Journal homepage www.jzs.univsul.edu.iq
Journal of Zankoy Sulaimani
Part-A- (Pure and Applied Sciences)

A Deep Learning Technique For Lung Nodule Classification Based on False Positive Reduction

Hunar Abubakir Ahmed[1] & Sozan Abdulla Mahmood[2]

1 College of Basic Education - Raparin University, Sulaimani-Iraq
Email: hunar.abubakir@raparinuni.org
2 College of Science – Sulaimani university , Sulaimani-Iraq
Email: sozan.mahmood@univsul.edu.iq

Article info

Original: 14 December 2018
Revised: 14 January 2019
Accepted: 17 February 2019
Published online: 20 June 2019

Key Words: *lung cancer, deep learning, convolutional neural network, LUNA16, classification.*

Abstract

Cancer is of the major reasons of human death universally, one of the deadliest types of cancer is lung cancer, that causes the highest rate of the dead in both genders combined. Detecting lung cancer in early stage does not guarantee the survive of the patient's life but it can reduce the mortality ratio by a high degree, early detection mainly includes screening unhealthy human's lung using most valuable imaging modality which is CT scan. Classifying nodules in lung CT images adopting an automatic computer system become a necessary task due to a huge number of situations every day to help human expert's in decision making procedure. Over the past few years, a numerous computer system is presented, each done a certain task such as detecting, segmenting, and classifying lung tumors using dissimilar algorithms. The objective of this study is to design an automated lung nodule classification system using two distinct deep learning architectures which are Network In Network (NIN) and standard Convolution Neural Network (CNN). The two models are trained and tested using 13,500 2D cubes around the nodule location that obtained from LUNA16 dataset, the database consists of 888 3D CT scans with annotation file determined a nodule position in every scan. The models are trained with a diverse cube size and hyperparameters in order to develop a high-performance structure for each model. The experimental results showed that best achieved scores for NIN are accuracy 90%, precision 99%, recall 68%, and false positive rate 0.06%, but for the typical CNN are accuracy 90%, precision 85%, recall 85%, and false positive rate 7.52%.

Introduction

Lung cancer is a harmful disease that occurs in human lungs and most probably causes the loss of life of an injured person if it's detected in advanced stages. According to global cancer statistics 2018, it could be classified as a primary reason of human death among all cancer types in both men's and woman's combined (18.4% of the overall cancer deaths) [1]. Studies have disclosed that early stage detection can reduce the mortality rate, identifying in an early stage mostly includes X-ray, Computed Tomography (CT), Low-Dose Computed Tomography (LDCT), and Magnetic Resonance Imaging (MRI), however CT scan is among the more common and effective used ones, CT can detect what is called lung (pulmonary) nodules, that is splatter inside lungs that appear in different sizes and shapes. A nodule on the lungs could be malicious or generous. In recent decades, diagnosing CT scans are mainly interpreted manually by a radiologist that is a time-consuming task because of a huge number of cancer cases each day, also there is a limitation on the detection ability because of human's subjectivity, and there can be a major difference between various readers. For that, computer expertise thought about using a computer system as an effective tool to detect and analysis lung tumors, that will be a great assist for physicians workflow [2][3].

In late years, researchers performing on totally different medical imaging tasks such as segmentation, detection, and classification, via applying several techniques begin with low pixel processing until came to Machine Learning (ML) algorithms. In 2012, Orozco et al. presented a model to classify lung nodules inside CT thorax images in the frequency domain. They picked a Region of Interest (ROI) manually, then a two-dimensional Discrete Cosine Transform (2D-DCT) and the two-dimensional Fast Fourier Transform (2D-FFT) calculated. Finally, the work uses a support vector machine (SVM) for classification from the two statistical features that extracted from each CT image. The method acquired the accuracy of 82.66% [4]. Eskandarian & Bagherzadeh in 2015 proposed a Computer-Aided Diagnosis (CAD) system of Pulmonary Nodules in CT scans. They worked on 147 patients scan from LIDC image database. In the beginning, they decreased the data volume by data mining techniques, then divided by chest location, and lastly, the unnormal nodules are specified and detected. The method able to reduce the false positive ratio by combing the threshold with support vector machine, they achieved a sensitivity of 89.9% [5].

