10. Set the model parameters (Hyper parameter setting).
11. Finally, ensure that the Model construction checks Green "OK" else fine-tune your selections.

Fig. 2 summarizes the above steps of building deep learning models.

Setting up of model parameters is critical to the performance of the model. There is no rule of thumb for setting these parameters. The general guideline for parameter setting is that which has worked well for others, just try that. ReLU is the preferred activation function for inner ConvNets. Filter size is usually 3×3 and number of filters used may vary in terms of multiples of two. A dropout percentage of 0.25–0.50 is applied to speed up the training. The user must check that the number of outputs in the output layer is proper. If the training accuracy is less, then the epochs and convolution layers should be increased. A pooling layer down-samples the volume along the spatial dimensions by a certain factor. This reduces the size of the representation and the number of parameters which makes the model more efficient and prevents overfitting. Use the setting to prevent model overfitting.

3.1. Methodology

The DLS, Model Driven Architecture Tool provides Neural Network Modeling components as a stack of drag-drop & develop art. The sequence of important general steps involved as a research methodology in this paper are as follows:

1 Data Preparation
2 Creating a project and loading the dataset
3 Building a deep learning classifier
4 Tuning the model
5 Checking result
6 Drawing Inferences
7 Code Access
8 Deploying the model as a REST API.

Data preparation is a preprocessing phase in which the raw data is processed by cleaning and formatting. The DLS accepts the data in the form of comma separated variables (.csv) files. We have used the app [26] (Irfan view) for bulk naming and customizing image features. Creating the project and the remaining steps follow with a quick familiarity of the dashboard of DLS. Deploying the model as a REST API is left as future work. We experimented with the following set of models in our research.

3.2. Our tested models

MODEL1:INPUT(), BATCHNORM(3), CONV2D11(32,3) RELU, CONV2D2(32,3) RELU, CONV2D3(64,3) RELU, CONV2D4(64,3) RELU, MAXPOOL1(2), CONV2D5(128,3) RELU, CONV2D6(128,3) RELU, CONV2D7(256,3) RELU, CONV2D8(256,3) RELU, MAXPOOL2(2), FLATTEN(), DROP(0.3), DENSE(2) RELU, OUTPUT().

MODEL2:INPUT(), BATCHNORM(0),CONV2D1(32,3) RELU, CONV2D2(32,3) RELU, MAXPOOL(8), CONV2D3(64,3) RELU, CONV2D4(64,3) RELU, 4*(RESNET50(TRUE,3,224,224)) DROP(0.3), FLATTEN(), DENSE(128) RELU, DENSE(8) SOFTMAX, OUTPUT().

MODEL3:INPUT(), BATCHNORM(0),CONV2D1(32,3) RELU, CONV2D2(32,3) RELU, MAXPOOL(8), CONV2D3(64,3) RELU, CONV2D4(64,3) RELU, 4*(SQUEEZNET(TRUE,3,224,224)) DROP(0.3), FLATTEN(), DENSE(128) RELU, DENSE(8) SOFTMAX, OUTPUT().

MODEL4:INPUT(), BATCHNORM(0), CONV2D1(32,3) RELU, CONV2D2(32,3) RELU, MAXPOOL(8), CONV2D3(64,3)RELU, CONV2D4(64,3) RELU,4*(DENSENET201(TRUE,3,224,224)) DROP (0.3),FLATTEN(),DENSE(128) RELU, DENSE(8) SOFTMAX, OUTPUT().

MODEL5: INPUT(), BATCHNORM(0), CONV2D 1(32,3)RELU, CONV2D 2(32,3) RELU, MAXPOOL(8), CONV2D3(64,3) RELU, CONV2D4(64,3) RELU, 4*(INCEPTIONV3(TRUE,3,224,224))DROP (0.3),FLATTEN(),DENSE(128) RELU, DENSE(8) SOFTMAX, OUTPUT().

MODEL1 above was built using 2D convolutional networks as an initial experimentation and tested on two datasets. The ROC analysis are presented for this model on the two datasets as shown in Figs. 4 and 5. The remaining models use pre-trained networks, and results of experiments with these pre-trained models are presented in Table 1.

4. Experiments & results

The pre-trained models above (MODEL2-MODEL5) were trained and tested using the HAM10000 dataset, which is a benchmark public database for machine learning. The data was divided into 80:10:10:: Training:Validation:Testing ratios. The results of each of the above models is presented in Table 1. The Fig. 3 shows a training instance.

