Lung nodule classification using Convolutional Autoencoder and Clustering Augmented Learning Method(CALM)

Soumya Suvra Ghosal
soumyasuvraghosal@gmail.com
NIT Durgapur
Durgapur, India

Indranil Sarkar
indranil.sarkar.nitdgp@gmail.com
NIT Durgapur
Durgapur, India

Issmail El Hallaoui
issmail.elhallaoui@gerad.ca
Ecole Polytechnique de Montreal
Montreal, Canada

ABSTRACT

Early detection of lung cancer can help in a sharp decrease in the lung cancer mortality rate, which accounts for more than 17% percent of total cancer-related deaths. A large number of cases are encountered by radiologists daily for initial diagnosis. Computer-Aided Diagnosis(CAD) systems can assist radiologists by offering a second opinion and making the whole process faster. However, one drawback of CAD systems is a large amount of data needed to train them, which can be expensive in the medical field.

In this paper, we propose using a generative adversarial network(GAN) as a potential data augmentation strategy to generate more training data to improve CAD systems. We also propose a convolutional autoencoder deep learning framework to support unsupervised image features learning for lung nodule through unlabeled data. The paper also introduces Clustering Augmented Learning Method (CALM) classifier which is based on the concept of simultaneous heterogeneous clustering and classification to learn deep feature representations of the features obtained from Convolutional autoencoder.

The classification model within CALM consists of a Feedforward Neural Net (FNN) architecture. To improve the accuracy of the classification model, CALM iterates between clustering and learning to form robust clusters, thereby leveraging the learning process of the FNN.

Computational experiments using the National Cancer Institute (NCI) Lung Image Database Consortium (LIDC) dataset resulted in an overall accuracy of 95.3% with a precision of 94.9%.

CCS CONCEPTS

• **Computing Methodologies** → Machine learning; Feature Selection; • **Information systems** → Information systems applications; Data mining; • **Applied Computing** → Health informatics.

KEYWORDS

Convolutional Autoencoder Neural Network, Lung Nodule, Generative Adversarial Networks, Deep Features

ACKNOWLEDGEMENT

This work was presented at the first Health Search and Data Mining Workshop [5].

1 INTRODUCTION

The use of computer tools, basic machine learning to facilitate and enhance medical analysis and diagnosis is a promising area. The study of the correlation between gene expression profiles and disease states or stages of cells plays an important role in biological and clinical applications. The gene expression profiles can be obtained from multiple tissue samples and comparing the diseased tissue with the normal one. One main challenge in this regard is to determine the difference between cancerous gene expression in tumor cells and the gene expression in normal, non-cancerous tissues. Many machine learning classification techniques and algorithms have been proposed to address this problem. Hence intelligent healthcare systems are an important research direction to assist doctors in harnessing medical big data.

And among all types of cancer Lung cancer is harder to detect in early stages as there is only a dime-sized lesion growth known as a nodule, inside the lung. By the time when it can be detected, is already too late for the patient. Also, these small lesions are only detectable by a CT scan.

Especially it is difficult to identify the images containing nodules, which should be analyzed for assisting early lung cancer diagnosis, from a large number of pulmonary CT images. At present, the image analysis methods for assisting radiologists to identify pulmonary nodules consist of four steps:1) region of interest(ROI) definition, 2) segmentation, 3) hand-crafted features and 4) categorization. In particular, radiologist has to spend a lot time on checking each image for accurately marking the nodule, which is critical for diagnosis and is a research hotspot in intelligence healthcare.

For example, it is proposed to extract texture features for nodules analysis, but it is hard to find effective texture feature parameters. Previously nodules were analyzed by the morphological method through shape, size, and boundary, etc. However, this analytical approach is difficult to provide accurate descriptive information. It is because even an experienced radiologist usually gives a vague description based on personal experience and understanding. Therefore, it is a challenging issue to effectively extract features for representing the nodules.

Recently CAD systems have taken advantage of the popular Convolutional Neural Network(CNN), producing state of art detection results, with 95% sensitivity at only 10 false positives per scan. However, CNN requires a large amount of training data to learn effectively; in the medical field, obtaining the required data is often costly, time-consuming, or simply not feasible. To deal with these issues, data augmentation is often used to better train these CAD systems.

In [3], the authors addressed the challenges by training a deep learning architecture based on the Convolutional Autoencoder Neural Network(CANN) for the classification of pulmonary nodules. Inspired by results obtained, we also use a similar architecture for extracting deep features from CT images. Besides, we present a

new way to improve lung nodule detection in existing systems by augmenting training datasets with the generated image of nodules. To create these images, we propose the use of a type of Generative Adversarial Network (GAN). The augmentation of data would help in more accurate supervised fine-tuning of proposed model.Overall, the proposed method utilizes both the original and generated image for unsupervised feature learning and some amount of data for fine-tuning. Computational experiments show that the proposed method is effective to extract image features via a data-driven approach, and achieves faster labeling for medical data. Specifically, the main contributions of this paper are :

- Application of GANs to augment the training data for computer-aided lung nodule detection systems and address the issue of the insufficiency of training data.
- Image features are available to be directly extracted from the raw image. Such an end-to-end approach does not use an image segmentation method to find the nodules, avoiding loss of important information which might affect classification results.
- The unsupervised data-driven approach can extend to implement in other data sets and related applications.
- Devising a classification approach in which data is clustered based on their inherent characteristics. In the process of learning the best clustering solution, the parameters of the classification model are optimized, thereby substantially improving the learning process.