In the past few years, a new subfield of machine learning named Deep Learning (DL) is appear to researcher's community and could entice extra scientist's attention specially in medical image diagnose and analyze, Gulshan et al. [6], Esteva et al. [7], due to the fact that DL algorithms chiefly Convolutional Neural Network (CNN) capable of gaining outcomes more accurate than human expertise. Most deep learning techniques are depending on Neural Networks (NN) architecture with more hidden layers. One of the greatest characteristics of DL methodology is that it can directly work on images (data) without the need for hand-crafted feature extraction because the network learns to extract features whilst training which makes the process further precise with more less time. In 2016, Li et al. proposed a classifier using CNN algorithm for identifying pulmonary nodule, the classifier could recognize the ground glass opacity, solid, and semisolid nodule kinds. For training, the model uses 62,492 regions of interest images from "LIDC", the data is split to 21,720 non-nodules, and 40,772 nodules. The best-achieved results by the proposed method are 0.864% accuracy [8]. Song et al. they develop three kinds of network architecture which are CNN, DNN, and SAE to classify lung tumors, and then stratify these networks on CT scans that obtained from free dataset "Lung Image Database Consortium\Image Database Resource Initiative (LIDC-IDRI)". The experiments showed that CNN got better accuracy than other two networks which is the accuracy of 84.15% [9]. Paul et al. in 2018, the study proposed to use their own trained CNN, pre-trained CNN and radiomics features for lung nodule analysis in CT images. The system is enhanced by an ensemble of classifiers using different feature sets and learning approaches, they extracted probability predictions from different models on an unseen test set and combined them. At last ensemble able to get best-known accuracy of 89.45%, which are significant improvements over the previous best accuracy of 76.79% [10].

The objective of this study is to develop an automated lung nodule classification system from CT scan images using two different deep learning architectures which are Network In Network (NIN) (the NIN is instantiated from CNN) and standard Convolution Neural Network (CNN). The work concentrates on playing with the image size (cropped cube size) and different CNN hyper-parameters to gain the best possible accuracy, and lowest possible false positive rate.

Materials and Methods

A. Convolution Neural Network

CNN is a type of deep learning model for processing data that has a grid pattern, such as images, which is designed to automatically and adaptively learn spatial hierarchies of features, from low- to high-level patterns. CNN is a mathematical construct that is typically composed of three types of layers : convolution, pooling, and fully connected layers, as shown in (Figure-1). The first two, convolution and pooling layers, perform feature extraction, whereas the third, a fully connected layer,maps the extracted features into final output, such as classification.[11]

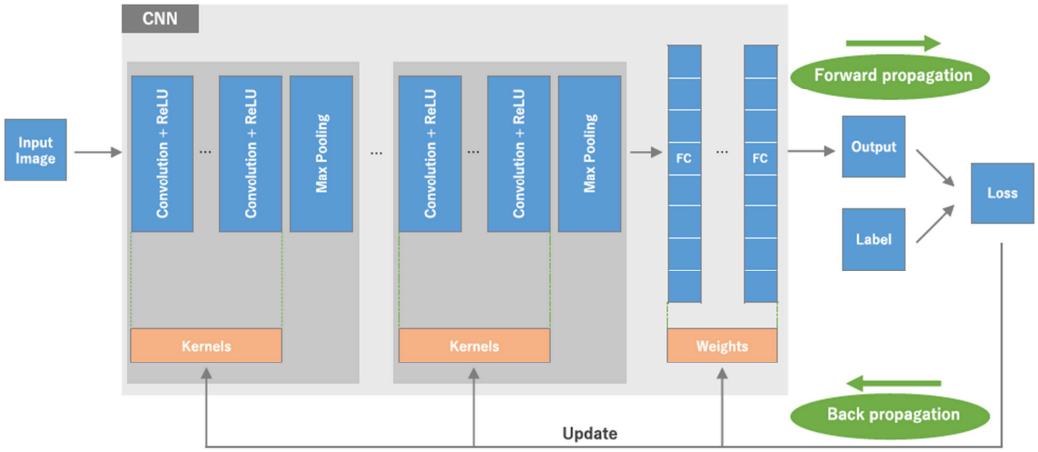

Figure-1: A CNN is composed of a stacking of several building blocks: convolution layers, pooling layers , and fully connected (FC) layers [11]

B. System Workflow

The workflow of almost all systems designed utilizing one of the deep learning algorithms follow the same principle. The model designed for this work using NIN and CNN follow up the identical steps to done its procedure, the preprocessing are applied to the images before feeding it to the networks, after that the images (pixel values) are directly feed into the network, which is do the process of feature learning and classification by itself, then the estimated outputs achieved from a classifier are compared with the targeted values in order to produce the system outcome, as shown (Figure: 2).