4.1. Training

In order to start training the model, one needs to start the compute instance and push the red power button. The power button will go green and the training instance using the remote GPU will start. The dashboard shows the progress while training, accuracy changes, and completion of epochs. Training will stop if there is any error in the run. If everything is correct, training will stop after the stopping criteria.

5. Discussion

The models built using DLS were tested against the benchmark machine learning databases (HAM 10000) and the results are very promising. A higher value of ROC suggests that the classifier model rarely fails to diagnose cancer as cancer, and non-cancer as non-cancer. Among the pre-trained models, squeeznet, densenet, and inception v3 models have a higher ROC as compared to the resnet model. These observations suggest that the models built using DLS produce better results than many of the findings in the related work. The model building is easy with DLS and effective.

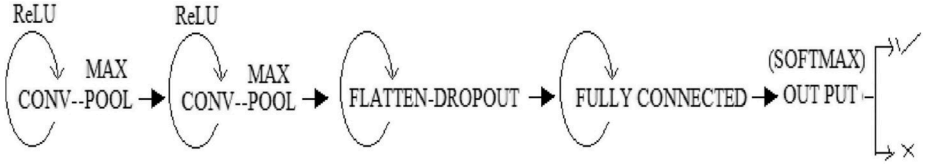


Fig. 2. Constructing a deep learning model: steps.

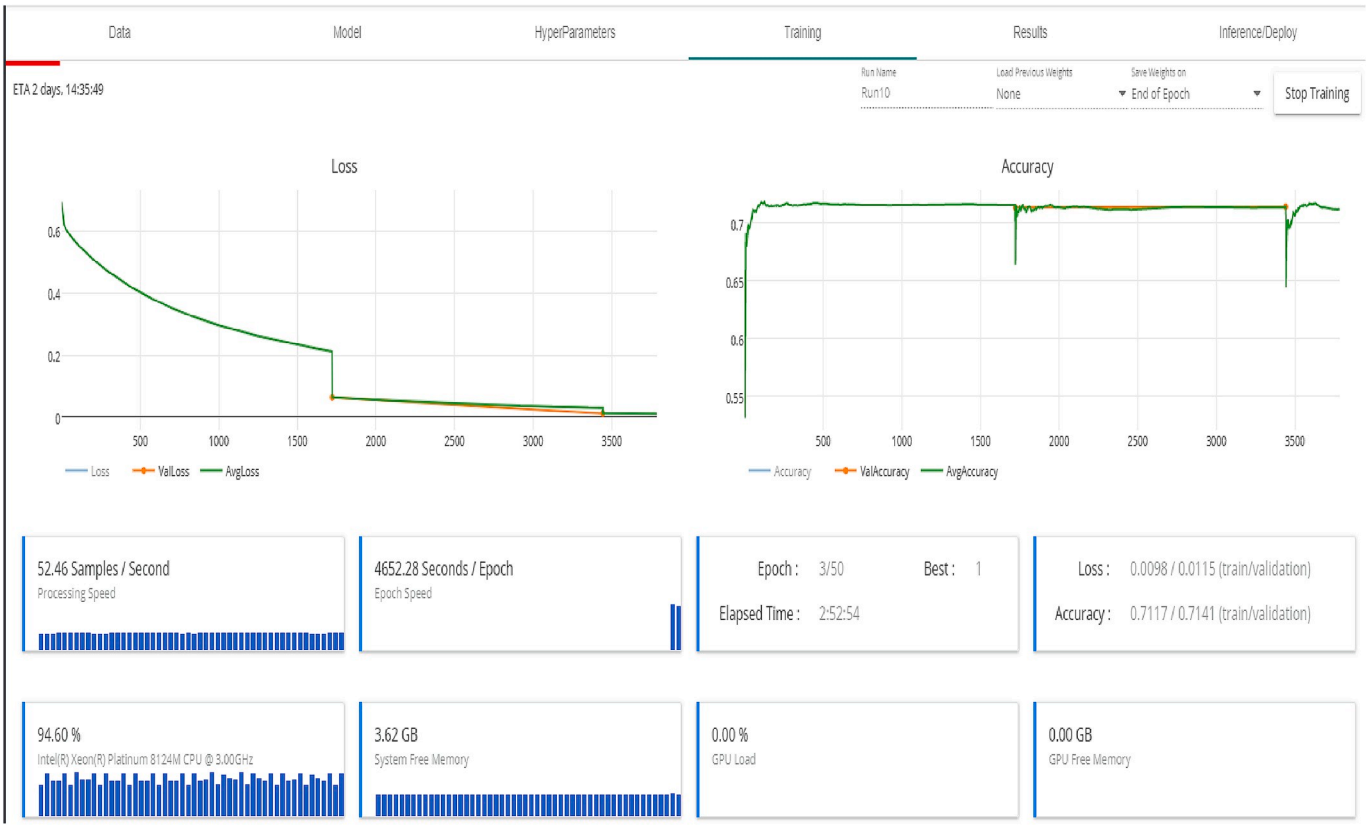


Fig. 3. An instance of training the model.

