International Journal on Advances in Engineering Technology and Science
Volume: 5, Issue: 1, 2024, e-ISSN: 2455-3131
DOI: 10.5281/zenodo.10666856

Performance Analysis and Diagnosis of Thyroid Disease Detection using Deep Learning

Aparna Anand[1], Kirti Jain[2]

[1]UG Student, [2]Associate Professor
[1,2]School of Advanced Computing, Sanjeev Agrawal Global Educational University, Bhopal

Abstract—**Thyroid Disease is very common problem that affects lot of people around the world. That is why, it is important to diagnose this disease accurately and effectively. But it is not that, this disease has not been diagnosed yet. It is been done by doctors for long time. To facilitate doctors, I developed a deep learning model to identify Thyroid disease accurately. This is accomplished by providing dataset which was provided by Garavan Institute, located in Sydney [1]. The dataset comprises features related to thyroid conditions, where thyroid positivity being labelled as 0 and negativity as 1. This model involves densely connected layers which employs Rectified Linear Unit (ReLU) activation function. Techniques such as dropout and batch normalization are incorporated to enhance generalization and preventing overfitting. The Adam optimizer is utilized for gradient descent. The training process is monitored using metrics such as accuracy and loss, and model's performance is also evaluated using a separate test set. Standard evaluation metrics such as accuracy, precision, recall and F1_score is computed to access the model's predictions. We can figure out how good our computer is at recognizing different thyroid conditions with the help of confusion matrices. A classification threshold is explored to convert continuous predictions into binary outcomes. This model involves preprocessing steps such as feature scaling with the help of Min Max Scaler. Data Augmentation is done to enhance the model's ability. This model includes deep learning techniques such as neural networks, optimization algorithms and evaluation metrics to develop a robust binary classification model for thyroid disease diagnosis.**

I. INTRODUCTION

Thyroid disease is a common health problem that affects many people around the world. Providing accurate and effective diagnostic tools is critical. Over time, doctors have conducted extensive research on thyroid disease. However, advances in technology, especially in deep learning, may provide promising opportunities to improve the diagnostic process. In this study, we developed a deep learning model designed to detect thyroid disease more accurately.

Our initiative is derived from the realization that utilizing computational models can greatly assist doctors in their diagnosis. The developed deep learning model is intended to serve as a supplemental tool that aims to simplify and improve the process of diagnosing thyroid conditions. Through the use of artificial intelligence-powered technology, we are contributing to the ongoing pursuit of improved healthcare outcomes for individuals dealing with thyroid-related issues.

The dataset employed in this research, donated by the Garavan Institute in Sydney [1], includes a comprehensive variety of traits associated with thyroid conditions. Notably, the dataset is characterized by the presence of thyroid antibodies that are marked as 0 and those that are marked as 1. The utilization of a large dataset is crucial to the training and evaluation of our deep learning model, this will facilitate the accurate identification of thyroid diseases. The model's architecture is characterized by layers that are densely connected and employ the ReLU function as their activation function, this function is selected because of its effectiveness in recognizing complex patterns in the data. To improve the generalizability of the model and reduce overfitting, methods such as dropout and batch normalization are easily incorporated. The process of optimization utilizes the Adam optimizer to facilitate efficient gradient descent.

Monitoring the learning process, including key metrics such as accuracy and loss, provides insight into the learning dynamics of the model. In addition, the performance of this model is rigorously evaluated through a specific set of tests to ensure its generalizability and effectiveness in real world scenarios. Standard evaluation metrics such as precision, accuracy, recall, and F1_score were calculated to evaluate the predictive ability of the model. In addition, a confusion matrix was used to gain a deeper understanding of the model's ability to recognize different thyroid conditions. Classification threshold learning allows you to convert continuous predictions into binary results, further increasing the diagnostic accuracy of the model. Preprocessing steps, feature scaling and data augmentation techniques via Min-Max Scaler help improve model robustness and adaptability. By incorporating deep learning techniques such as neural networks, optimization algorithms, and evaluation metrics, our approach aims to develop a robust binary classification model for thyroid disease diagnosis.