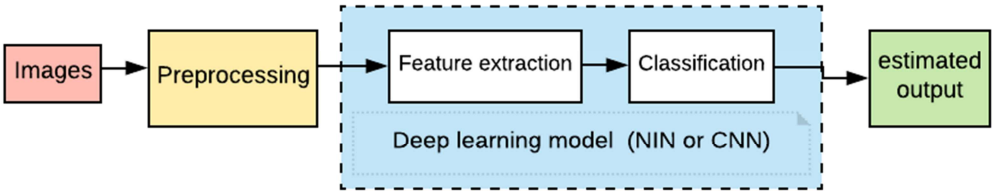

Figure-1: Demonstrates the designed system workflow.

C. Data preparation procedure

The database utilized for this study is 3D CT scan images from Lung Nodule Analysis 2016 (LUNA16) grand challenge that is a subset of "LIDC-IDRI". The original dataset ("LIDC-IDRI") is consist of 1018 scans [12]. But LUNA16 organizers constructed only 888 scans from the original dataset. The annotations are stored in (CSV) file format and the images were supplied in MetaImage (mhd/raw) file format. All the dataset is available for free download at LUNA16 website [13].

The data used for training, validation, and testing are small patches (cubes) cropped around the nodule location that constructed from 2D slices of the three-dimensional CT scans, the image content is read from (mhd) file. Each 3D scan has a unique ID and is consisting of a varying number of 2D slices, every slice is 512 x 512 in size, and it can contain a large number of nodules. However, within the dataset there is an annotation file that consist of 754,975 records, each record includes a scan unique DI, nodule location coordinates (X, Y, Z), and a class that indicates nodule positivity or negativity. There is a huge imbalance between the two classes, the positive class is only 1,557 samples and the rest is of negative class, so to solve this issue, 1,550 samples of minority (i.e. positive) class is chosen then to enlarge this class each cropped cube is augmented two times by rotating 60 and 180 degrees, in total 4,650 of positive samples are prepared, but for the majority (i.e. negative) class 1550 x 6 random samples are chosen (i.e. 9,300). Totally 13,950 samples are collected, in which %66 are negatives and %33 are positives. The algorithm for preparing the data is discussed as follows:

Begin

Declare data paths, image size, ***dataset*** *list.*

*Read annotations in (csv) file, pick 1,550 of positives (i.e. class = 1), then choose 1,550 * 6 of negatives (i.e. class = 0) randomly, save it in new (csv) file.*

Read newly created annotation file then read images from (mhd) file, through image ID, compare the image with the chosen annotation to be sure that the image and annotation are belong to each other.

For *i in img_list* ***do****:*

read image content from (mhd) file as an array, then get image origin and spacing.

For *i in annot_list* ***do****:*

read nodule cords (node_x, node_y, node_z) with a label.

Check if there is a problem with nodule location or given cords.

Crop 2D image around nodule location using node_x, node_y and ignore node_z.

If nodule at the corners of the image, pad the cropped image.

Save the cropped image (as an array) and a label in a list.

If label = 1, rotate the image 60 degree and save it again with the label in the list.

Again, if label = 1, rotate the image 180 degree and save it in the list.

Save all cropped image with associated labels in ***dataset*** *list.*

End*.*

End*.*

Save ***dataset*** *in (.npy) file.*

End*.*

The prepared cubes are small size square images that cropped around the nodule location from 2D slices according to the given nodule coordinates (X, Y) in annotation file, however, some of the nodules that are located at the corner of the images, so to crop the image around these nodules a part of the information is missing, to deal with this problem the cubes that taken from around these nodules are padded by linear ramp between end value and the array edge. To reduce computational time and cost, the cubes are cropped in two small sizes which are 32x32, 50x50, see (Figure: 3), then all images with their corresponding labels are stored as a 2D array in (.npy) file format.