Variable	label
Classification variable	predictions

Sample size	55043
Positive group [a]	15849 (28.79%)
Negative group [b]	39194 (71.21%)

[a] predictions = 1
[b] predictions = 0

Disease prevalence (%)	28.8

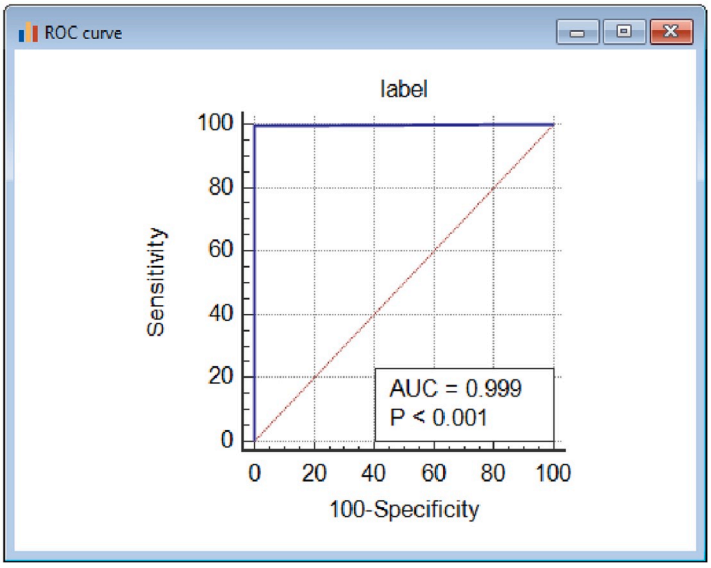
Area under the ROC curve (AUC)

Area under the ROC curve (AUC)	0.999
Standard Error [a]	0.000106
95% Confidence interval [b]	0.999 to 1.000
z statistic	4720.856
Significance level P (Area=0.5)	<0.0001

[a] DeLong et al., 1988
[b] Binomial exact

Youden index

Youden index J	0.9989
Associated criterion	>0
Sensitivity	99.96
Specificity	99.93



Criterion values and coordinates of the ROC curve [Hide]

Criterion	Sensitivity	95% CI	Specificity	95% CI	+LR	-LR	+PV	95% CI	-PV	95% CI
≥0	100.00	100.0 - 100.0	0.00	0.0 - 0.009	1.00		28.8	28.8 - 28.8		
>0	99.96	99.9 - 100.0	99.93	99.9 - 100.0	1506.80	0.00044	99.8	99.8 - 99.9	100.0	100.0 - 100.0
>1	0.00	0.0 - 0.02	100.00	100.0 - 100.0		1.00			71.2	71.2 - 71.2

Fig. 4. ROC analysis of MODEL1 on dimizC dataset.

Variable	label
Classification variable	predictions

Sample size	55043
Positive group [a]	12163 (22.10%)
Negative group [b]	42880 (77.90%)

[a] predictions = 1
[b] predictions = 0

Disease prevalence (%)	22.1

Area under the ROC curve (AUC)

Area under the ROC curve (AUC)	0.799
Standard Error [a]	0.00214
95% Confidence interval [b]	0.795 to 0.802
z statistic	139.501
Significance level P (Area=0.5)	<0.0001

[a] DeLong et al., 1988
[b] Binomial exact

Youden index

Youden index J	0.5974
Associated criterion	>0
Sensitivity	75.37
Specificity	84.37

Criterion values and coordinates of the ROC curve [Hide]

Criterion	Sensitivity	95% CI	Specificity	95% CI	+LR	-LR	+PV	95% CI	-PV	95% CI
≥0	100.00	100.0 - 100.0	0.00	0.0 - 0.009	1.00		22.1	22.1 - 22.1		
>0	75.37	74.6 - 76.1	84.37	84.0 - 84.7	4.82	0.29	57.8	57.2 - 58.4	92.4	92.1 - 92.6
>1	0.00	0.0 - 0.03	100.00	100.0 - 100.0		1.00			77.9	77.9 - 77.9