2 RELATED WORKS

In the past, several methods have been proposed to detect and classify lung cancer in CT images using a different algorithm. Aliferis et al. [2] used recursive feature elimination with single variable association filtering approaches to select a small subset of the gene expressions as a reduced feature set. For better classification Ramaswamy [13] applied recursive feature elimination using SVM to find similarly a small number of genes. Wang et al. [18] proved that if the correlation-based feature selector can be combined with a classification approach then it can obtain good classification results with high confidence. Sharma et. al [15] proposed to find an informative subset of gene expression using feature selection methods. It's like the "Divide & Conquer" approach. As form the subset they are finding the informative genes, and then they are combining to form the overall subset. Nanni et al. [11] proposed a method that combines different feature reduction approaches, useful for gene microarray classification. In Zinovev et al. [21], the authors used decision trees to classify lung nodules using the LIDC dataset. The features taken by them are lobulation, texture, speculation, etc. Those are used to create a 63-dimensional feature vector for classification of 914 instances. The authors got an overall accuracy of 68.66%. Kuruvilla et al. [10] used six distinct parameters including skewness and fifth & sixth central moments, which are extracted from segmented single slices, containing 2 lung images along with the features mentioned in [1] and have trained a feed-forward backpropagation neural network. There has also been a renewed interest in the field of deep learning and the latest research in the area of medical imaging using deep learning shows some good results. One such paper is of Suk et al., [17] in which the authors propose a novel latent and shared feature representation of neuro-imaging data of the brain using Deep Boltzmann Machine (DBM) diagnosis. The methods achieved a maximal diagnostic accuracy of 95.52%. In Riccardi et al. [14] the authors proposed a new algorithm, which can automatically detect nodules with an overall accuracy of 71%. It used 3D radial transforms. Kumar et al. [9] proposed to use deep features extracted from an autoencoder along with a binary decision tree as a classifier to build their proposed system for lung cancer classification. Wu et al. [19] proposed deep feature learning for deformable registration of brain MR images. They demonstrate that a general approach can be built to improve image registration by using deep features. Fakoor et al. [6] proposed a method to enhance cancer diagnosis and classification from gene expression data using unsupervised and deep learning methods. Their model used PCA (Principal Component Analysis) to achieve dimensionality reduction in case of the very high dimensionality of the initial raw feature space. Chuquicusma et al. [4] proved in his paper that the GANs are able to generate realistic fake images that fool even experienced radiologists.
Maayan et al. [7] used GANs to augment liver lesion images to improve the multiclass CNN classification. He got an increase from 85.7% and 92.4% sensitivity and specificity which is much higher as compared to recent state-of-the-art liver classification methods. Zhu et al. [20] showed in his work that Generative Adversarial Networks(GANs) can be used to complement and complete the training data manifold. It can find better margins between classes. They had done their work by using GANs to augment the emotion categories that were lacking in face data and they could achieve a 5% to 10% increase in the accuracy of emotion classification.

In this paper, we propose a convolutional autoencoder unsupervised learning algorithm for lung CT features learning and CALM classifier for pulmonary nodules classification. To tackle the issue of scarcity of medical labeled images, we use a type of Generative Adversarial Networks(GANs) to augment data to the training set.

3 PRELIMINARIES

3.1 Generative Adversarial Networks(GAN)

Generative Adversarial Networks(GANs) are a type of neural network where two competing networks - the generator and the discriminator - are adversarially trained against one another. The discriminator is trained to differentiate between real data and generated data while the generator attempts to fool the discriminator by generating synthetic data. More specifically, the generator G samples from a previously known data distribution z $\sim P_z(z)$ (usually a Gaussian) and generates data G(z) by putting z through a function G. The discriminator D takes in data x and produces a probability that x is a sample from the real data distribution $P_{data}(x)$. The loss function that the discriminator D maximizes and the generator G minimizes is

$$\text{L=}\min_G \max_D \mathbb{E}_{x\sim P_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim P_z(z)}[\log(1 - D(G(z))]$$

While this original GAN is useful for a multitude of tasks, the Jensen-Shannon divergence as loss function inherently struggles to learn probability distributions between low dimensional manifolds in a higher-dimensional space. Wasserstein GANs (WGANs) attempt to solve this problem by using an approximation of the Earth-Mover

distance as the loss function, which enables more stable GAN training. The discriminator is now replaced with a critic as its output is no longer a probability; rather, it is a 1-Lipschitz function that tries to maximize the difference in score between the real data and the generated data. A function is 1-Lipschitz if and only if the norm of its gradient everywhere is at most 1. The authors of the WGAN paper enforces that the critic is 1-Lipschitz by weight-clipping, which may lead to optimization difficulties. The new loss function is as follows:

$$\text{L=}\min_G \max_{D\in\mathbb{D}} \mathbb{E}_{x\sim P_{data}(x)}[\log D(x)] - \mathbb{E}_{z\sim P_z(z)}[\log D(G(z))]$$

Where D is the set of 1-lipshitz functions.

