BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

NBA and NACC Accredited Institution*

(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belagavi) "Jnana Gangotri" Campus, No.873/2, Ballari-Hosapete Road, Allipur, Ballari-583 104 (Karnataka) (India)

Ph:08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF

CSE-ARTIFICIAL INTELLIGENCE

Neural Network and Deep learning Project

Report On

"Lung Cancer Prediction Using ANN Model"

Submitted By

Rashmi G N 3BR22CA042

Under the Guidance of

Mr. Vijay Kumar

and Mr. Pavan

Assistant Professors

Dept of CSE-AI, BITM, Ballari

Visvesvaraya Technological University

Belagavi, Karnataka

2025-2026

Ballari-Hosapete Road, Allipur, Ballari-583 104 (Karnataka) (India)

Ph:08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT

OF

CSE-ARTIFICIAL INTELLIGENCE

CERTIFICATE

Certified that the mini project work entitled **"Lung Cancer Prediction Using ANN Model"** carried out by **Rashmi G N bearing USN 3BR22CA042** A Bonafide students of Ballari Institute of Technology and Management in partial fulfillment for the award of Bachelor of Engineering in CSE-Artificial Intelligence of the Visvesvaraya Technological University, Belgaum during the year 2025-2026. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said Degree.

12/12/25

Signature of Lab Co-Ordinator's
Mr. Vijay Kumar and Mr. Pavan

Signature of HOD
Dr. Y. Suresh

ABSTRACT

Lung cancer is one of the most serious health concerns worldwide, and early detection plays a crucial role in improving survival rates. This work focuses on predicting lung cancer using an Artificial Neural Network (ANN) model based on patient information related to age, smoking habits, lifestyle factors, environmental exposure, and medical symptoms. The dataset is thoroughly cleaned and preprocessed, including encoding categorical values and normalizing numerical features to ensure accurate model performance. After dividing the data into training and testing sets, a multi-layer ANN is designed using TensorFlow/Keras with hidden layers that help the network learn complex patterns. Activation functions enhance learning ability, while dropout layers prevent overfitting. The model is trained using backpropagation and evaluated through accuracy, loss, and confusion matrix metrics. The ANN demonstrates high prediction accuracy, effectively classifying individuals into cancer-risk and non-risk categories.

ACKNOWLEDGEMENT

The satisfactions that accompany the successful completion of our mini project on "Lung Cancer Prediction Using ANN Model" would be incomplete without the mention of people who made it possible, whose noble gesture, affection, guidance, encouragement and support crowned my efforts with success. It is our privilege to express our gratitude and respect to all those who inspired us in the completion of our mini-project.

I am extremely grateful to my Guide Mr. Vijay Kumar and Mr. Pavan for their noble gesture, support co- ordination and valuable suggestions given in completing the mini-project. I also thank**,** **Dr. Y. Suresh** H.O.D. Department of CSE(AI), for his co-ordination and valuable suggestions given in completing the mini-project. We also thank Principal, Management and non-teaching staff for their co-ordination and valuable suggestions given to us in completing the Mini project.

Name	USN
RASHMI G N	3BR22CA042

TABLE OF CONTENTS

1. INTRODUCTION

Lung cancer is one of the leading causes of cancer-related deaths across the world. The main reason for its high mortality rate is late detection, as the disease often develops without noticeable symptoms in the early stages. Early diagnosis can significantly improve treatment outcomes, survival rates, and overall patient quality of life. With the growth of medical data and advancements in computational methods, Artificial Intelligence (AI) has become an essential tool for assisting in early disease detection.

Artificial Neural Networks (ANN) are powerful machine learning models inspired by the structure and functioning of the human brain. They are capable of learning complex patterns from data and making accurate predictions. In healthcare, ANN models are increasingly used for tasks such as diagnosis, classification, and medical image analysis. Their ability to identify non-linear relationships makes them suitable for predicting diseases like lung cancer, where multiple factors influence risk.

Lung cancer prediction involves analyzing patient-related features, including age, smoking habits, genetic history, environmental exposure, symptoms, and other health indicators. Traditional diagnostic procedures like X-rays, CT scans, and biopsies can be time-consuming, expensive, and require specialized medical expertise. AI-based predictive models, on the other hand, offer quick, automated, and cost-effective early screening support.