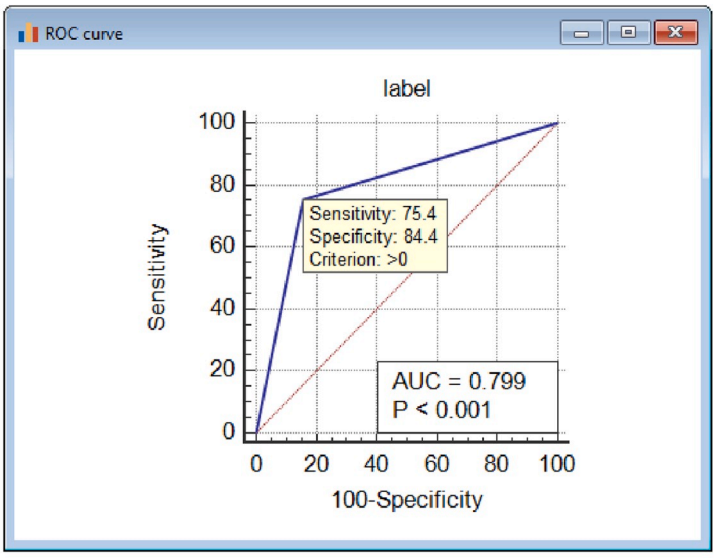


Fig. 5. ROC analysis of MODEL1 on Kaggle dataset.

Table 1
Metric values of pre-trained deep learning classifiers.

Model	(Precision)	(F1-Score)	(ROC AUC)
MODEL2(resnet)	94.24	94.22	98.61
MODEL3(squeeznet)	97.40	94.57	99.77
MODEL4(densenet)	97.51	96.27	99.09
MODEL5(inceptionv3)	98.19	95.74	99.23

6. Conclusion & future work

This paper introduced a tool in which a non-programming background person can develop complex deep learning models. It opened the options of flexibility in designing deep learning classifiers by hinting at the general procedures and the looping patterns in the development of deep learning models (Fig. 2).

The paper used Deep Learning Studio a Model Driven Architecture for Deep learning. The paper introduced the features of the DLS tool and demonstrated the process of constructing a Deep Learning Model using the tool. The paper narrated the process of data preparation using dermal cell images and its test application into the DLS model to detect cancer cells. The DLS models achieved an AUC of 99.77% in detecting cancer cells from the images of cancer cells.

The paper pointed at the provision of obtaining the programming code for the model for further exploration by the programming specialist. The provision to download the trained model and develop enterprise level applications is the best seed level research that this paper observes for the future work. Finally, the paper achieved the set objectives of the introductory section.

Ethical Statement for informatics in medicine unlocked

Hereby, I Dr. Mohammad Ali Kadampur ensure that for the manuscript "Deep Learning: Detecting Skin Cancer by Classifying Dermal Cell Images Using a Model Driven Architecture in the Cloud" the following is fulfilled:

1) This material is the authors' own original work, which has not been previously published elsewhere.

2) The paper is not currently being considered for publication elsewhere.

3) The paper reflects the authors' own research and analysis in a truthful and complete manner.

4) The paper properly credits the meaningful contributions of co-authors and co-researchers.

5) The results are appropriately placed in the context of prior and existing research.

6) All sources used are properly disclosed (correct citation). Literally copying of text must be indicated as such by using quotation marks and giving proper reference.

7) All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

The violation of the Ethical Statement rules may result in severe consequences.

To verify originality, your article may be checked by the originality detection software iThenticate. See also http://www.elsevier.com/editors/plagdetect.

I agree with the above statements and declare that this submission follows the policies of Solid State Ionics as outlined in the Guide for Authors and in the Ethical Statement.

Funding received

None.

Declaration of competing interest

All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.

This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.

The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript.

The following authors have affiliations with organizations with direct or indirect financial interest in the subject matter discussed in the manuscript:

Acknowledgements

We thank DeepCognition for creating the Deep Learning program used in this study.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.imu.2019.100282.