3.2 Autoencoder

An autoencoder takes an input x$\in \mathbb{R}_d$ and first maps it to latent representation h$\in \mathbb{R}_{d'}$ using a deterministic function of type **h** $= f_\theta = \sigma(Wx + b)$ with parameters θ = {W,b}. This "code" is then used to reconstruct the input by a reverse mapping of f: **y**= $f_{\theta'}(h) = \sigma(W^{'}x + b^{'})$ with $\theta^{'}$ = {W$^{'}$,b$^{'}$}. The two parameter sets are usually constrained to be of form $W^{'} = W^T$, using the same weights for encoding the input and decoding the latent representation. Each training pattern x_i is then mapped onto its code h_i and its reconstruction y_i. The parameters are optimized, minimizing an appropriate cost function over the training set $D_n = \{(x_0, t_0), ..., (x_n, t_n)\}$.

3.3 Denoising Autoencoders(DAE)

Without any additional constraints, conventional autoencoders learn identity mapping. This problem can be circumvented by using a probabilistic RBM(Restricted Boltzmann Machine) approach, or sparse coding, or denoising autoencoders trying to reconstruct noisy inputs. The latter performs as well as or even better than RBMs. Training involves the reconstruction of a clean input from a partially destroyed one. Input x becomes corrupted input $\overline{x}$ by adding a variable amount v of a noise distributed according to the characteristics of the input image. Common choices include binomial noise(switching pixels on or off) for black and white images or uncorrelated Gaussian noise for color images. Parameter v represents the percentage of permissible corruption. The auto-encoder is trained to denoise the inputs by first finding the latent representation **h**= $f_\theta(\overline{x}) = \sigma(W\overline{x} + b)$ from which it reconstructs the original input **y**= $f_{\theta'}(h) = \sigma(W^{'}h + b^{'})$

3.4 Convolutional Neural Networks

CNN's are hierarchical models whose convolutional layers alternate with subsampling layers, reminiscent of simple and complex cells in the primary visual cortex. The network architecture consists of three basic building blocks to be stacked and composed as needed,i.e, the convolution layer, the max-pooling layer, and the classification layer.

3.5 Convolutional Auto Encoder(CAE)

A fully connected autoencoder ignores a 2-D image structure. This is not only a problem when dealing with realistically sized inputs but also introduces redundancy in the parameters, forcing each feature to be global. However, the trend in vision and object recognition adopted by most successful models is to discover localized features that repeat themselves all over the input. CAEs differ from conventional AEs as their weights are shared among all the input, preserving spatial locality. The reconstruction is hence due to a linear combination of basic image patches based on latent code. CAE combines the local convolution connection with the autoencoder, which is a simple operation to add a reconstruction input for the convolution operation. The procedure of the convolutional conversion from feature maps input to output is called convolutional encoder. Then the output values are reconstructed through the inverse convolutional operation, which is called a convolutional decoder. Moreover, the parameters of the encode and decode operation are calculated through standard autoencoder unsupervised greedy training.

Input feature maps $x \in \mathbb{R}^{n\times l\times l}$, which are obtained from the input

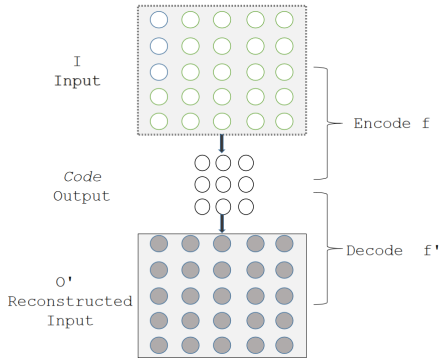

Figure 1: Convolutional Autoencoder

layer or the previous layer. It contains n feature maps, and size of each feature map is $l \times l$ pixels. The convolutional autoencoder operation includes m convolutional kernels, and the output layer output m feature maps. When the input feature maps from previous layer, n represents the number of output feature maps from the previous layer. The size of convolutional kernel is $d\times d$, where $d \leq l$. θ ={W,$\hat{W}$, b, $\hat{b}$} represents the parameters of convolutional autoencoder layer need to be learned, while b$\in \mathbb{R}^m$ and W={w_j,j=1,2,...,m} represents the parameters of convolutional autoencoder, where $w_j \in \mathbb{R}^{n\times l\times l}$ is defined as a vector $w_j \in \mathbb{R}^{nl^2}$. And $\hat{W}$={$\hat{w}_j$,j=1,2,...,m} and $\hat{b}$ represent the parameters of convolutional decoder, where $\hat{w}_j \in \mathbb{R}^{nl^2}$.

First the input image is encoded that each time a $d \times d$ pixels patch x_i,i=1,2,...,p is selected from input image, and then the weight w_j of the convolutional kernel j is used for convolutional calculation. Finally the neuron value o_{ij},j=1,2,...,m is calculated from the output layer.

$$o_{ij} = f(x_i) = \sigma(W_j x_i + b)$$

where σ is a nonlinear activation function, often including three functions,i.e, the sigmoid function, the hyperbolic tangent function, and the rectified linear function(Relu). We implemented Relu in this paper.