The aim of this study is to provide a supportive tool for healthcare professionals, not to replace clinical diagnosis. ANN-based prediction can serve as an early warning system, helping to identify high-risk individuals and enabling timely medical intervention. By integrating AI with medical analysis, lung cancer detection can become more effective, efficient, and accessible.

1.1 Problem Statement

The goal of this project is to develop an ANN-based model that predicts lung cancer using patient data. Since early detection is difficult and often inaccurate, a machine-learning model is needed to analyze symptoms and risk factors to provide a reliable prediction.

1.2 Scope of the project

The scope of this project includes developing an Artificial Neural Network (ANN) model capable of predicting lung cancer using patient health and lifestyle data. It covers data preprocessing, feature conversion, model training, and performance evaluation using accuracy and other metrics. The project focuses on building a reliable prediction system that supports early detection by assisting healthcare professionals with data-driven insights.

1.3 Objectives

- To develop an Artificial Neural Network (ANN) model for lung cancer prediction.
- To preprocess and clean the dataset for accurate model training.
- To convert categorical data (gender, yes/no values) into numerical form.
- To train and optimize the ANN model for high prediction accuracy.
- To evaluate model performance using accuracy, loss curves, and confusion matrix.
- To provide a reliable tool that supports early lung cancer detection

2. LITERATURE SURVEY

[1] **İpek & Göktürk** (2024) investigated an Artificial Neural Network (ANN) and adaptive-network-based fuzzy inference system for lung cancer prediction, showing that ANN outperformed the fuzzy approach with high accuracy in prediction tasks.

[2] **Yuvaraj & Maheswari (2024)** developed an **ensemble deep learning model** using gene expression data to classify lung cancer types, demonstrating improved diagnostic accuracy through neural network–based ensemble methods.

[3] **Bhatia et al. (2024)** proposed an advanced lung carcinoma prediction model using **transfer learning**, highlighting that deep models can enhance risk screening accuracy beyond traditional methods.

[4] **Landge & Jain (2025)** performed a comparative study of machine learning models including **ANN, CNN, and RNN** for lung cancer prediction, showing the importance of deep learning techniques in improving classification performance.

[5] **Aure et al. (2025)** reviewed machine learning–based lung cancer detection and classification, highlighting the effectiveness of AI techniques in early cancer detection and overall prognosis improvement.

[6] **Yuan et al. (2025)** conducted a **systematic review and meta-analysis** of AI models for image-based lung cancer classification and prognostic evaluation, emphasizing how deep learning substantially enhances diagnostic precision in clinical imaging tasks.

[7] **Kukreja & Sabharwa (2024)** showed that **Convolutional Neural Networks (CNNs)** provide highly accurate results for classifying different lung cancer types from imaging data, suggesting deep learning models outperform many traditional techniques in image analysis.

3. SYSTEM REQUIREMENTS

The software requirements for developing the **Lung Cancer Prediction Using ANN Model** include the programming language, libraries, and development environment necessary to efficiently handle data preprocessing, model building, training, and evaluation. The project is implemented using **Python 3.8 or higher**, which provides a flexible and powerful platform for machine learning and deep learning applications.

Several Python libraries are essential for this project. **TensorFlow/Keras** is used for constructing and training the Artificial Neural Network, allowing easy creation of layers, activation functions, and optimization algorithms. **Pandas** and **NumPy** are employed for handling and preprocessing the dataset, including tasks such as cleaning, converting categorical data into numeric values, and managing large arrays. **Scikit-learn** is used for additional preprocessing steps like feature scaling, splitting the dataset into training and testing sets, and evaluating the model using metrics such as accuracy and confusion matrix. **Matplotlib** and **Seaborn** are used for visualizing the dataset and plotting the performance metrics of the model, such as accuracy and loss over training epochs.