II. LITERATURE REVIEW

Intelligent Diagnosis of Thyroid Ultrasound Imaging Using an Ensemble of Deep Learning Methods [2]. Thyroid disease has a high incidence in the world population, so it is necessary to

develop alternative methods to improve the diagnostic process. Materials and Methods: For this purpose, they developed an ensemble method that combines two deep learning models, one based on convolutional neural networks and the other based on transfer learning. For the first model, called 5-CNN, they developed an efficient final training model with five convolutional layers, and the second model, the pre-trained VGG-19 architecture was redesigned, optimized and trained. They trained and validated our model using a dataset of ultrasound images consisting of four types of thyroid imaging: autoimmune, nodular, micro-nodular, and normal.

Deep learning-based artificial intelligence model to assist thyroid nodule diagnosis and management: a multicentre diagnostic study. [3] Strategies for integrating artificial intelligence (AI) into thyroid nodule management require additional development and testing. We developed a deep-learning AI model (ThyNet) to differentiate between malignant tumours and benign thyroid nodules and aimed to investigate how ThyNet could help radiologists improve diagnostic performance and avoid unnecessary fine needle aspiration.

Thyroid Disease Treatment prediction with machine learning approaches [4]. This work, differently, aims to predict the LT4 treatment trend for patients suffering from hypothyroidism. To this end, a dedicated dataset was built that includes medical information related to patients being treated in the" AOU Federico II" hospital of Naples. For each patient, the clinical history is available over time, and therefore on the basis of the trend of the hormonal parameters and other attributes considered it was possible to predict the course of each patient's treatment in order to understand if this should be increased or decreased. To conduct this study, we used different machine learning algorithms. In particular, we compared the results of 10 different classifiers. The performances of the different algorithms show good results, especially in the case of the Extra-Tree Classifier, where the accuracy reaches 84%.

Deep learning to diagnose Hashimoto's thyroiditis from sonographic images [5]. Hashimoto's thyroiditis (HT) is the main cause of hypothyroidism. We develop a deep learning model called HTNet for diagnosis of HT by training on 106,513 thyroid ultrasound images from 17,934 patients and test its performance on 5051 patients from 2 datasets of static images and 1 dataset of video data. HTNet achieves an area under the receiver operating curve (AUC) of 0.905 (95% CI: 0.894 to 0.915), 0.888 (0.836–0.939) and 0.895 (0.862–0.927). HTNet exceeds radiologists' performance on accuracy (83.2% versus 79.8%; binomial test, $p < 0.001$) and sensitivity (82.6% versus 68.1%; $p < 0.001$). By integrating serologic markers with imaging data, the performance of HTNet was significantly and marginally improved on the video (AUC, 0.949 versus 0.888; DeLong's test, $p = 0.004$) and static-image (AUC, 0.914 versus 0.901; $p = 0.08$) testing sets, respectively. HTNet may be helpful as a tool for the management of HT.

Diagnosis method of Thyroid Disease combining knowledge graph and deep learning [6]. This research uses knowledge graph technology to connect trivial and scattered knowledge in various medical information systems to assist in disease diagnosis. This research takes thyroid disease as an example, constructs a medical knowledge graph and applies it to intelligent medical diagnosis. First, extract the relationships between biomedical entities to construct a biomedical knowledge graph. Then, the entities and relationships in the knowledge graph are transformed into low-dimensional continuous vectors through the knowledge graph embedding method. Finally, the known pathological disease relationship data is used to train the disease diagnosis model of the bidirectional long short-term memory network (BSTLM). Experiments show that the thyroid disease diagnosis method that combines knowledge graphs and deep learning has a better diagnostic effect. This shows that smart medical care based on the knowledge graph will provide a solution path for alleviating the shortage of domestic high-quality medical resources.

