A

Major Project

On

# ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By


C. Raghavi (187R1A0571)

P. Saivardhan Reddy (187R1A05B2)

G. Shravan kumar (187R1A0582)


Under the Guidance of

**Dr. M. MALYADRI**

(Associate Professor)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CMR TECHNICAL CAMPUS**

**UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.


2018-2022

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING


# CERTIFICATE

This is to certify that the project entitled **"ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK"** being submitted by **C. RAGHAVI (187R1A0571), P. SAI VARDHAN REDDY (187R1A05B2) & G. SHRAVAN KUMAR (187R1A0582)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him/her under our guidance and supervision during the year 2021-22.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**Dr. M. Malyadri**
**Associate Professor**
**INTERNAL GUIDE**

**Dr. A. Raji Reddy**
**DIRECTOR**

**Dr. K. Srujan Raju**
**HoD**

**EXTERNAL EXAMINER**

**Submitted for viva voice Examination held on** _____

# ACKNOWLEGDEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completionof this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Mr. M. Malyadri,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. Theblessing, help and guidance given by his shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, Mrs. G. Latha, Mr. A. Kiran Kumar,** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Scienceand Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch.Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

**C. Raghavi (187R1A0571)**

**P. Saivardhan Reddy (187R1A05B2)**

**G. Shravan Kumar (187R1A0582)**

# ABSTRACT

The project elaborating Neural Network, its various characteristics and business applications. A Neural Network is a machine which is designed to work like brain. It has the ability to perform complex calculations with ease. Cryptography is the exchange of information among the users without leakage of information to others. Many public key cryptography are available which are based on number theory but it has the drawback of requirement of large computational power, complexity and time consumption during generation of key. To overcome these drawbacks, we analyzed neural network is the best way to generate secret key. In this paper we proposed a very new approach in the field of cryptography. We are using neural networks in the field of cryptography. In our project, we have learned different neural network architectures as well as training algorithms. we use auto associative neural network concept of soft computing in combination with encryption technique to send data securely on communication network. The basic idea of cryptography is concealing of the data from unauthenticated users as they can misuse the data.

# LIST OF FIGURES/TABLES

# LIST OF SCREENSHOTS

# TABLE OF CONTENTS

# 1. INTRODUCTION

# 1.INTRODUCTION

## 1.1 PROJECT SCOPE

This project aims at removing the need for the encoding to follow a general ruleby using a neural network for decoding the cipher text. Hence introducing the randomness in coding making it so much more difficult to decode. We have also introduced the concept of including lies in the information transmitted to misguide any eavesdropper who manages to decipher the cipher text.

## 1.2 PROJECT PURPOSE

Security is one of the most important needs in network communication. Cryptography is a science which involves two techniques encryption and decryption and it basically enables to send sensitive and confidential data over the unsecure network. The basic idea of cryptography is concealing of the data from unauthenticatedusers as they can misuse the data.

## 1.3 OBJECTIVE OF PROJECT

The cryptography deals with building such systems of security of news that secure any from reading of trespasser. Systems of data privacy are called the cipher systems. The file of rules are made for encryption of every news is called the cipher key. Encryption is a process, in which we transform the open text, e.g. message to cipher text according to rules. Cryptanalysis of the news is the inverse process, in which the receiver of the cipher transforms it to the original text. The cipher key must have several heavy attributes. The best one is the singularity of encryption and cryptanalysis.

# 2. SYSTEM ANALYSIS

# 2.SYSTEM ANALYSIS

## 2.1 PROBLEM DEFINITION

Security is one of the most important needs in network communication. Cryptography is a science which involves two techniques encryption and decryption and it basically enables to send sensitive and confidential data over the unsecure network.

The basic idea of cryptography is concealing of the data from unauthenticated users asthey can misuse the data.