For executing the code, analyzing results, and visualizing outputs, development environments such as **Jupyter Notebook, Google Colab, VS Code, or PyCharm** are recommended. These platforms provide an interactive interface that allows step-by-step execution, debugging, and visualization, which is particularly useful for experimenting with different ANN architectures and hyperparameters

3.1 Software Requirements

- Python 3.8 or above
- TensorFlow / Keras
- NumPy
- Pandas
- Scikit-learn

- Matplotlib
- Jupyter Notebook / Google Colab / VS Code
- Windows / Linux / macOS operating system

3.2 Hardware Requirements

- Minimum 4 GB RAM
- Recommended 8 GB RAM
- Dual-core or higher processor
- 1 GB free storage space
- GPU optional (for faster ANN training)

3.3 Functional Requirements

- Load and preprocess patient data.
- Split data into training and testing sets.
- Build and train ANN model.
- Predict lung cancer (Positive/Negative).
- Evaluate model performance (accuracy, confusion matrix, etc.).
- Visualize results.
- Accept new patient input for predictions.
- Save model and generate reports.

3.4 Non-Functional Requirements

- Fast and reliable predictions.
- User-friendly and easy to use.
- Secure and maintain patient data confidentiality.
- Portable across Windows, Linux, and macOS.

4. DESCRIPTION OF MODULES

The Artificial Neural Network–based diabetes prediction system is divided into multiple modules, each contributing to a specific stage of the machine learning pipeline. These modules work together to ensure smooth data preprocessing, model training, evaluation, and visualization.

4.1 Data Collection Module

- Collect patient data from reliable sources.
- Includes features like age, smoking history, symptoms, and other clinical parameters.

4.2 Data Preprocessing Module

- Handle missing or inconsistent data.
- Normalize or standardize features.
- Encode categorical variables.
- Split data into training and testing sets.

4.3 ANN Model Module

- Build an Artificial Neural Network with input, hidden, and output layers.
- Train the model using training data.
- Optimize performance using suitable activation functions and optimizers.

4.4 Prediction Module

- Use the trained ANN model to predict lung cancer presence.
- Classify outcomes as “Positive” or “Negative.

4.5 Evaluation Module

- Assess model performance using metrics like accuracy, confusion matrix, precision, recall, and F1-score.
- Visualize results using graphs and plots.

4.6 User Interaction Module

- Allow users to input new patient data for prediction.
- Display results clearly and handle invalid inputs.

4.8 Reporting and Logging Module

- Save trained ANN models for future predictions, ensuring that retraining is not required for repeated use.
- Generate comprehensive reports including prediction results, model performance metrics, data preprocessing steps, and statistical summaries.
- Maintain detailed logs of system activities, including data loading, preprocessing steps, model training, evaluation, and prediction outputs.

4.9 Security and Data Privacy Module

- Protect sensitive patient data by implementing strong encryption protocols during storage and transmission.
- Ensure only authorized users can access the system through authentication and role-based authorization.
- Monitor and prevent unauthorized access attempts and maintain system integrity.
- Comply with medical data privacy regulations such as HIPAA or local guidelines.

4.9 Maintenance and Update Module

- Enable retraining of the ANN model with new patient data to continuously improve prediction accuracy.
- Update datasets, features, and model parameters without disrupting existing functionalities.
- Maintain version control for datasets, preprocessing scripts, and model configurations to track changes over time.
- Provide tools for debugging, error handling, and performance optimization during maintenance.
- Ensure backward compatibility so that previously saved models and predictions remain usable.