Detecting Six Different Types of Thyroid Diseases Using Deep Learning [7]. In this project, we apply deep learning algorithms to detect six different types of thyroid diseases and its presence without the need for several consultations from different doctors. This leads to earlier prediction of the presence of the disease and allows us to take prior actions immediately to avoid further consequences in an effective and cheap manner avoiding human error rate.

Online transfer learning for differential diagnosis of benign and malignant thyroid nodules with ultrasound images [8]. We aimed to propose a highly automatic and objective model named online transfer learning (OTL) for the differential diagnosis of benign and malignant thyroid nodules from ultrasound (US) images.

Automated segmentation of thyroid nodule, gland, and cystic components from ultrasound images using deep learning [9]. A prospective study was conducted, collecting data from 234 patients undergoing a thyroid ultrasound exam before biopsy. The training and validation sets encompassed 188 patients total; the testing set consisted of 48 patients. The algorithm effectively segmented thyroid anatomy into nodules, normal gland, and cystic components. The algorithm achieved a mean Dice coefficient of 0.76, a mean true positive fraction of 0.90, and a mean false positive fraction of 1.61×10^{-6}. The values are on par with a conventional seeded algorithm. The proposed algorithm eliminates the need for a seed in the segmentation process, thus automatically detecting and segmenting the thyroid nodules and cystic components. The detection rate for thyroid nodules and cystic components was 82% and 44%, respectively.

Multitask cascade convolution neural networks for automatic thyroid nodule detection and recognition [10]. In particular, we develop a multitask cascade convolution neural network (MC-CNN) framework to exploit the context information of thyroid nodules. It may be noted that our framework is built upon a large number of clinically confirmed thyroid ultrasound images with accurate and detailed ground truth labels. Other

key advantages of our framework result from a multitask cascade architecture, two stages of carefully designed deep convolution networks in order to detect and recognize thyroid nodules in a pyramidal fashion, and capturing various intrinsic features in a global-to-local way. Within our framework, the potential regions of interest after initial detection are further fed to the spatial pyramid augmented CNNs to embed multiscale discriminative information for fine-grained thyroid recognition. Experimental results on 4309 clinical ultrasound images have indicated that our MC-CNN is accurate and effective for both thyroid nodules detection and recognition. For the correct diagnosis rate of malignant and benign thyroid nodules, its mean Average Precision (mAP) performance can achieve up to 98.2% accuracy.

Diagnosis of thyroid disease using deep convolutional neural network models applied to thyroid scintigraphy images: a multicenter study [11]. The aim of this study was to improve the diagnostic performance of nuclear medicine physicians using a deep convolutional neural network (DCNN) model and validate the results with two multicenter datasets for thyroid disease by analyzing clinical single-photon emission computed tomography (SPECT) image data.

III. Materials and Methods

Dataset Information:

The Garavan Institute, located in Sydney, Australia, has provided six databases with documentation authored by Ross Quinlan. Each of these databases comprises approximately 2,800 instances for training data and 972 instances for testing, featuring a notable prevalence of missing data. The datasets consist of around 29 attributes, which are either Boolean or continuously-valued. Additionally, two databases, namely Hypothyroid data and sick-euthyroid data, also attributed to Ross Quinlan, are present, although Quinlan suspects potential corruption in their format despite their striking similarity to the other databases. There is an additional database encompassing 9,172 instances, spanning 20 classes, and accompanied by a related domain theory.

Furthermore, Stefan Aeberhard contributes another thyroid database with three classes, 215 instances, and five attributes, notable for its absence of missing values.

The dataset contains 3772 rows and 30 columns. This dataset was created by garavan institute which is located at Sydney. The columns contains information like age, on thyroxin, query on thyroxin, and many others.