## 2.2 EXISTING SYSTEM

In existing system,Neural cryptography (Kanter and Kinzel 2002, Kinzel 2002) isbased on the effect that two neural networks are able to synchronize by mutual learning(Ruttor et al. 2006). In each step of this online procedure they receive a common input pattern and calculate their output. Then, both neural networks use those outputs present by their partner to adjust their own weights. This process leads to fully synchronized weight vectors.

### 2.2.1 DISADVANTAGES OF EXISTING SYSTEM

But, a third network which is only trained by the other two clearly has a disadvantage, because it cannot skip some repulsive steps.Therefore, bidirectional synchronization is much faster than unidirectional learning.

## 2.3 PROPOSED SYSTEM

In this project author is using neural network to encrypt and decrypt and this neuralnetwork will be trained with keys and plain text. While training neural networkapplication calculate weight between keys and neural network and this weight will be consider as encrypted data.This encrypted data can be send to any receiver and then receiver will perform below steps to decrypt text .

### 2.3.1 ADVANTAGES OF PROPOSED SYSTEM

It involves both encryption and decryption of data. And it enables to send the data securely over the insecure network.Encryption is applying key on plain text to convert it into cipher text and decryption is the reverse process of encryption.

## 2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposalis put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirementsfor the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

### 2.4.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thusthe developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### 2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

### 2.4.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

## 2.5 HARDWARE & SOFTWARE REQUIREMENTS

### 2.5.1 SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regards to what the areas of strength and deficit are and how to tackle them.

- **Python idel 3.7 version (or)**
- **Anaconda 3.7 ( or)**
- **Jupiter (or)**
- **Google colab**

### 2.5.2 HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

| | | |
|---|---|---|
| **3** | **Operating system** | **: windows, linux** |
| **4** | **Processor** | **: minimum intel i3** |
| **5** | **Ram** | **: minimum 4 gb** |
| **6** | **Hard disk** | **: minimum 250gb** |

# 3. ARCHITECHTURE

# 3.ARCHITECTURE

## 3.1 PROJECT ARCHITECTURE

Architecture diagram is a visual presentation of all of the aspects that constitute a system, either in part or whole. It is a depiction of a set of concepts that comprise architecture, such as its principles, components, and materials. It is also a system diagram used to abstract the general layout of the software system as well as the interactions, limitations, and limits between parts.