Then o_{ij} output from the convolutional decode is encoded that x_i is reconstructed via o_{ij} for generated $\hat{x}_i$.

$$\hat{x}_i = f^{'}(o_{ij}) = \phi(\hat{W}_i o_{ij} + \hat{b})$$

$\hat{x}_i$ is generated after each convolutional encode and decode. P patches are obtained from reconstruction operation of dimension

$d \times d$. We use the mean square error between the original patch of input image x_i,(i=1,2,...p) and the reconstructed patch of image $\hat{x}_i$,(i=1,2,...p) as the cost function. Furthermore, the cost function and reconstruction error is described as:

$$J_{CAE}(\theta) = \frac{1}{p}\sum_{i=1}^{p} L[x_i, \hat{x_i}]$$

$$L_{CAE}[x_i, \hat{x_i}] = ||x_i - \hat{x_i}||^2 = ||x_i - \phi(\sigma(x_i))||^2$$

Through stochastic gradient descent(SGD), the weight and error are minimized, and the convolutional autoencoder layer is optimized. Finally, the trained parameters are used to output the feature maps which are transmitted to the next layer.

4 METHODOLOGY

For our model, we will be using WGAN with gradient penalty (WGAN-GP), a version of WGAN that replaces weight-clipping with a gradient penalty of the critic - constraining the gradient norm of the critic's output concerning its input. This allows for more stable GAN training. The optimal WGAN or WGAN-GP critic will contain straight lines with gradient norm 1 connecting coupled points between P_{data} and $P_{G(z)}$; since enforcing the unit gradient norm constraint everywhere is intractable, it is only enforced along these straight lines. The new loss function is as follows:

$$\text{L=min}_G \max_{D\in\mathbb{D}} \mathbb{E}_{z\sim P_z(z)}[\log D(G(z)] - \mathbb{E}_{x\sim P_{data}(x)}[D(x)] + \lambda\mathbb{E}_{\hat{x}\in P_{\hat{x}}(\hat{x})}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2]$$

Where λ is the weight given to the gradient penalty. $\hat{x} \sim P(\hat{x})$ are random samples that have uniform distribution along straight lines between pairs of points sampled from the real data distribution P_{data} and the generated data distribution $P_{G(z)}$. We hypothesize that generated data can improve lung nodule detection sensitivity, allowing for better training of CAD systems with existing data. We can use the generator to produce new training data to augment the existing training data.

Since the workload for labeling ROI is high and the pulmonary nodules are difficult to be recognized, the CT images are divided into small patch areas for training the network. The patch divided from the CT image is input to Convolutional Autoencoder(CAE) for the purpose of learning the feature representation, which is used for classification. The parameters of convolution layers in CNN are determined by autoencoder unsupervised learning, and some data is used for fine-tuning the parameters of the CAE and training the classifier.

The patch divided from the original CT image can be represented as $x \in$ X, X $\subset \mathbb{R}^{m\times d\times d}$, where m represents the number of the input channel, and $d \times d$ represents the input image size. The labeled data are represented as $y \in Y$, Y$\subset \mathbb{R}^n$, where n represents the number of output classification. Through the proposed model, it is expected to deduce the hypothesis function from the training,i.e.,f: X$\longrightarrow$Y and the set of parameters θ.

In the proposed model, the hypothesis function f based on deep learning architecture consists of multiple layers, which is not a direct mapping from X to Y. Specifically, the first layer L_1 receives the input image x and the middle layer has three convolution layers and three pooling layers.

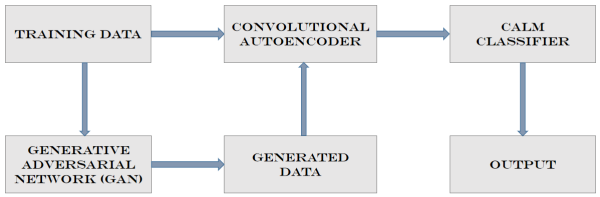

Figure 2: Block Diagram of Proposed Model

Algorithm 1: Unsupervised Training of CAE

1 Given dataset U, number of convolution, pooling layer along with all weight matrices and bias vectors are randomly initialized
2 i⟵1
3 **if** *i==1* **then**
4 The input of C_i is U
5 **else**
6 The input of C_i is output of P_i
7 Greedy layer wise training C_i
8 Find parameters of C_i by cost function
9 Output of C_i is input to P_i
10 Max Pooling Operator
11 **if** $i < N$ **then**
12 goto line 3

The convolutional autoencoder has the following architecture :

- Input: 40×40 patch image from CT image
- C1: Convolution kernel of size 5×5, Number of kernel is 50, non linear function is ReLU.
- P1: Max pooling is used, the size of pooling area is 2×2 with stride 2.
- C2: Convolution kernel of size 3×3, Number of kernel is 50, non linear function is ReLU.
- P2: Max pooling is used, the size of pooling area is 2×2 with stride 2.
- C3: Convolution kernel of size 3×3, Number of kernel is 50, non linear function is ReLU.
- P3: Max pooling is used, the size of pooling area is 2×2 with stride 2.