The last column in my dataset was binaryclass, that was the target variable, that variable represents the output, that is whether a person is suffering from thyroid disease or not.

ut[4]:

	count	unique	top
age	3772	94	59
sex	3772	3	F
on thyroxine	3772	2	f
query on thyroxine	3772	2	f
on antithyroid medication	3772	2	f
sick	3772	2	f
pregnant	3772	2	f
thyroid surgery	3772	2	f
I131 treatment	3772	2	f
query hypothyroid	3772	2	f
query hyperthyroid	3772	2	f
lithium	3772	2	f
goitre	3772	2	f
tumor	3772	2	f
hypopituitary	3772	2	f
psych	3772	2	f
TSH measured	3772	2	t
TSH	3772	288	?
T3 measured	3772	2	t
T3	3772	70	?
TT4 measured	3772	2	t
TT4	3772	242	?
T4U measured	3772	2	t
T4U	3772	147	?
FTI measured	3772	2	t
FTI	3772	235	?
TBG measured	3772	1	f
TBG	3772	1	?
referral source	3772	5	other
binaryClass	3772	2	P

Binary Cross entropy Loss Function

Binary crossentropy, commonly called log loss, is a commonly used loss function in binary classification problems. In the context of the deep learning model, this loss function is used to divide the example into one of two classes, called the positive class (1) and the negative class (0).

In binary classification, each instance is assigned a label indicating whether it belongs to the positive class (1) or the negative class (0). The Binary Crossentropy loss quantifies the dissimilarity between the predicted probability distribution and the true distribution of the labels.

The formula for Binary Crossentropy loss is as follows:

$$L_{BCE} = -\frac{1}{n}\sum_{i=1}^{n}(Y_i \cdot \log \hat{Y}_i + (1 - Y_i) \cdot \log (1 - \hat{Y}_i))$$

where

yi – truel label for instance i

pi – predicted probability, for instance, i by the model

How to interpret Cross Entropy Loss?

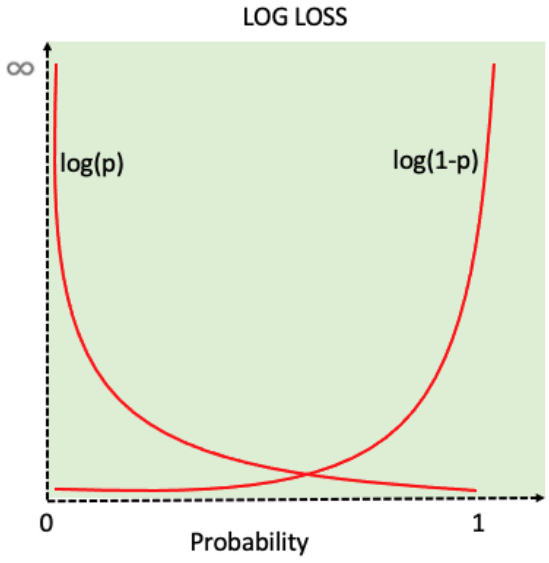
The cross-entropy loss is a scalar value that quantifies how far off the model's predictions are from the true labels. For each sample in the dataset, the cross-entropy loss reflects how well the model's prediction matches the true label. A lower loss for a sample indicates a more accurate prediction, while a higher loss suggests a larger discrepancy.

Interpretability with Binary Classification:

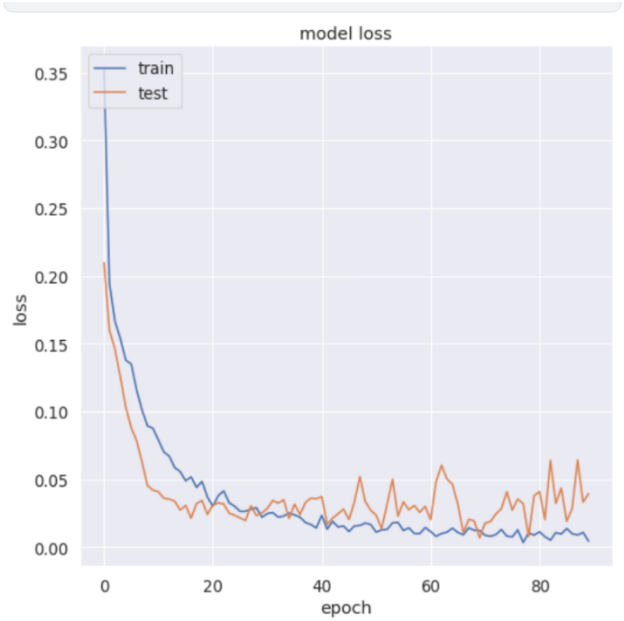
In binary classification, since there are two classes (0 and 1) it is start forward to interpret the loss value,