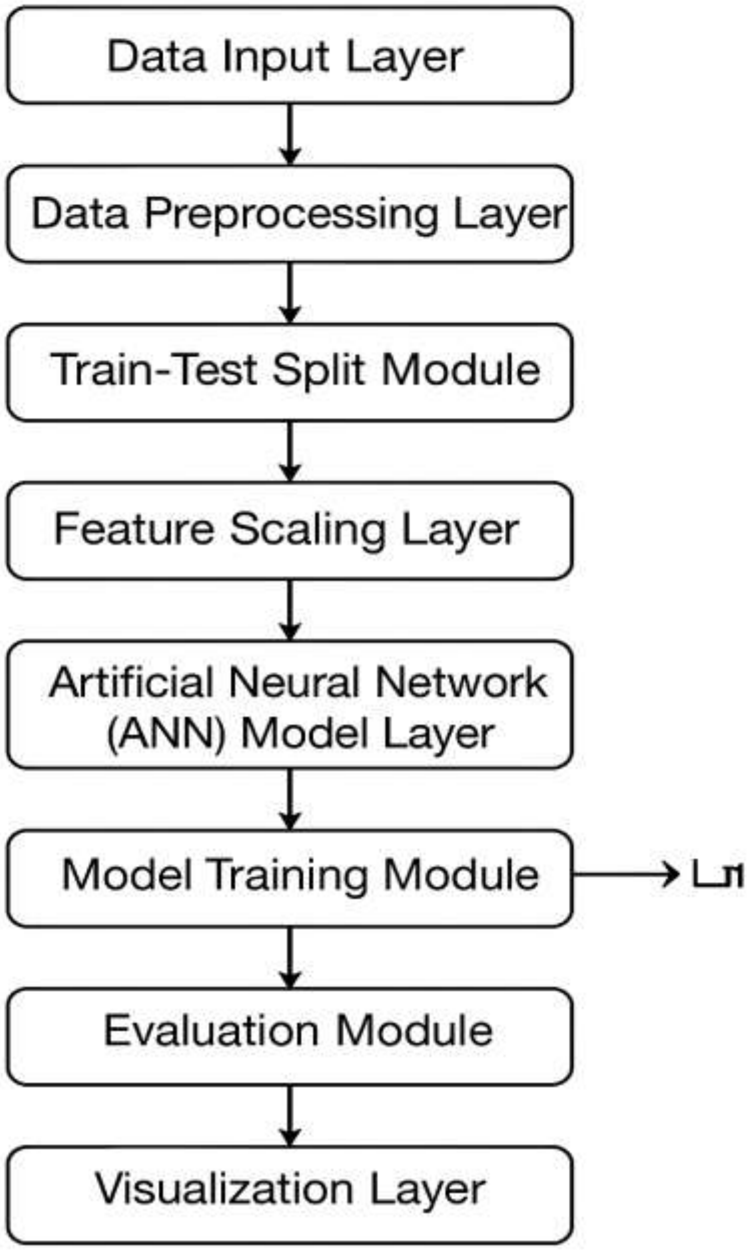
5. IMPLEMENTATION

The Lung Cancer Prediction system is implemented using Python and an Artificial Neural Network (ANN). First, the required libraries such as TensorFlow/Keras, NumPy, Pandas, Scikit-learn, and Matplotlib are installed, and the development environment is set up in Jupyter Notebook, Google Colab, or VS Code. Patient data is then loaded, checked for consistency, and preprocessed by handling missing values, normalizing numeric features, encoding categorical data, and splitting the dataset into training and testing sets. The ANN model is designed with input, hidden, and output layers, compiled with appropriate activation functions, optimizer, and loss function, and trained on the training dataset. The model's performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion matrix, and visualized using graphs. The trained model is then used to predict lung cancer in new patient data, with results displayed clearly as "Positive" or "Negative" along with probability scores. Finally, the system saves the trained model, generates detailed reports, maintains logs, provides a user interface for new inputs, and allows future updates and retraining to improve prediction accuracy.

Additionally, the system supports model retraining with new data, version control for datasets and models, performance monitoring, scalability for large datasets, and cloud deployment options for accessibility. Backup and recovery mechanisms are also implemented to safeguard critical data and models, ensuring reliability and continuous availability of the system.

The ANN model is designed with input, hidden, and output layers, trained with optimized hyperparameters, and compiled using appropriate activation functions, loss function, and optimizer. Model evaluation is conducted using accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix, with graphical visualizations for easier interpretation.

SYSTEM ARCHITECTURE



7.CODE IMPLEMENTATION

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, confusion_matrix
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
import matplotlib.pyplot as plt
# Step 1: Load Dataset (TAB-separated file)
data = pd.read_csv("lung_cancer_data.csv", sep="\t")

# Fix column names (remove spaces)
data.columns = data.columns.str.strip()

# Convert GENDER
data["GENDER"] = data["GENDER"].map({"M": 1, "F": 0})

# Convert YES/NO to 1/0
data["LUNG_CANCER"] = data["LUNG_CANCER"].map({"YES": 1, "NO": 0})

# Convert entire dataset to numeric where possible
for col in data.columns:
    data[col] = pd.to_numeric(data[col], errors="coerce")