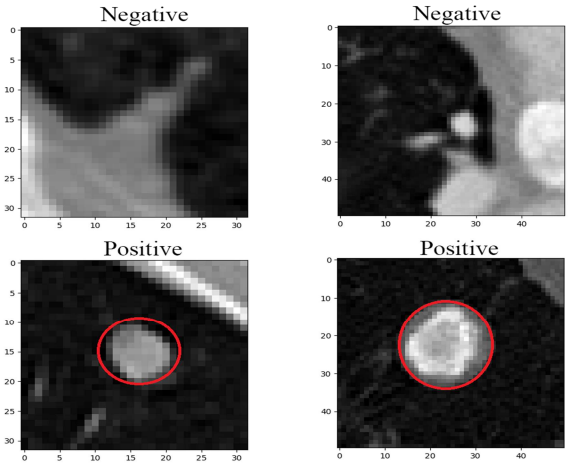

Figure-3: Shows the negative and positive sample images in each used size.

D. Network architecture

The architecture of networks adopted in this study is 2 different CNN's which are standard CNN and NIN. The NIN is novel CNN structure that proposed in late 2013 [13], the work made two changes in standard CNN architecture, the first one is replacing the general linear filter in convolution layer by ”micro-network”

which is a general non-linear method that instantiated using multilayer perceptron named "Mlpconv" for finer abstraction (Figure: 4), and the second one is removing the Fully Connected (FC) layer(s) to decrease networks complexity and lowering overfitting problem, as well as, the FC(s) are replaced by producing as many feature maps as there are classes within the last "Mlpconv" layer, and this followed by averaging (average pooling layer) activation maps to get to the final result.

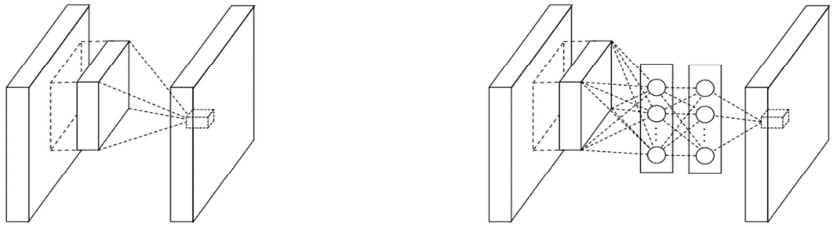

Figure-4: Displays the Difference between Linear Conv layer and "Mlpconv" layer [12].

The original NIN is comprised of a packing of three mlpconv layers and one average pooling layer, but the network that designed for this work have a few adjustments than the original one, it's made up of the stacking of two mlpconv layers and one average pooling, as well as, another attached layers. It's include a pair of Convolution layers, the Conv layers are followed by batch normalization [14], and an activation function of type Rectified Linear Unit (RELU) [15], afterward there are a pair of "Mlpconv" layers followed each of the 2 conv layers, also every "Mlpconv" layer followed by batch normalization, and RELU, later, the max pool and dropout layer respectively follow up the first stacking of 3 layers (i.e. 1 conv and 2 "Mlpconv"), but the second stacking of 3 layers are followed by average pooling, dropout, and subsequently an output layer that turns out the predicted output as shown in (Figure: 5).

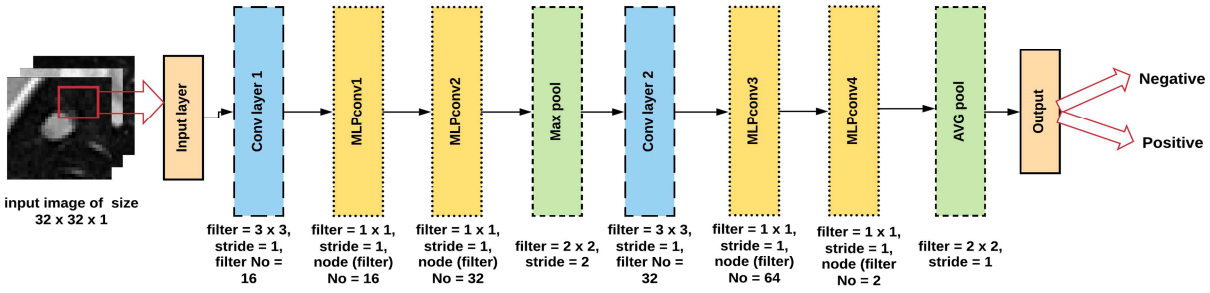

Figure-5: Demonstrates a 3x3 convolution filter of the prepared NIN architecture for proposed model.

The typical CNN is self-designed and experimented architecture, it's comprised of a quadruple of Conv layers, every Conv layer followed by a batch normalization, and RELU respectively, a max pool layer follows up each group of 2 Conv layers, however, after second max pool there is a Fully Connected (FC) layer that followed by RELU, dropout, and at last an output layer severally (Figure: 6).