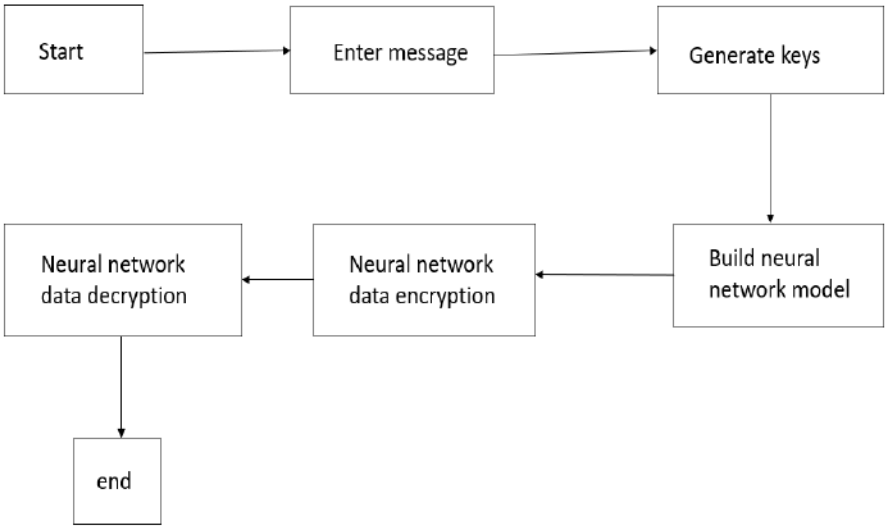



**Figure 3.1 ARCHITECTURE DIAGRAM FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

## 3.2 DATA FLOW DIAGRAM

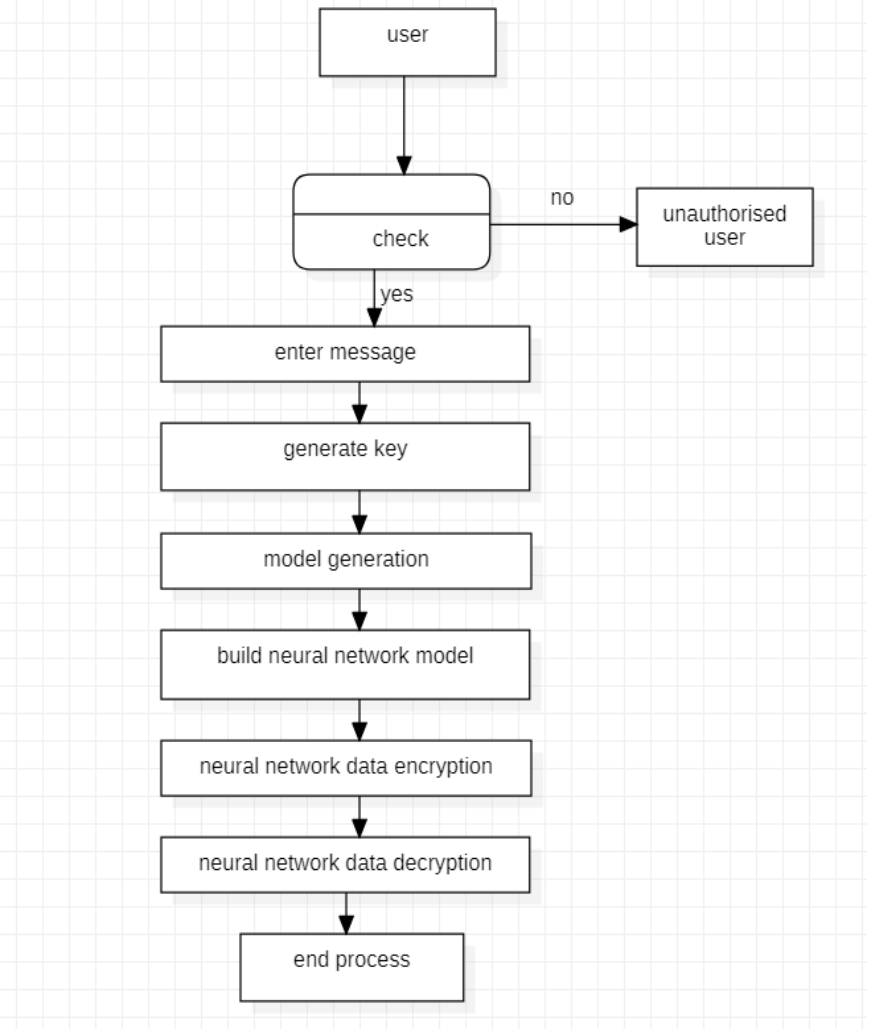



**Figure 3.2 DATAFLOW DIAGRAM FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, variousprocessing carried out on this data, and the output data is generated by this system.

1. The data flow diagram (DFD) is one of the most important modeling tools. It isused to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

2. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
3. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