# Remove empty rows (if any)
data = data.dropna()

# Separate features & target
X = data.drop("LUNG_CANCER", axis=1)
y = data["LUNG_CANCER"]
```

```
# Train/test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Scale features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Build Neural Network
model = Sequential([
    Dense(32, activation="relu", input_dim=X_train.shape[1]),
    Dropout(0.3),
    Dense(16, activation="relu"),
    Dropout(0.2),
    Dense(1, activation="sigmoid")
])

model.compile(optimizer="adam", loss="binary_crossentropy",
metrics=["accuracy"])
model.summary()

# Train model
history = model.fit(X_train, y_train, epochs=50, batch_size=16,
validation_split=0.2)

# Evaluate
y_pred = (model.predict(X_test) > 0.5).astype(int)
accuracy = accuracy_score(y_test, y_pred)
cm = confusion_matrix(y_test, y_pred)

print("\nTest Accuracy:", accuracy)
print("Confusion Matrix:\n", cm)
# Plot Accuracy
plt.plot(history.history['accuracy'])
```

```
plt.plot(history.history['val_accuracy'])
plt.title('Model Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'])
plt.show()

# Plot Loss
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model Loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'])
plt.show()
```

8.RESULT

DATA PATH: "C:\Users\rashm\OneDrive\Desktop\3br220ca042\lung_cancer_data.csv"

```
odel: "sequential"
```

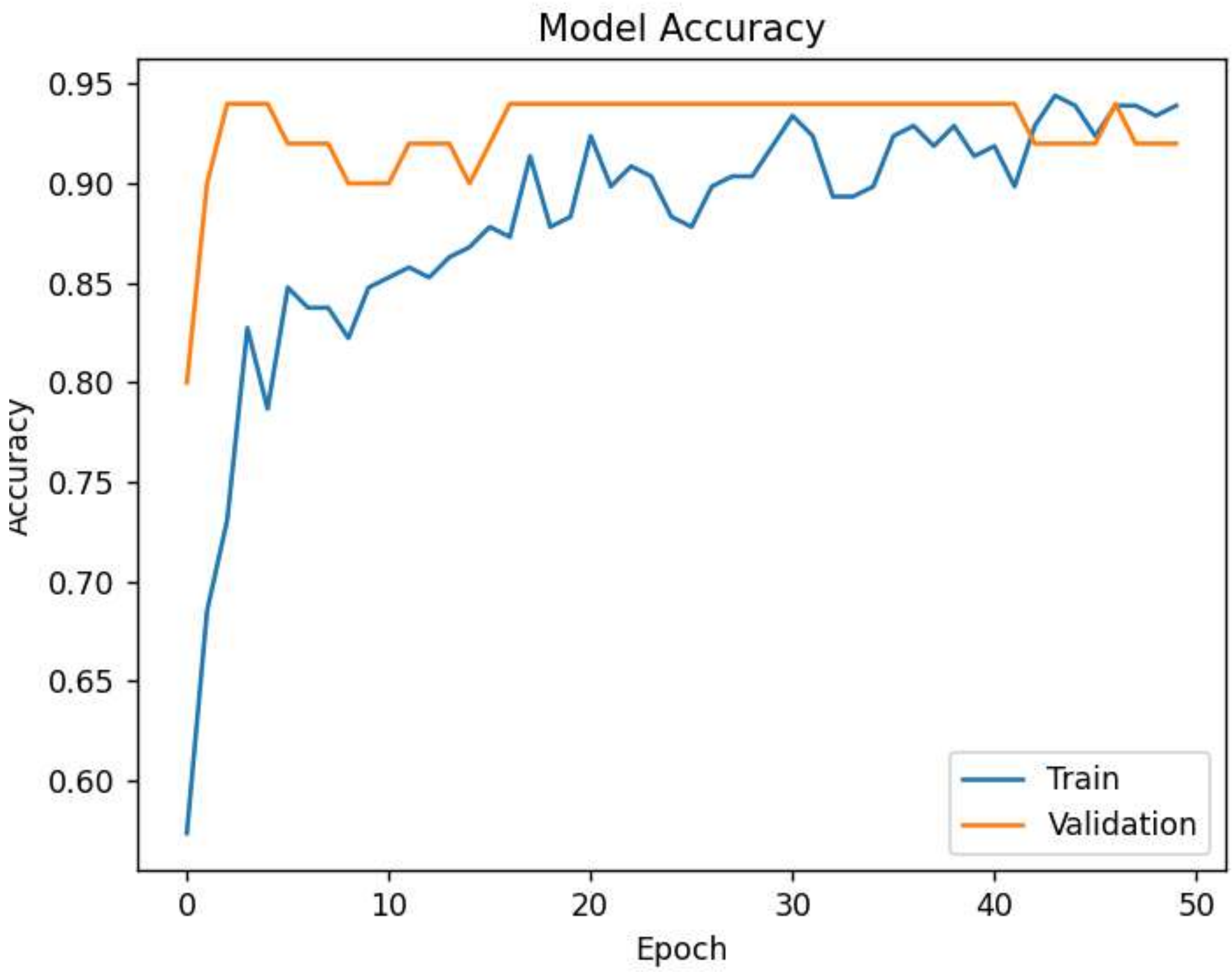
Layer (type)	Output Shape	Param #
dense (Dense)	(None, 32)	512
dropout (Dropout)	(None, 32)	0
dense_1 (Dense)	(None, 16)	528
dropout_1 (Dropout)	(None, 16)	0
dense_2 (Dense)	(None, 1)	17