The convolutional autoencoder is trained in an unsupervised manner, which is explained in Algorithm 1 and the parameters are optimized through SGD. A mini-batch size of 100 samples and 150 iterations for each batch is used.
The output from the last pooling layer is fed as input to the CALM

classifier, which is explained in 5.

5 CLUSTERING AUGMENTED LEARNING METHOD (CALM)

5.1 Proposed Approach

Input augmentation We consider a matrix of input data D and a set of cluster centers C. Since in this case study, there are probabilities of the nodule being either malignant or not, we keep C as 2. In this paper, we use clustering to augment input data $x \in D$ for better learning. To augment the input data, we add a new set of features representing either an input example belongs to a cluster or not. To distinguish input examples, we introduce an additional index $h \in \{1, \ldots, |D|\}$ representing the number of an input example (x_1 is the first input example of D). We define also a vector c_h composed of c_{hl}, $l \in C$ for each example $x_h \in D$. It is a one-hot representation containing zeros except for the index of the cluster it belongs to (e.g. $c_1 = [0, 1]$ means that the first input example x_1 belongs to the 2nd cluster out of 2 clusters). Finally, we augment input examples by concatenating the vector x_h with the vector c_h for each $h \in \{1, \ldots, |D|\}$.

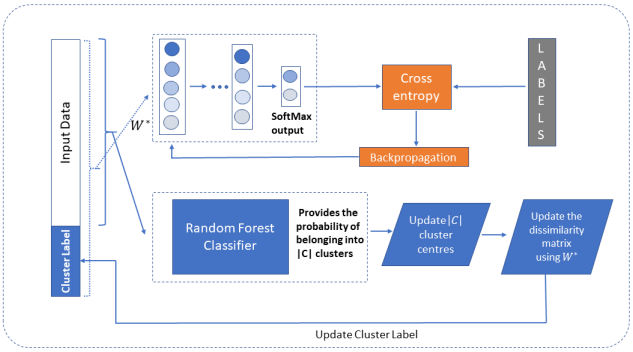

Figure 3: Architecture of Clustering Augmented Learning Method(CALM) Classifier

Cluster centers To determine the cluster centers, CALM consists of a clustering model and a Feed-Forward Neural Net(FNN) having a softmax output to classify the lung nodules. For the clustering model, we propose to use a Random Forest classifier to determine cluster centers. After the FNN is trained using a state-of-the-art solver for data belonging to a single cluster $\in \{1, \ldots, |C|\}$, a Random Forest Classifier is used to find the best cluster center. Hence we repeat $|C|$ instances of training the FNN to find the $|C|$ centers. For any instance l of the model, we use one hot encoded vector of l as labels for all the input sample in that cluster to train the random classifier in a supervised manner. In simple words, while predicting center of 2nd cluster (for example) we use $[0, 1]$ as label for all input sample in that cluster, since $|C|$ is 2. We propose that the input sample which has the lowest error in predicting its cluster label is considered as the center of that cluster in the subsequent iteration of the proposed approach. In such a manner, the center would be the input sample which is the most fitting representative of that cluster. As a result, the clustering process would aggregate the data having similar characteristics resulting in better learning by the FNN model. We include the following additional constraints:

$$c_{hl} = 1 \tag{1}$$

$$c_{ha} = 0, \quad \forall a \in \{1, \ldots, |C|\}, l \neq a \tag{2}$$

5.2 Clustering Problem

We have a distance/dissimilarity measure d_{il} between input examples $i \in D$ and cluster centers $l \in C$. The clustering problem aims to assign each input example to a cluster such that the total distance between the elements of a cluster and its center is minimized. We introduce a new set of binary variables c_{il} that is equal to 1 if input example $i \in D$ belongs to the cluster whose center is $l \in C$, and 0 otherwise. The clustering problem is formulated as follows:

$$\min \quad \sum_{i \in D} \sum_{l \in C} d_{il}\, c_{il} \tag{3}$$

$$\textbf{s.t.} \quad \sum_{l \in C} c_{il} = 1, \ \forall i \in D \ \textbf{And} \ c_{il} \in \{0, 1\}, \ \forall i \in D, \forall l \in C \tag{4}$$

The objective function (3) minimizes the total distance between a cluster center and its elements. Constraints (4) ensure that each element is assigned to exactly one cluster and that the decision variables are binary.

In this paper, we also propose a novel dissimilarity measure based on the weights of the trained FNN model. It uses the average of weights linked to each neuron of the input layer. Assuming that the original input (without the new clustering feature) has d dimensions ($x_h = [x_h^1, \ldots, x_h^d], h \in \{1, \ldots, |D|\}$) and the weight linking node n of the input layer to node $j \in \{1, \ldots, n_1\}$ of the following layer is w_j^n, the two distances measures are formulated as follows:

$$d_{il} = \sum_{n \in \{1 \ldots d\}} \ \underset{j \in \{1, \ldots, n_1\}}{\text{avg}} \ w_j^n |x_i^k - x_l^k|$$

Thus the distance measure computes the distance between two examples based on how important is the contribution of each input feature to the resulting prediction. Therefore, the resulting clusters contain examples with similar potential to improve the classification results.