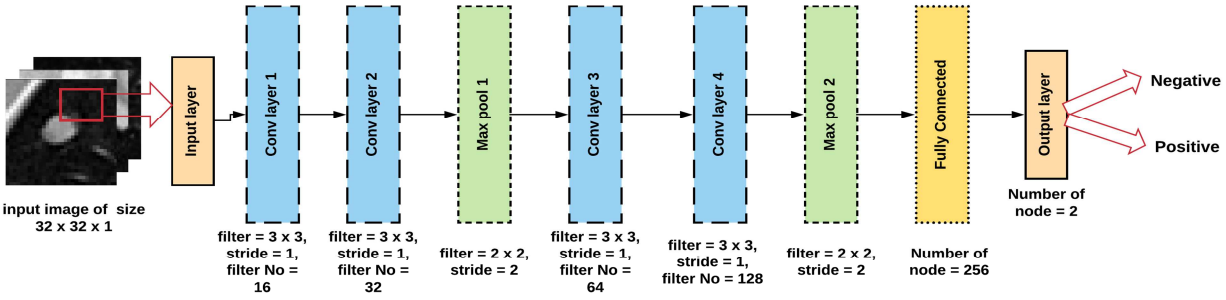

Figure-6: Demonstrates a 3x3 filter of the prepared Typical CNN architecture.

E. Training process

The two architectures are trained, validated, and tested using 13,500 samples from the prepared 13,950 samples, and it's divided into 60%, 20%, %20 respectively. In order to attain maximum doable accuracy, several hyperparameters are practiced for training both models, begin with filters, a diverse size of filters are utilized like 3x3, 5x5, and 7x7, going to learning rate, various learning ratios are used such as 0.001, 0.0001, 0.00001, furthermore, three distinctive optimizers are tested which are Adam, Adagrad, and Gradient Descent Optimizer (GDO), ending with dropout rate, the four rates are examined that are 0.5, 0.6, 0.7, and 0.8, also, the training operation includes using both patch size (i.e. 32x32 and 50x50) for each of the models. However, either models trained with the batching idea instead of feeding all data samples to the network once, a batch size of 162 is used for each iteration, at each 2 iterations the loss and accuracy are displayed for training and validation set, lately, the average loss and accuracy for both sets are taken and calculated as a final achieved result for the above-named sets, lastly, the accuracy, precision, and recall for test set is measured by the following equations:

$$Accuracy = \frac{TN+TP}{TP+FP+FN+TN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$FPR = \frac{FP}{FP+TN} \quad (4)$$

The two models are trained on Dell Inspiron core i7 laptop using CPU version of TensorFlow library for python programming language, which is a novel open source machine learning framework designed by google brain team.

Results and Discussion

This section, discusses in detail the experiments and the results of each model separately, after that, the differences and similarities among the two model outcomes are shown.

Starting with NIN architecture, the prepared model is trained and practiced with all hyperparameters that explained in materials and methods section C, the experimented results for this model showed a high accuracy and precision with low recall rate as follows, starting with the varying dropout rates, they do not change much in the scores, the four examined dropout rates almost have a constant impact on acquire results. By coming to learning rate, it confirmed a remarkable influence, the larger of the three finished with a better result. Goin to filter size, the scores between 3x3 and 5x5 are close to each other but the smaller one is more efficient. Also, as expected the Adam optimizer still favorite optimizer among all others. Reaching the image size, the smaller size gets a bit better results as well as it can be trained with the shorter amount of time. Lastly, a model with the use of 32x32 cube size, filter size 3x3, learning rate 0.001, and Adam optimizer achieved highest scores which are accuracy 90%, precision 99%, and recall 68%, detailed information shown in table-1.

The typical CNN is also trained and experimented with all above named hyperparameters, this architecture showed better outcomes, unlike NIN, the dropout rate has a sensible effect on regular CNN, a higher rate has a leverage on training scores. Our examinations presented that 0.0001 is the best learning ratio among the three tested ones. By talking about filter size, the experiments displayed that the bigger one is better in our case, also, does not forget that the achieved results between 5x5 and 7x7 filters are convergent. However, better results achieved with the use of 32x32 cube size than 50x50 size, furthermore, the model train time is way less by using smaller patches. Finally, the standard CNN obtained highest outcomes with the use of 32x32 image size, 5x5 filter size, 0.7 dropout rate, 0.0001 learning rate, and Adam optimizer which are accuracy 90%, precision 85%, and recall 85%, more information shown in table-2.