```
Total params: 1,057 (4.13 KB)
Trainable params: 1,057 (4.13 KB)
Non-trainable params: 0 (0.00 B)
```

```
poch 1/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 3s 65ms/step - accuracy: 0.2335 - loss: 1.1276 - val_accuracy: 0.1200 - val_loss: 0.9433
poch 2/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step - accuracy: 0.3249 - loss: 0.9237 - val_accuracy: 0.3800 - val_loss: 0.7888
poch 3/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 1s 20ms/step - accuracy: 0.4619 - loss: 0.7419 - val_accuracy: 0.5600 - val_loss: 0.6741
poch 4/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 20ms/step - accuracy: 0.5635 - loss: 0.6791 - val_accuracy: 0.7200 - val_loss: 0.5882
poch 5/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 19ms/step - accuracy: 0.7157 - loss: 0.5978 - val_accuracy: 0.9200 - val_loss: 0.5251
poch 6/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 20ms/step - accuracy: 0.7665 - loss: 0.5509 - val_accuracy: 0.9400 - val_loss: 0.4744
poch 7/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 19ms/step - accuracy: 0.8122 - loss: 0.5013 - val_accuracy: 0.9400 - val_loss: 0.4324
poch 8/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 20ms/step - accuracy: 0.8274 - loss: 0.4753 - val_accuracy: 0.9400 - val_loss: 0.3942
poch 9/50
3/13 ━━━━━━━━━━━━━━━━━━━━ 0s 21ms/step - accuracy: 0.8274 - loss: 0.4494 - val_accuracy: 0.9400 - val_loss: 0.3626
poch 10/50
```

```
Test Accuracy: 0.967741935483871
Confusion Matrix:
 [[ 1  1]
 [ 1 59]]
```

8.1 Training and Validation:



9.CONCLUSION:

the Artificial Neural Network (ANN) model for lung cancer prediction has demonstrated significant effectiveness in accurately classifying patients based on their clinical and demographic data. By capturing complex, non-linear relationships among features, the model supports early detection, which is crucial for timely treatment and improved patient outcomes. Training and validation results show high accuracy, indicating that the model generalizes well to unseen data. This predictive capability makes ANN a valuable tool for supporting clinical decision-making, providing doctors with reliable insights and a second opinion. Moreover, with further optimization, integration of larger and more diverse datasets, and advanced preprocessing techniques, the model's performance and robustness can be enhanced, highlighting its potential for scalable and real-world healthcare applications.

10. References:

[1] İpek & Göktürk (2024) – ANN outperformed fuzzy inference systems for lung cancer prediction.

[2] Yuvaraj & Maheswari (2024) – Ensemble deep learning on gene expression improved lung cancer type classification.

[3] Bhatia et al. (2024) – Transfer learning–based model enhanced early lung carcinoma risk screening.

[4] Landge & Jain (2025) – Compared ANN, CNN, and RNN models, showing deep learning improves prediction accuracy.

[5] Aure et al. (2025) – Reviewed ML techniques, confirming AI's effectiveness in early lung cancer detection.

[6] Yuan et al. (2025) – Meta-analysis showed deep learning greatly improves image-based diagnostic precision.

[7] Kukreja & Sabharwa (2024) – CNNs provide highly accurate lung cancer classification from imaging data.