## 3.3 USE CASE DIAGRAM

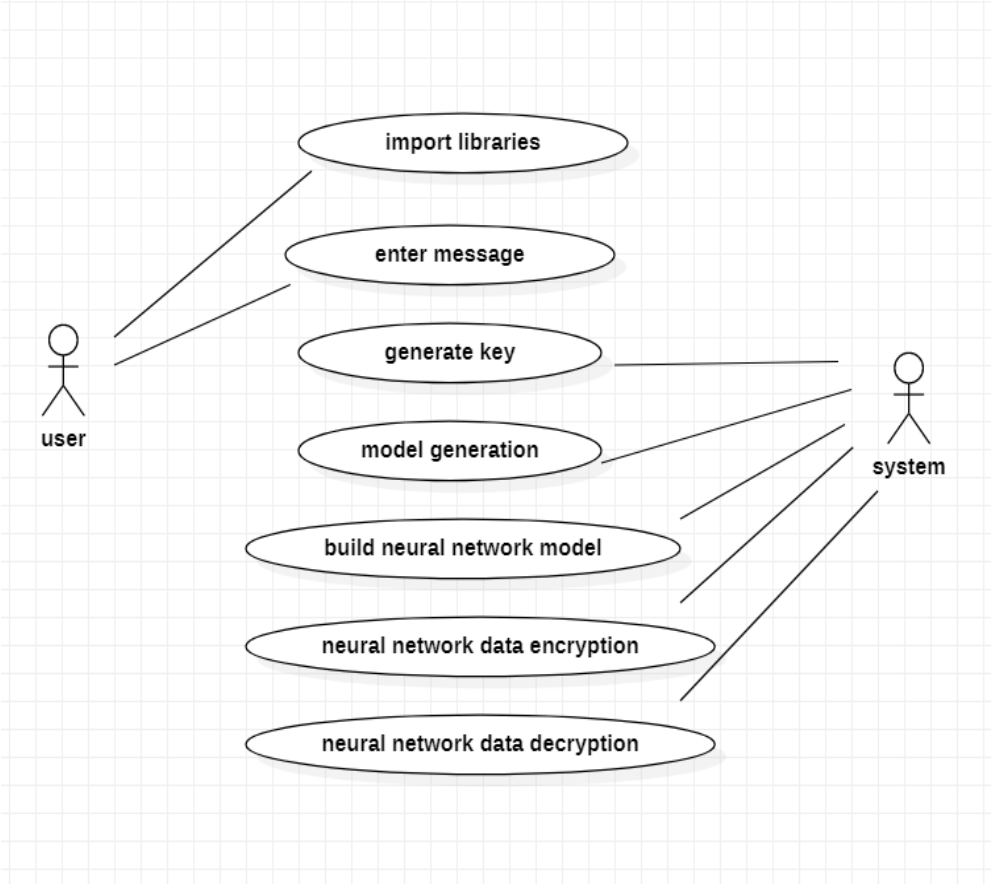



**Figure 3.3 USECASE DIAGRAM FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.

Its purpose is to present a graphical overview of the functionality provided by a system in terms of factors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

## 3.4 SEQUENCE DIAGRAM

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages".