Comparing the two architectures, the train time needs for best achieved result by NIN is about 30 minutes, but regular CNN is trained in about a half of NIN time that is 16 minutes, naturally the train time depends on train sample size, image size, and filter size, bigger sizes of each one requires more time. The final experimented results showed that, the NIN carry out better precision and this means the lowest false

positive rate shown in (Figure :7), for that, it can be said that the contribution of this work which is the designed NIN architecture, can attain better scores. In (Figure:8) and (Figure:9) are shown the changes in accuracy and cost for two model per time.

Table-1: shows the best achieved results by Network in network.

Image size = 32x32, Learning rate = 0.001					
Filter size	***Dropout rate***	***Optimizer***	***Accuracy***	***Precision***	***Recall***
3x3	*0.8*	*Adam*	*90%*	*99%*	*68%*
5x5	*0.7*	*Adam*	*90%*	*100%*	*68%*
7x7	*0.7*	*Adam*	*89%*	*97%*	*68%*
3x3	*0.5*	*GDO*	*89%*	*95%*	*69%*
7x7	*0.7*	*Adagrad*	*89%*	*97%*	*68%*
Image size = 50x50, Learning rate = 0.001					
3x3	*0.7*	*Adam*	*89%*	*99%*	*66%*
5x5	*0.8*	*Adam*	*89%*	*100%*	*66%*
7x7	*0.5*	*Adam*	*89%*	*100%*	*66%*
7x7	*0.8*	*GDO*	*88%*	*95%*	*66%*
7x7	*0.8*	*Adagrad*	*88%*	*97%*	*66%*

For testing the system and producing a confusion matrix in both networks, 20% of the hole dataset, which means 2700 samples are applied, the actual values for those samples are 1809 negative, and 891 positive samples. The system with NIN architecture could result out 1808 of negative samples as a negative and did only one error, but for the positive samples it could recognize 609 of positive samples as positive and did 282 errors. However, with the CNN architecture, the system could outcome 1673 of negative samples as a negative and did 136 errors, but for the positive samples it could realize 753 of positive samples as positive and did 138 errors (Figure :7).

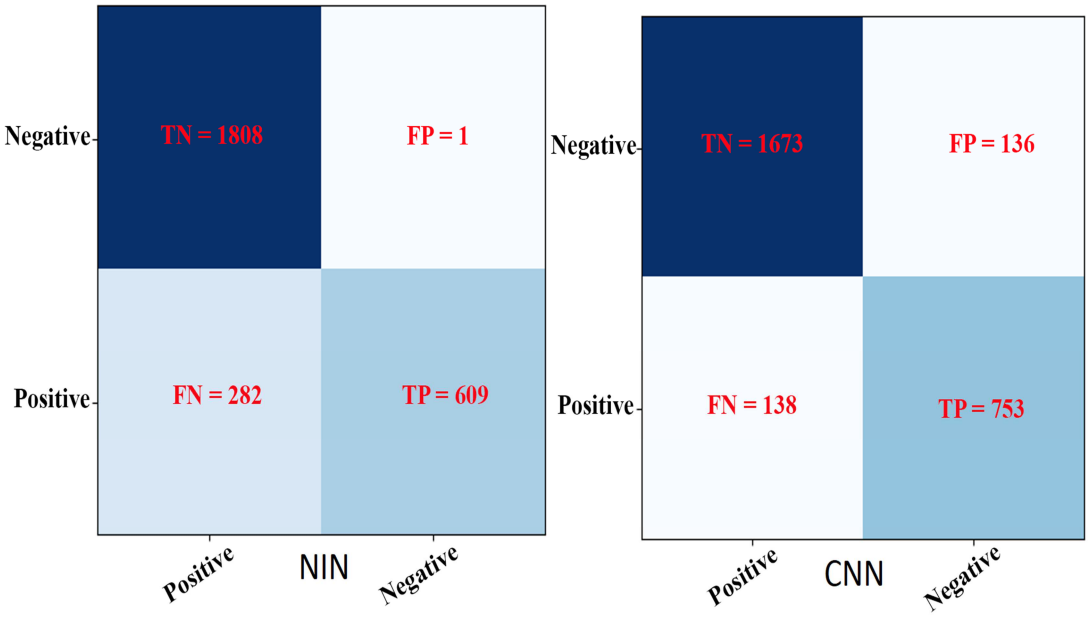

Figure-7: Shows the confusion matrix for the best result of both models.