If the true label is 1, the loss is primarily influenced by how close the predicted probability for class 1 is to 1.0.

If the true label is 0, the loss is influenced by how close the predicted probability for class 1 is to 0.0.



In our Project, the binary cross entropy loss function appears in this way:



Feed Forward Neural Network Algorithm:

Feed-forward neural networks are a key component of this amazing technology, as they help software with pattern recognition and classification, linear regression, and function estimation. A feedforward neural network is a type of artificial neural network in which nodes' connections do not form a loop.

Often referred to as a multi-layered network of neurons, feedforward neural networks are so named because all information flows in a forward manner only.

The data enters the input nodes, travels through the hidden layers, and eventually exits the output nodes. The network is devoid of links that would allow the information exiting the output node to be sent back into the network.

The purpose of feedforward neural networks is to approximate functions.

Here's how it works

There is a classifier using the formula y = f* (x).

This assigns the value of input x to the category y.

The feed forward network will map y = f (x; θ). It then memorizes the value of θ that most closely approximates the function.

The following are the components of a feedforward neural network:

Layer of input

It contains the neurons that receive input. The data is subsequently passed on to the next tier. The input layer's total

number of neurons is equal to the number of variables in the dataset.

Hidden layer

This is the intermediate layer, which is concealed between the input and output layers. This layer has a large number of neurons that perform alterations on the inputs. They then communicate with the output layer.

Output layer

It is the last layer and is depending on the model's construction. Additionally, the output layer is the expected feature, as you are aware of the desired outcome.

Neurons weights

Weights are used to describe the strength of a connection between neurons. The range of a weight's value is from 0 to 1.

Cost Function in Feedforward Neural Network

The cost function is an important factor of a feedforward neural network. Generally, minor adjustments to weights and biases have little effect on the categorized data points. Thus, to determine a method for improving performance by making minor adjustments to weights and biases using a smooth cost function.

The mean square error cost function is defined as follows:

$$C(w,b) \equiv \frac{1}{2n} \sum_{x} \|y(x) - a\|^2.$$

Where,

w = weights collected in the network

b = biases

n = number of training inputs

a = output vectors

x = input

‖v‖ = usual length of vector v

dense (Dense): The first dense layer with 256 neurons. It has 7424 parameters, which includes weights and biases.

dropout (Dropout): Dropout layer after the first dense layer. It has no trainable parameters because dropout layers don't have weights.

dense_1 (Dense): The second dense layer with 128 neurons. It has 32,896 parameters.

dropout_1 (Dropout): Dropout layer after the second dense layer.

dense_2 (Dense): The third dense layer with 63 neurons. It has 8,127 parameters.

dropout_2 (Dropout): Dropout layer after the third dense layer.

dense_3 (Dense): The final dense layer with 1 neuron (output layer). It has 64 parameters.

Total params: The total number of trainable parameters in the model, which is the sum of parameters in all layers.

Trainable params: The number of parameters that will be updated during training.