5.3 Proposed Algorithm

As in Fig. We propose an approach (Algorithm 2) where we iteratively train the FNN classifier, use its weights for input data clustering thus changing the input vector, train again the FNN classifier using the new input data, and so on until a stopping criterion is attained. The stopping criterion is triggered if the cluster assignment remains the same for consecutive 10 iterations, i.e., the clustering problem converges.

The configuration of the proposed model is given as:

- **A) Classification Model**: `FC1` ⟶ Leaky ReLU ⟶ `FC2` ⟶ Leaky ReLU ⟶`FC3` ⟶ Softmax . Dimension of `FC1`: 128. Dimension of `FC2`: 32. Dimension of `FC3`: 2.
- **B) Optimizer**: ADAM Learning Rate 0.001, momentum rate 0.9, weight decay(L2 regularization):1e-4.

6 DATASET

The Lung Image Database Consortium (LIDC) has made a database publically available that contains thoracic CT images of 1010 patients of lung cancers, and each scan has been annotated by up to 4

Algorithm 2: Clustering-augmented learning method

Step 0: Data obtained after extracting information using Convolutional Autoencoder(CAE) acts as input to CALM.

Step 1: Initialization of the cluster centers $u_1, ...u_{|C|}$ randomly. Clustering of the output data obtained from Convolutional Autoencoder(CAE) and augmenting each data sample with its one-hot encoded cluster label.

Step 2: Training the FNN classifier & clustering model

foreach $l \in \{1 \dots |C|\}$ **do**
Train the FNN model on data belonging to cluster l to learn classification.
For supervised training of the random forest classifier we use one hot encoded representation of clusters as labels.
Running the clustering model gives the cluster center u_l.

Step 3: Clustering

Update dissimilarity matrix using W^*

if *stopping criterion is attained* **then** Stop.
else go to **Step 2**.

radiologists on semantic characteristics and malignancy.The ratings were obtained by performing the biopsy, surgical resection, progression or reviewing the radiological images to show 2 years of nodule state at two levels; first at the patient level and second diagnosis at the second level. The LIDC database of thoracic CT studies for 1010 patients was acquired over a long period with various scanners.

We excluded nodules with outliers in x, y or z dimensions. Outliers are defined as values more than 1.5 times the interquartile range above the third quartile. We also excluded scans with slice thickness greater than 2.5 mm. This left 666 CT scans for training and 86 CT scans for evaluation. To reduce noise in our training data, we also exclude nodules by less than 3 radiologists.

The LIDC dataset also provides information and coordinates on each nodule. We chose an input size of 40×40 since that is large enough to fully contain the largest nodules. Classic data augmentation was performed on the positive examples: translations of up to 10 pixels in the XY plane are added to the positive training set. Negative data is defined as inputs that did not contain nodules agreed on by any radiologists. The final input data has 5422 image labels of size $40px \times 40px$. For comparison, the size of a whole CT scan is $512px \times 512px \times N$ slices, where N corresponds to the number of slices, ranging from [65,764] for different CT scans. The training and evaluation sets are randomly partitioned following proportion 8:2. Precisely, there are [0.8×5422]=4338 initial positive training examples, and since we want our initial training data to be balanced, we also take 4338 initial negative training examples of a practically infinite number available. In total, the initial training data consists of 8676(4338 positive + 4338 negative) training examples and 2168(1084 positive + 1084 negative) validation examples.

In this paper, to improve lung nodule detection in existing CADe systems, we augment training data-sets with generated images obtained using Generative Adversarial Network (GAN). We used an augmentation rate of 50% while using GANs. Since the original number of training samples is 4338 positive + 4338 negative= 8676, so the number of augmented data added is [4338*0.5]=2169 positive and [4338*0.5]=2169 negative samples. Since negative training volumes are easy to obtain, the WGAN-GP is trained on all of the positive training examples so that it will generate positive data.

Table 1: Performance of models

Model	Accuracy	Precision	Recall	F1	AUC
GAN+CAE+CALM (Proposed Model)	**95.3%**	**94.9%**	**95%**	**95%**	**0.97**
GAN+CAE+NN	94.2%	94.6%	93.5%	93.5%	0.93
GAN+CAE+LR	90.3%	92%	92%	92%	0.91
GAN+CAE+SVM	90.1%	92%	92%	92%	0.90
AE[9]	77%	76%	77%	77%	0.83
CNN	89%	88%	90%	89%	0.95

Where GAN represents Generative Adversarial Network, CAE represents Convolutional Autoencoder, CALM represents Clustering Augmented Learning Method, NN represents Neural Network, LR represents Logistic Regression, SVM represents linear kernel Support Vector Machine.