Table-2: shows the best achieved results by standard CNN.

Image size = 32x32, Learning rate = 0.0001					
Filter size	***Dropout rate***	***Optimizer***	***Accuracy***	***Precision***	***Recall***
3x3	*0.7*	*Adam*	*90%*	*91%*	*78%*
5x5	*0.7*	*Adam*	*90%*	*85%*	*85%*
7x7	*0.8*	*Adam*	*90%*	*85%*	*84%*
7x7	*0.8*	*GDO*	*90%*	*100%*	*68%*
3x3	*0.7*	*Adagrad*	*90%*	*100%*	*68%*
Image size = 50x50, Learning rate = 0.0001					
3x3	*0.7, 0.8*	*Adam*	*%89*	*100%*	*67%*
5x5	*0.8*	*Adam*	*90%*	*99%*	*70%*
7x7	*0.8*	*Adam*	*90%*	*98%*	*71%*
7x7	*0.8*	*GDO*	*89%*	*100%*	*66%*
7x7	*0.8*	*Adagrad*	*89%*	*100%*	*66%*

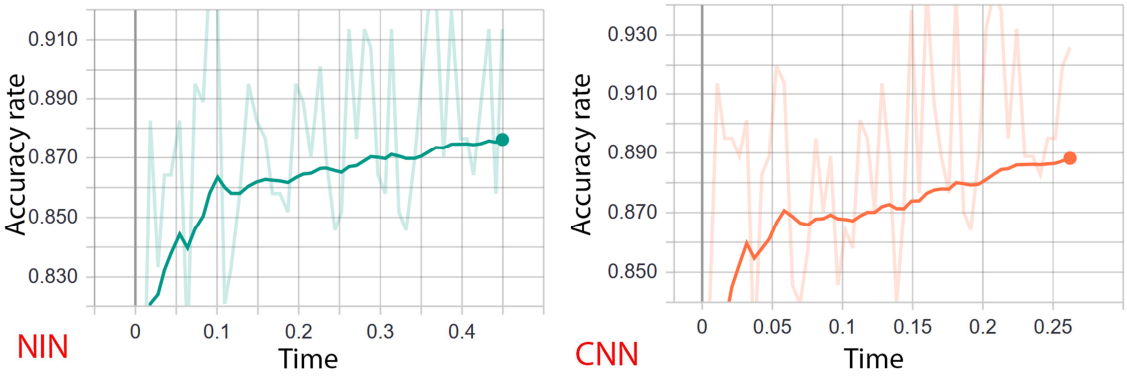

Figure-8: Illustrates changing in train accuracy per time for both models.

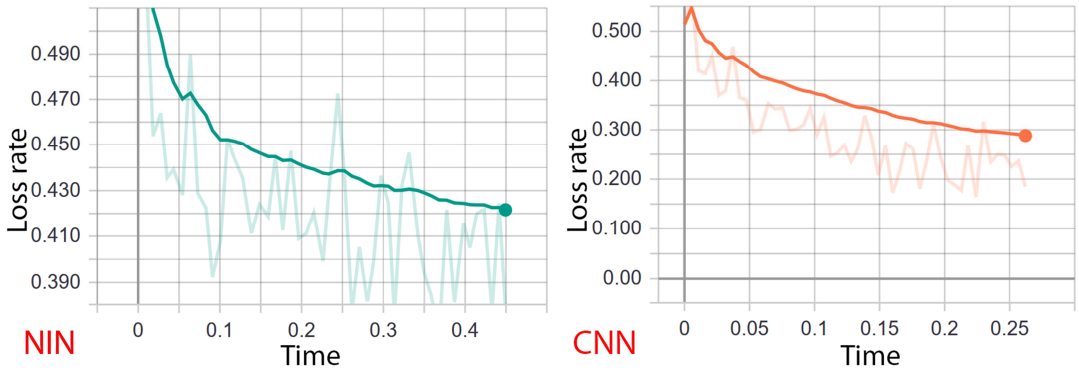

Figure-9: Illustrates changing in train cost per time for both models.