Non-trainable params: The number of parameters that are not trainable. In this case, it's zero, indicating that all parameters are trainable.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 256)               7424
_________________________________________________________________
dropout (Dropout)            (None, 256)               0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               32896
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 63)                8127
_________________________________________________________________
dropout_2 (Dropout)          (None, 63)                0
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 64
=================================================================
Total params: 48,511
Trainable params: 48,511
Non-trainable params: 0
_________________________________________________________________
```

Sigmoid Activation Function:

The sigmoid function is a special form of the logistic function and is usually denoted by σ(x) or sig(x). It is given by:

σ(x) = 1/(1+exp(-x))

The graph of sigmoid function is an S-shaped curve as shown by the green line in the graph below. The figure also shows the graph of the derivative in pink color. The expression for the derivative, along with some important properties are shown on the right. [12]

Domain: (-∞, +∞)

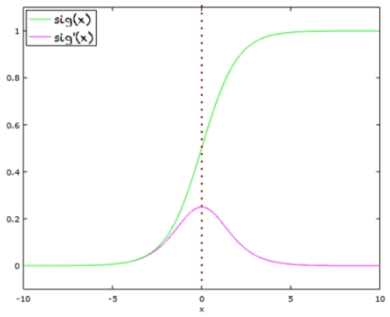
Range: (0, +1)

σ(0) = 0.5

The function is monotonically increasing.

The function is continuous everywhere.

The function is differentiable everywhere in its domain.



Domain: (−∞, +∞)
Range: (0, +1)
σ(0) = 0.5

Other properties

$$\sigma(x) = 1 - \sigma(-x)$$

$$\sigma(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1}$$

$$\sigma'(x) = \sigma(x)(1-\sigma(x))$$

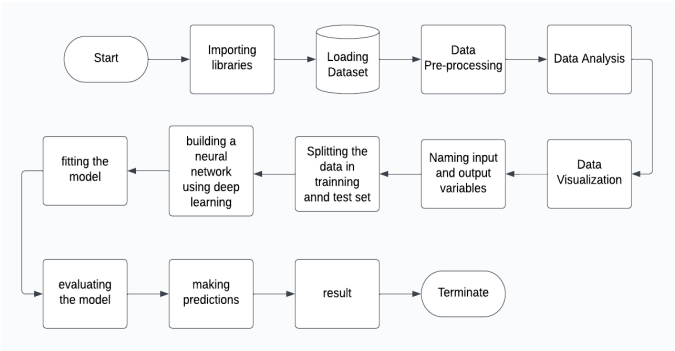
Plot of σ(x) and its derivate σ'(x)

Numerically, it is enough to compute this function's value over a small range of numbers, e.g., [-10, +10]. For values less than -10, the function's value is almost zero. For values greater than 10, the function's values are almost one.

This function divides the raw product of the model into the range [0, 1] and interprets it as a probability. In the context of binary classification, the sigmoid function is applied in the final layer of the model and the output samples are considered to belong to the positive class.

In summary, binary crossentropy loss, and a sigmoid activation function in the output layer, is a common combination for training deep learning models on binary classification problems. This allows us to learn and optimize the parameters to make accurate predictions for examples belonging to one of the two classes.

Flow chart of methodology followed:



First of all, I have imported all the basic libraries required for making my project such as numpy, pandas, matplotlib, statsmodels and seaborn and some deep learning libraries such as tensorflow, keras, in my environment which was Kaggle.

Then I have loaded my dataset using pandas. The dataset contains 3772 rows and 30 columns. This dataset was created by garavan institute which is located at Sydney. The columns contain information like age, on thyroxin, query on thyroxin, and many others. The last column in my dataset was binaryclass, that was the target variable, that variable represents the output, that is whether a person is suffering from thyroid disease or not. Then I performed some basic data pre-processing practices to know my dataset such as top 10 rows of my data, any missing or null values in my data or not, shape and description of the dataset.My dataset contained some missing and null values, and there was also some variables which contained numeric values but having datatype object. I changed the datatypes to numeric datatype. Then, my output column, named binaryclass contained two types of values, positive and negative, in which 3481 were positive and 291 were negative. So, in my case, it was a binary classification problem. Then I performed some data analysis, such as how many thyroid positives and negatives were there, and how many of them were pregnant, how were not.