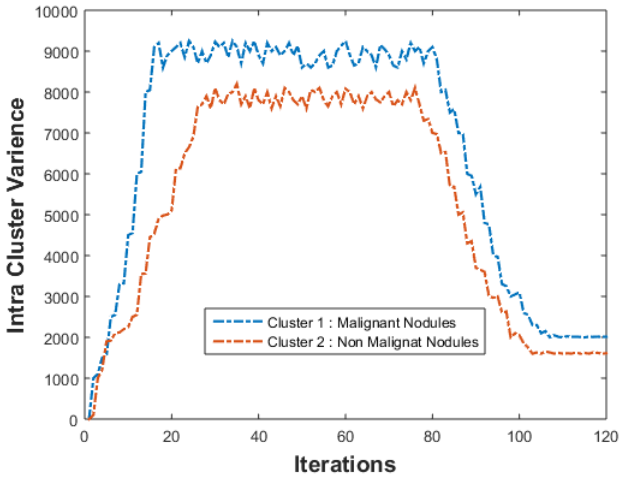

Figure 4: Plot of Intra-Cluster Variance vs Iterations

7 RESULT

The convolutional neural network for learning lung nodule image feature is similar to common image feature learning. Both CNN and conventional learning use the labeled dataset, and learn the network parameters between each layer from the input layer to the output layer by use of forwarding and backward propagation methods. We compare the classification performance of the proposed model, autoencoder(AE)[9], convolutional neural network(CNN) with the same dataset. Results are shown in Table(1) and Receiver Operating Characteristics Curve(ROC) is shown in Fig.6. To justify

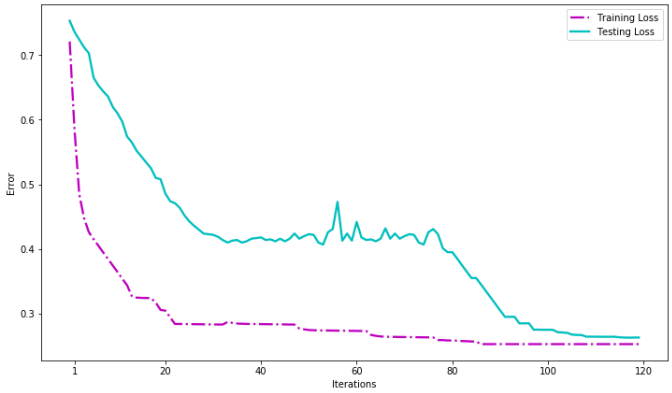

Figure 5: Training and Testing Loss

the contribution of the CALM classifier, we also compare the results by using traditional classifiers such as logistic regression, linear kernel support vector machine on the features obtained from the last pooling layer of the convolutional autoencoder. Moreover, Fig.4 shows how the intra-cluster variance decreases after approximately 75 iterations and then stabilizes. To measure intra-cluster variance, we used Euclidean distance in this case study. Similarly, it is evident from Fig. 5 that testing loss starts decreasing after 80 epochs and gradually as the clustering solution converges the accuracy begins to improve. This observation bolsters our initial assumption that clustering data based on inherent characteristics would improve the learning process of FNN.

The accuracy, precision, recall, F1, and AUC of the proposed method are 95.3%, 94.9%, 95%, 95% and 0.97 respectively. For AE (Autoencoder) method, we train the neural net in an unsupervised manner and test on the same dataset for classification. We use 1024 neurons in the fully connected layer in the AE method. We have also

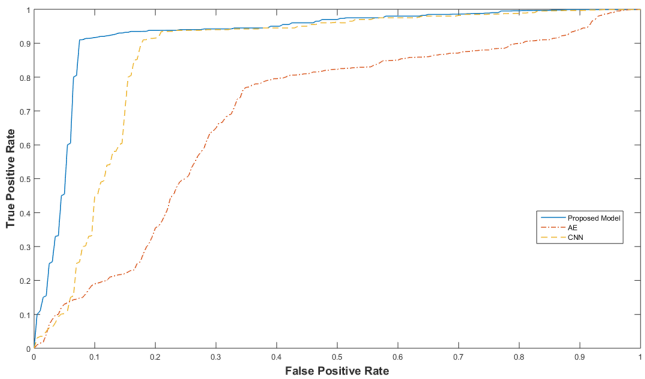

Figure 6: ROC on classification

compared the proposed model with accuracy obtained by previous literature. Comparison is shown in Table(2).

Table 2: Comparison with Literature

Model	Accuracy
Proposed model	**95.3%**
Kuruvilla and Gunavathi[10]	93.3%
Nascimento et al. [12]	92.78%
Krewer et al. [8]	90.91%
da Silva [16]	82.3%
Kumar et al. [9]	77%

8 CONCLUSION

In this paper, we present a novel approach to assist in CT image analysis. Approaches based on segmentation and handcrafted features are time-consuming and labor-intensive, while the data-driven approach is available to avoid the loss of important information in nodule segmentation. Methods based on Convolutional Neural Network(CNN) suffer from the scarcity of labeled data in the medical domain. To overcome that issue, in this paper, we propose the use of Generative Adversarial Networks to augment training data. We leverage Convolutional Autoencoder architecture for feature learning, in which the network is initially trained in an unsupervised manner with a large amount of data and later on the classifier is fine-tuned using a supervised approach. Referring to the result and the comparison table, our proposed system outperforms the literature mentioned in the related works section. In the future, we will work on amalgamating domain knowledge and data-driven feature learning.