Conclusion

Lung cancer is a major cause of death rate worldwide among all other kinds of cancer, for that, Detecting and classifying lung nodules in CT scan images is a challenging task for radiologist experts due to a vast number of recorded cases each year. A good solution for this problem is the use of a computer system to aid radiologists in the process of reading and recognizing lesions in CT scans. However, this work discusses designing a computer system for the purpose of analyzing lung nodules in CT scan images using the two close but different deep learning architectures. The two models are examined with different hyperparameters and extensively estimated for the objective of comparison. The data is obtained from LUNA16 challenge, and because of limited availability of computation power, a 32x32 images around the nodule location were cropped and prepared for training and testing purpose. The proposed methodology within the restricted availability of the data achieved a very good outcome, also it can be further improved by employing more data. Most published studies in same task foxing on lowering false positive rate, according to that, the results that achieved by designed NIN model is preferable one, also this can be seen as the studies effort that tries to reduce the false positives into lowest possible rate, and in this attempt the offered model could obtain fairly very well consequence. At last, the obtained scores of the two architectures are as follows, the developed NIN achieved an accuracy of 90%, precision 99%, recall 68%, and false positive rate 0.06%, but the standard CNN achieved an accuracy of 90%, precision 85%, recall 85%, and false positive rate 7.52%, which shows that designed NIN architecture can perform better in terms of reducing false positive rate.

Acknowledgments

I wish to give my full thank and gratitude to Lung Nodule Analysis 2016 grand challenge organizers who are tired for preparing this useful dataset with its annotations and give every one accessibility to download it.

References

[1] F. Bray, J. Ferlay, L. A. T. Isabelle Soerjomataram, Rebecca L. Siegel, and A. Jemal, *"Incidence and survival in sarcoma in the United States: A focus on musculoskeletal lesions",* CA. Cancer J. Clin. Vol. 68, No. 6, pp. 394–424. (2018).

[2] H. Tang, D. R. Kim, and X. Xie, *"Automated pulmonary nodule detection using 3D deep convolutional neural network",* in 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), No. Isbi, pp. 523–526. (2018).

[3] J. Lyu, S. H. Ling, and S. Member, *"Using Multi-level Convolutional Neural Network for Classification of Lung Nodules on CT images",* in 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 686–689. (2018).

[4] H. M. Orozco, O. O. V. Villegas, L. O. Maynez, V. G. C. Sanchez, and H. D. J. O. Dominguez, *"Lung nodule classification in frequency domain using support vector machines",* 11th International Conference on Information Science, Signal Processing and their Applications, ISSPA 2012, pp. 870–875. (2012).

[5] P. Eskandarian and J. Bagherzadeh, *"Computer-aided detection of Pulmonary Nodules based on SVM in thoracic CT images",* 7th Conference on Information and Knowledge Technology, IKT 2015, pp. 1–6. (2015).

[6] Gulshan V, Peng L, Coram M et al, *"Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathyin Retinal Fundus Photographs"*. JAMA. Vol. 316, pp. 2402–2410. (2016).

[7] Esteva A, Kuprel B, Novoa RA et al, *"Dermatologist-Level Classification of Skin Cancer With Deep Neural Networks",* Nature. Vol. 542, pp. 115–118. (2017).

[8] W. Li, P. Cao, D. Zhao, and J. Wang, *"Pulmonary Nodule Classification with Deep Convolutional Neural Networks on Computed Tomography Images",* Comput. Math. Methods Med. pp.1-7. (2016).

[9] Q. Song, L. Zhao, X. Luo, and X. Dou, *"Using Deep Learning for Classification of Lung Nodules on*

Computed Tomography Images", J. Healthc. Eng. Vol. 2017. (2017).

[10] R. Paul, L. Hall, D. Goldgof, M. Schabath, and R. Gillies, *"Predicting Nodule Malignancy using a CNN Ensemble Approach",* in 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. (2018).

[11] R. Yamashita, M. Nishio, R. K. G. Do, K. Togashi, *"Convolutional neural networks: an overview and application in radiology",* Insights into imaging ,Vol. 9, No. 4, pp. 611–629. (2018).

[12] S. G. Armato *et al.*, *"The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans", Med. Phys.* Vol. 38, No. 2, pp. 915–931. (2011).

[13]A. A. A. Setio *et al.*, *"Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge",* Med. Image Anal. Vol. 42, pp. 1–13. (2017).

[14] M. Lin, Q. Chen, and S. Yan, *"Network In Network",* Neural Evol. Comput., pp. 1–10. (2013).

[15] R. S. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, *"Dropout: A Simple Way to Prevent Neural Networks from Overfittin",* J. Mach. Learn. Res. Vol. 15, pp. 1929–1958. (2014).