Some columns contained missing values. I have used simpleImputer class from scikit-learn library to handle the missing values. Then I performed some data visualization to know my data effectively. Then I founded some correlations in between my data variables and created a heatmap of those correlations. Then I started preparing data for machine learning. Named the input data as x and output variable as y. Then I have used scikit learn to split up my data in training and test sets. Random state for my data was 42. Then I normalized my data using standardscaler from scikit learn library. Then I started building the neural networks using deep learning libraries, keras and tensorflow. I have used libraries from these two main libraries which are, sequential from models, dense, dropout from from layers, adam from optimizer, reducedLROnPlateau, modelcheckpoint, earlystopping.

Evaluation Metrics:

Classification accuracy is what we usually mean when we use the term accuracy. We calculate this by calculating the ratio of correct guesses to the total number of input samples.

$$\text{Accuracy} = \frac{\text{No. of correct predictions}}{\text{Total number of input samples}}$$

It works best if there is an equal number of samples for each class. For example, we have a sample of 90% class A and 10% class B in our training set. Then, our model predicts all training samples belonging to class A with 90% accuracy, if we test the same model with a test set of 60% from class A and 40% from class B, then the accuracy will fall and we get 60% accuracy.

Classification accuracy is good, but tests to achieve high accuracy give positive feedback. The problem arises from the fact that the probability of subclass samples is too high.

IV. RESULTS

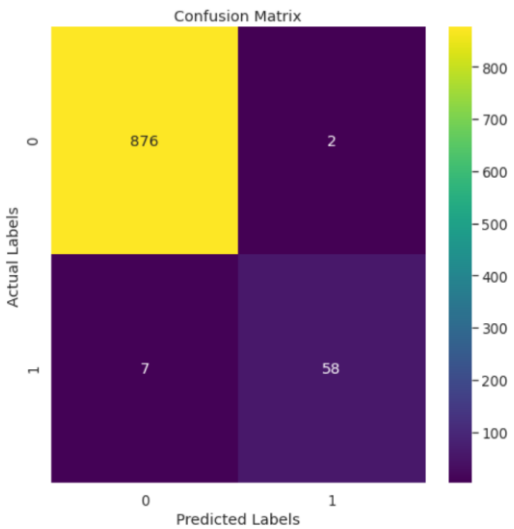
The model evaluate function is used to evaluate the model trained in the test database (x_test and y_test). The evaluation results include two values:

1. Loss: 0.0542

2. Accuracy: 98.94%

Continuous entries in y_test will be converted to binary values (0 or 1) using a threshold of 0.5. If the value is greater than or equal to 0.5, it is equal to 1; Otherwise, it is set to 0. Model accuracy is calculated by comparing binary ground truth labels (binary_labels) and binary predictions (predictions_binary). The classification report provides precision, recall, F1-score, and support (0 and 1) for each class, as well as the average macro.

The confusion matrix shows the number of true positives, true negatives, false positives, and false negatives.



Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	0.99	878
1	0.97	0.89	0.93	65
accuracy			0.99	943
macro avg	0.98	0.95	0.96	943
weighted avg	0.99	0.99	0.99	943

V. CONCLUSION

I faced several challenges that added complexity to the project. In particular, having a balanced database poses an initial challenge due to the imbalance of data in thyroid disease.I solved this problem by using data augmentation techniques to improve the robustness of the model. Furthermore, fine-tuning the hyperparameters to achieve optimal model performance requires iterative tuning, adding to the overall complexity. Despite these obstacles, the success of the project is a testament to the effectiveness of our problem-solving approach.