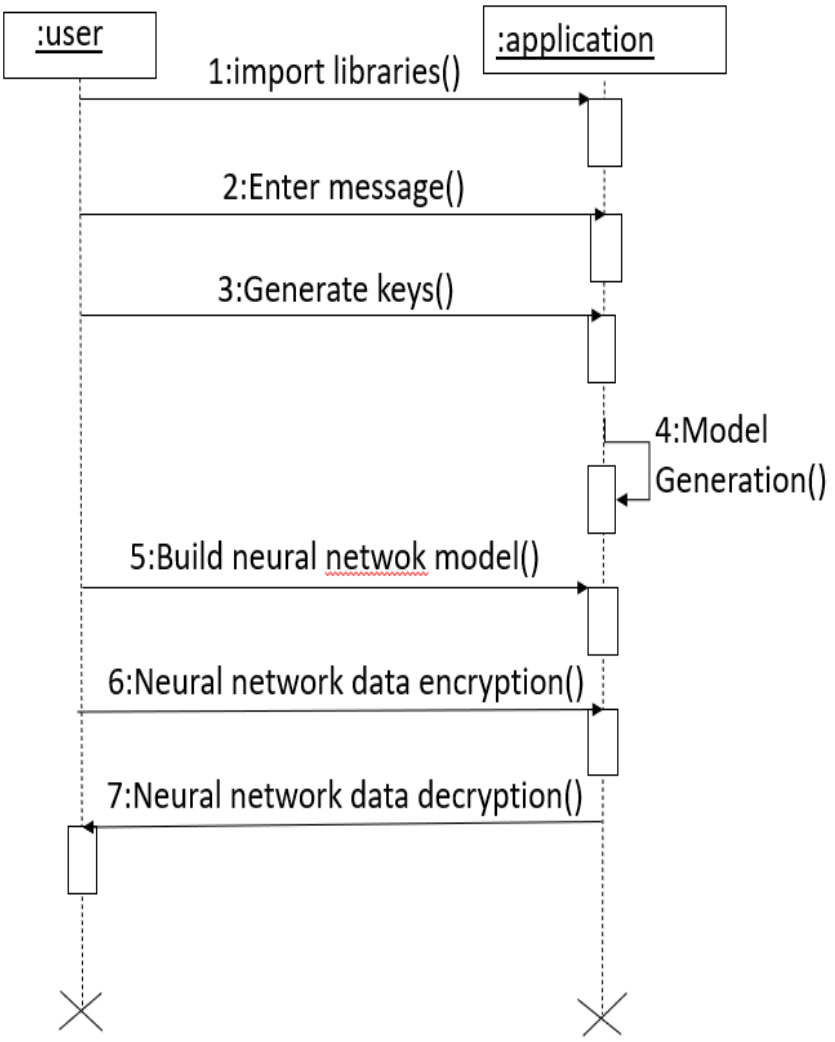



**Figure 3.4 SEQUENCE DIAGRAM FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

## 3.5 CLASS DIAGRAM

The class diagram is used to refine the use case diagram and define a detailed design of the system. The relationship or association between the classes can be either an "is-a" or "has-a" relationship. Each class in the class diagram may be capable of providing certain functionalities. These functionalities provided by the class are termed "methods" of the class.