References

[1] Masood A, Al-Jumaily AA. Review article computer aided diagnostic support system for skin cancer: a review of techniques and algorithms. Int J Biomed Imaging 2013:1–22. https://doi.org/10.1155/2013/323268.
[2] Siegel R, Naishadham D, Jamal A. Cancer statistics, 2012. Cancer J. Clin. 2012: 10–29. https://doi.org/10.3322/caac.20138.
[3] Siegel R, Miller KD, Jamal A. Cancer statistics, 2018. Cancer J. Clin. 2018:7–30. https://doi.org/10.3322/caac.21442.
[4] Esteva1 A, Kuprel1 B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S. Dermatologist-level classification of skin cancer with deep neural networks. Nature 2017:115–26. https://doi.org/10.1038/nature21056.
[5] W. H. Organization. WHO. [link]. https://www.who.int/uv/faq/skincancer/en/index1.html.
[6] Marks R. Epidemiology of melanoma, clinical and experimental dermatology. Clin Dermatol 2000:459–63. https://doi.org/10.1046/j.1365-2230.2000.00693.x.
[7] Kittler H, Pehamberger H, Wolf K, Binder MJTIO. Diagnostic accuracy of dermoscopy. Clin Dermatol 2002:159–65. https://doi.org/10.1046/j.1365-2230.2000.00693.x.
[8] Erkol B, Moss RH, Stanley RJ, Stoecker WV, Hvatum E. Au- tomatic lesion boundary detection in dermoscopy images using gradient vector flow snakes. Skin Res Technol 2005:17–26.
[9] Zorman MMM, Kokol SP, Malcic I. The limitations of decision trees and automatic learning in real world medical decision making. J Med Syst 1997:403–15.
[10] Gehrke J. Classification and regression trees. IGI global; 2009.
[11] Larsen K. Generalized näıve bayes classifier. In: ACM SIGKDD explorations news letter, 7 of 1. USA: ACM; 2005. p. 76–81. T.
[12] Neal RM. Springer hidelburg Germany. 1996.
[13] Ruiz D, Berenguer V, Soriano A, Sanchez B. A decision support system for the diagnosis of melanoma: a comparative approach. Expert Syst Appl Nov-Dec 2011; 38(12):15217–23.
[14] Gilmore S, Hofmann-Wellenhof R, Soyer HP. A support vectormachine for decision support inmelanoma recognition. Experimental Dermatogra- phy; 2010. p. 830–5.
[15] Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den G. Mastering the game of go with deep neural networks and tree search. Nature 2016:484–9. https://doi.org/10.1038/nature16961.
[16] Codella1 N, Cai1 J, Abedini M, Garnavi R, Halpern A, Smith1 JR. Deep learning, sparse coding, and SVM for melanoma recog- nition in dermoscopy images. Springe LNCS; 2015. https://doi.org/10.1007/978-3-319-24888-2015.
[17] Souza AMFF, Bostian FMS, Sugomori KY. Deep learning practical neural networks with Java. third ed. Pack Publishing; 2017.
[18] Ramasubramaniam K, Singh A. Machine learning using R. second ed. Acta Press; 2018.
[19] Chollet F. Deep learning with Python. second ed. Manning Publications Company; 2017.
[20] Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S. Dermatologist-level classification of skin cancer with deep neural networks. Nature 2017:115. https://doi.org/10.1038/nature21056.
[21] Yu L, Chen H, Dou Q, Qin J, Heng P-A. Automated melanoma recog- nition in dermoscopy images via very deep residual networks. IEEE Trans Med Imaging 2017:994–1004. https://doi.org/10.1109/TMI.2016.2642839.
[22] Haenssle HA, Fink C, Schneiderbauera R, Toberer F, Buhl T, Blum A, Kalloo A, Hassen ABH, Thomas L, Enk A, Uhlmann L. Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 derma- tologists. Ann Oncol 2018: 1836–42. https://doi.org/10.1093/annonc/mdy166.
[23] Dorj U-O, Lee K-K, Choi J-Y, Lee1 M. The skin cancer classification using deep convolutional neural network. Multimedia Tools and Applica- tions; 2018. p. 9909–24. https://doi.org/10.1007/s11042-018-5714-1.
[24] Han SS, Kim MS, Lim W, Park GH, Park I, Chang SE. Classifica- tion of the clinical images for benign and malignant cutaneous tumors us- ing a deep learning algorithm. J Investig Dermatol 2018:1529–38. https://doi.org/10.1016/j.jid.2018.01.028.
[25] Brinker1 TJ, Hekler1 A, Utika JS, Grabe N, Schadendorf D, Klode J, Berking C, Steeb T, von Kalle1 AHEC. Skin cancer classification using convolutional neural networks: systematic review. J Med Internet Res 2018:e11936. https://doi.org/10.2196/11936.
[26] IrfanView. [link]. https://www.irfanview.net/.