REFERENCES

[1] A. A. Abdullah and S. M. Shaharum. 2012. Lung cancer cell classification method using artificial neural network. *Information Engineering Letters* 2, 1 (2012), 49–59.

[2] C Aliferis, I Tsamardinos, P Massion, P Fananapazir, D Hardin, A Statnikov, N Fananapazir, and D Hardin. 2003. Machine learning models for classification of lung cancer and selection of genomic markers using array gene expression data. In *FLAIRS*.

[3] Min Chen, Xiaobo Shi, Yin Zhang, Di Wu, and Guizani Mohsen. 2017. Deep Feature Learning for Medical Image Analysis with Convolutional Autoencoder Neural Network. *IEEE transactions on Big Data* (2017).

[4] M. J. M. Chuquicusma, S. Hussein, J. Burt, and U. Bagci. 2018. How to fool radiologists with generative adversarial networks? a visual turing test for lung cancer diagnosis. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE, 240–244.

[5] Carsten Eickhoff, Yubin Kim, and Ryen White. 2020. Overview of the Health Search and Data Mining (HSDM 2020) Workshop. In *Proceedings of the Thirteenth ACM International Conference on Web Search and Data Mining (WSDM '20)*. ACM, New York, NY, USA. https://doi.org/10.1145/3336191.3371879

[6] R. Fakoor, F. Ladhak, A. Nazi, and M. Huber. 2013. Using deep learning to enhance cancer diagnosis and classification. In *ICML Workshop on the Role of Machine Learning in Transforming Healthcare (WHEALTH)*. ICML, 4493–4498.

[7] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan. 2018. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *arXiv:1803.01229* (2018).

[8] H. Krewer, B. Geiger, and L. O. Hall. 2013. Effectoftexturefeatures in computer aided diagnosis of pulmonary nodules in low-dose computed tomography. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 3887–3891.

[9] D Kumar, A Wong, and D.A Clausi. 2015. Lung Nodule Classification using deep features in CT images. In *12th IEEE Conference on Computer and Robot Vision (CRV)*. IEEE, 133–138.

[10] J Kuruvilla and K Gunavathi. 2014. Lung cancer classification using neural networks for ct images. *Computer methods and programs in biomedicine* 113, 1 (2014), 202–209.

[11] L. Nanni, S. Brahnam, and A. Lumini. 2012. Combining multiple approaches for gene microarray classification. *Bioinformatics* (2012), 1151–1157.

[12] L. B. Nascimento, A. C. de Paiva, and A. C. Silva. 2012. Lung nodules classification in CT images using Shannon and Simpson diversity indices and SVM. In *Machine Learning and Data Mining in Pattern Recognition*. 454–466.

[13] S Ramaswamy, P Tamayo, R Rifkin, S Mukherjee, C Yeang, M Angelo, C Ladd, M Reich, E Latulippe, J.P Mesirov, T Poggio, W Gerald, M Loda, E.S Lander, , and T.R Golub. 2001. Multiclass cancer diagnosis using tumor gene expression signatures. In *National Academy of Sciences of the United States of America*.

[14] A. Riccardi, T. S. Petkov, G. Ferri, M. Masotti, and R. Campanini. 2011. Computer-aided detection of lung nodules via 3d fast radial transform, scale space representation, and zernike mip classification. *Medical physics* 38, 4 (2011), 1962–1971.

[15] A. Sharma, S. Imoto, and S Miyano. 2012. A top-r feature selection algorithm for microarray gene expression data. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* 9 (2012), 754–764.

[16] G. L. F. da Silva, A. C. Silva, A. C. de Paiva, and M. Gattass. [n.d.]. Lung nodules classification in CT images using Shannon and Simpson diversity indices and SVM. In *Machine Learning and Data Mining in Pattern Recognition*.

[17] H. I. Suk, S. W. Lee, D. Shen, and A. D. N. Initiative. 2014. Hierarchical feature representation and multimodal fusion with deep learning for ad/mci diagnosis. *NeuroImage* 101 (2014), 569–582.

[18] Y Wang, I.V Tetko, M.A Hall, E Frank, A Facius, K.F. X Mayer, and H.W Mewes. 2005. Gene selection from microarray data for cancer classification-a machine learning approach. *Comput. Biol. Chem.* 29, 1 (2005), 37–46.

[19] G. Wu, M. Kim, Q. Wang, Y. Gao, S. Liao, and D. Shen. 2013. Unsupervised deep feature learning for deformable registration of mr brain images. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013*. SPRINGER, 649–656.

[20] X. Zhu, Y. Liu, J. Li, T. Wan, and Z. Qin. 2018. Emotion classification with data augmentation using generative adversarial networks. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. SPRINGER, 349–360.

[21] D. Zinovev, J. Feigenbaum, J. Furst, and D Raicu. 2011. Probabilistic lung nodule classification with belief decision trees. In *Engineering in Medicine and Biology Society, EMBC*. IEEE, 4493–4498.