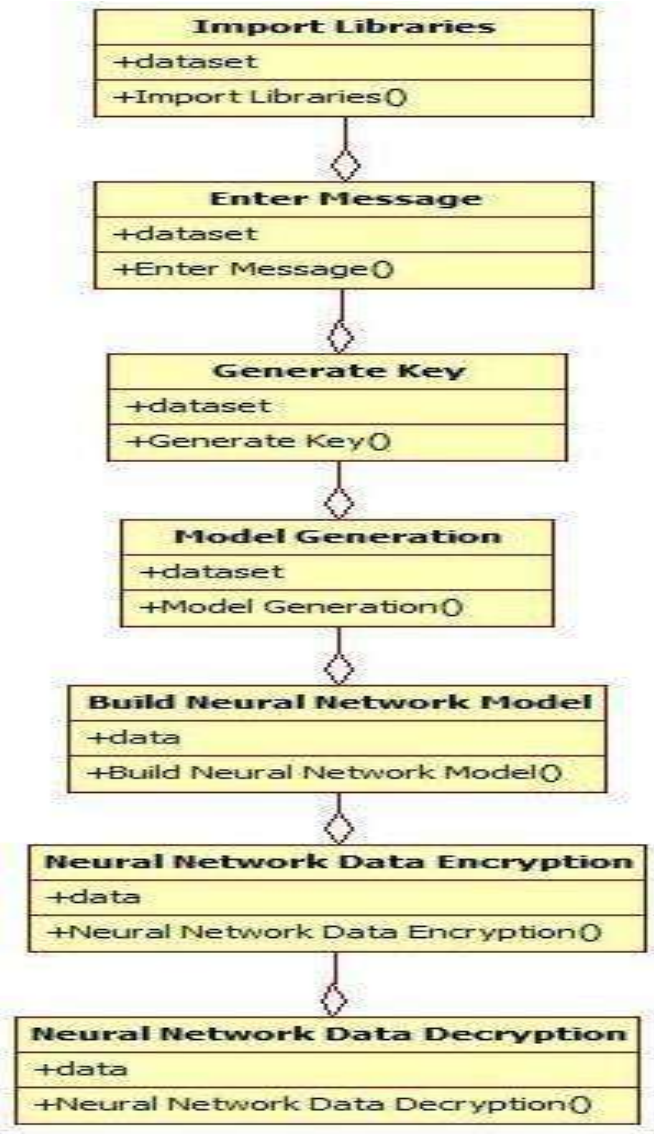



**Figure 3.5 CLASS DIAGRAM FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

## 3.6 MODULES:

To implement this project we have designed following modules

1) Generate key: using this module we will generate keys with random numbers.
2) Build neural network: using this module we will build neural network by calculating weight between keys and characters.
3) Encryption: using this module we will take input message from user and then apply neural network to recalculate weight between keys and user message. Calculate weightwill give encrypted matrix.
4) Decryption: using this module we will take encrypted weight value as input and then apply activation function to convert weight values to binary index and this index will be mapped to plain text to recover original message.