In conclusion, this study presented a reliable deep learning model to solve important problems in healthcare for the accurate diagnosis of thyroid disease. This model, developed on the database provided by the Garavan Institute, shows high accuracy and efficiency in distinguishing thyroid positives and negatives. Our study contributes to this field by using advanced deep learning techniques including layers closely related to ReLU activation, degradation, and party regulation. Human optimizers and standard evaluation metrics such as accuracy, precision, recall and F1 score are used, which comprehensively evaluates the performance of the model.

Comparison with conventional diagnostic methods shows the advantages of our deep learning approach, showing its potential to revolutionize the diagnosis of thyroid disease. The use of pre-processing steps such as feature scaling and data augmentation with Min-Max Scaler further improves the robustness of the model. Looking ahead, future technologies involve refining and expanding the proposed model, taking into account emerging technologies and databases. The ultimate goal is to integrate this model into real clinical settings to contribute to the early and accurate diagnosis of thyroid disease.

REFERENCES

[1] https://archive.ics.uci.edu/dataset/102/thyroid+disease
[2] Hui Yu, Jinqiu Li, Jinglai Sun, Jie Zheng, Shuo Wang, Guangpu Wan g, Yongzheng Ding, Jing Zha,Jie Zhang Biomedical Signal Processing and Control Volume 78, September 2022, 103924
[3] Kirti Jain, Pinaki Ghosh, Shital Gupta, "A Hybrid Model for Sentiment Analysis Based on Movie Review Datasets" in international journal on recent and innovation trends in computing and communication, Scopus Indexed Journal ISSN: 2321-8169, Vol 11 No 5s(2023).
[4] Kirti Jain, "A Transfer Learning Based Machine Learning Approach to Solve Problems of Ecommerce: Image Search" published in Proceedings of International Joint Conference on Advances in Computational Intelligence indexed on SpringerBook series ISBN 978-981-99-1435-7, June 2023
[5] Prof Sui Peng, PhD, Prof Weiming Lv, PhD Deep learning-based artificial intelligence model to assist thyroid nodule diagnosis and management: a multicentre diagnostic study VOLUME 3, ISSUE 4, E250-E259, APRIL 2021
[6] Lerina Aversano, Mario
Luca Bernardi, Marta Cimitile, Martina Iammarino, Paolo
Emidio Macchia, Immacolata Cristina Nettore, Chiara Verdone Thyroid Disease Treatment prediction with machine learning approaches Volume 192, 2021, Pages 1031-1040
[7] Qiang Zhang Deep learning to diagnose Hashimoto's thyroiditis from sonographic images Published: 29 June 2022 Article number: 3759 (2022)

[8] Xuqing Chai Diagnosis Method of Thyroid Disease Combining Knowledge Graph and Deep Learning 14 August 2020 Published in: IEEE Access (Volume: 8)

[9] R. Mageshwari, G. Sri Kiruthika, T. Suwathi Tina Susan Thomas Detecting Six Different Types of Thyroid Diseases Using Deep Learning 03 March 2022 LNNS, volume 339

[10] Hui Zhou, Kun Wang, Jie Tian Online Transfer Learning for Differential Diagnosis of Benign and Malignant Thyroid Nodules With Ultrasound Images, PMID: 32011998, DOI: 10.1109/TBME.2020.2971065

[11] Viksit Kumar, Jeremy Webb, Adriana Gregory, Duane D. Meixner, John M. Knudsen Automated Segmentation of Thyroid Nodule, Gland, and Cystic Components from Ultrasound Images Using Deep Learning Article number 9044381 volume 8 published in 2020

[12] Wei Li, Shuangshuang Wu, Haoran Zhang Enhanced Biological Photosynthetic Efficiency Using Light-Harvesting Engineering with Dual-Emissive Carbon Dots Volume 28 First published: 14 September 2018

[13] Huayi Zhao, Chenxi Zheng, Huihui Zhang Diagnosis of thyroid disease using deep convolutional neural network models applied to thyroid scintigraphy images: a multicenter study PMID: 37635985

[14] Mehreen Saeed A Gentle Introduction to Sigmoid Function, August 18, 2021