# 4. IMPLEMENTATION

# 4 .IMPLEMENTATION

## 4.1 SAMPLE CODE

```
from tkinter import messagebox from tkinter import *
from tkinter import simpledialog
import tkinter
from tkinter import filedialog
import matplotlib.pyplot as plt
from tkinter.filedialog import askopenfilename
import numpy as np

import os
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout
from keras.layers import LSTM
from keras.utils import np_utils
from keras.layers import Bidirectional
from keras.models import model_from_json

import pickle

main = tkinter.Tk()
main.title("Encryption And Decryption Algorithm Based On Neural Network")
main.geometry("1300x1200")


global filename
global classifier
global char_to_int
global int_to_char
vocab_list = []
dataX = []
dataY = []
global n_vocab
global encrypt

def getID(chars,data):
    index = 0
    for i in range(len(chars)):
        if chars[i] == data:
            index = i;
            break
    return index

def generateKey():
    global n_vocab
    dataX.clear()
```

```
    dataY.clear()
    global char_to_int
    global int_to_char
    global filename
    text.delete('1.0', END)

    sentences = ''
    with open('model/input.txt', "r") as file:
        for line in file:
            line = line.strip('\n')
            line = line.strip()
            line.lower()
            sentences+=line

    file.close()
    sentences = sentences.strip()
    vocab_list.clear()
    for i in range(len(sentences)):
        vocab_list.append(sentences[i])
    raw_text = sentences
    chars = sorted(list(set(raw_text)))
    char_to_int = dict((c, i) for i, c in enumerate(chars))
    int_to_char = dict((i, c) for i, c in enumerate(chars))
    n_chars = len(raw_text)
    n_vocab = len(chars)
    text.insert(END,"Key Generation Task Completed\n")
    for i in range(0, n_chars):
        dataX.append(char_to_int.get(raw_text[i]))
        dataY.append(getID(chars,raw_text[i]))
    text.insert(END,"Generated Key :
"+str(char_to_int['w'])+str(char_to_int['p'])+str(char_to_int['l'])+str(char_to_int['e'])+str(char_to_int['
A'])+str(char_to_int['b'])+"\n")

def buildModel():
    global classifier
    text.delete('1.0', END)
    n_patterns = len(dataX)
    if os.path.exists('model/nn_model.json'):
        with open('model/nn_model.json', "r") as json_file:
            loaded_model_json = json_file.read()
            classifier = model_from_json(loaded_model_json)
        json_file.close()
        classifier.load_weights("model/nn_model_weights.h5")
        classifier._make_predict_function()

    else:

        seq_length = 1
        X = np.reshape(dataX, (n_patterns, seq_length, 1))
```

```
        X = X / float(n_vocab)
        y = np_utils.to_categorical(dataY)
        print(X.shape)
        print(y.shape)

        model = Sequential()
        model.add(Bidirectional(LSTM(256, input_shape=(X.shape[1], X.shape[2]),
return  sequences=True)))
        model.add(Dropout(0.2))
        model.add(Bidirectional(LSTM(256)))
        model.add(Dropout(0.2))
        model.add(Dense(y.shape[1], activation='softmax'))
        model.compile(loss='categorical_crossentropy', optimizer='adam')

        hist = model.fit(X, y, epochs=8000, batch_size=64)
        model.save_weights('model/nn_model_weights.h5')
        model  json = model.to  json()
        with open("model/nn_model.json", "w") as json_file:
            json_file.write(model_json)
        json_file.close()
        f = open('model/nn_history.pckl', 'wb')
        pickle.dump(hist.history, f)
        f.close()
    f = open('model/nn_history.pckl', 'rb')
    data = pickle.load(f)
    f.close()
    loss = data['loss']
    lossValue = loss[7999]
    loss = loss[0:100]
    text.insert(END,"Neural Network Training Model Loss = "+str(lossValue)+"\n")
    plt.figure(figsize=(10,6))
    plt.grid(True)
    plt.xlabel('Epoch/Iterations')
    plt.ylabel('Loss')
    plt.plot(loss, 'ro-', color = 'blue')
    plt.legend(['Neural Network Loss'], loc='upper left')
    plt.title('Neural Network Loss Graph')
    plt.show()


def  decimalToBinary(n):
    return "{0:b}".format(int(n))

def encryption():
    text.delete('1.0', END)
    global encrypt
    global classifier
    encrypt = []
    message = tf1.get();
```

```
    binValue = ''
    for i in  range(len(message)):
        data = char_to_int[message[i]]
        temp = []
        temp.append(data)
        temp = np.asarray(temp)
        x = np.reshape(temp, (1, temp.shape[0], 1))
        x = x / float(n_vocab)
        encrypted = classifier.predict(x, verbose=0)[0]
        encrypt.append(np.argmax(encrypted))
        binValue+=str(decimalToBinary(np.argmax(encrypted)))+" "
    text.insert(END,"Original Message : "+message+'\n\n')
    text.insert(END,"Encrypted Message Matrix : "+str(encrypt)+"\n\n")
    text.insert(END,"Encrypted Binary Value  : "+str(binValue.strip())+"\n\n")

def decryption():
    text.delete('1.0', END)
    global encrypt
    global classifier
    encrypt = np.asarray(encrypt)

    output = ''
    for i in range(len(encrypt)):
        index = encrypt[i]
        result = int_to_char[index]
        output+=result
    text.insert(END,"Decrypted Message : "+str(output)+"\n\n")

def close():
    main.destroy()


font = ('times', 16, 'bold')
title = Label(main, text='Encryption And Decryption Algorithm Based On Neural Network')
title.config(bg='firebrick4', fg='dodger blue')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 12, 'bold')
text=Text(main,height=17,width=150)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=50,y=170)
text.config(font=font1)


font1 = ('times', 13, 'bold')
uploadButton = Button(main, text="Generate Key", command=generateKey, bg='#ffb3fe')
```

```
uploadButton.place(x=50,y=550)
uploadButton.config(font=font1)

lstmButton1 = Button(main, text="Build Neural Network Model", command=buildModel,
bg='#ffb3fe')
lstmButton1.place(x=350,y=550)
lstmButton1.config(font=font1)

l1 = Label(main, text='Enter Message')
l1.config(font=font1)
l1.place(x=50,y=100)

tf1 = Entry(main,width=40)
tf1.config(font=font1)
tf1.place(x=230,y=100)

gruButton = Button(main, text="Neural Network Data Encryption", command=encryption,
bg='#ffb3fe')
gruButton.place(x=50,y=600)
gruButton.config(font=font1)

graphButton = Button(main, text="Neural Network Data Decryption", command=decryption,
bg='#ffb3fe')
graphButton.place(x=350,y=600)
graphButton.config(font=font1)

predictButton = Button(main, text="Exit", command=close, bg='#ffb3fe')

predictButton.place(x=630,y=600)
predictButton.config(font=font1)

main.config(bg='LightSalmon3')main.mainloop
```

## IMPORTED LIBRARIES

### Tensorflow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

### Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

### Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data

structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

**Matplotlib**

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

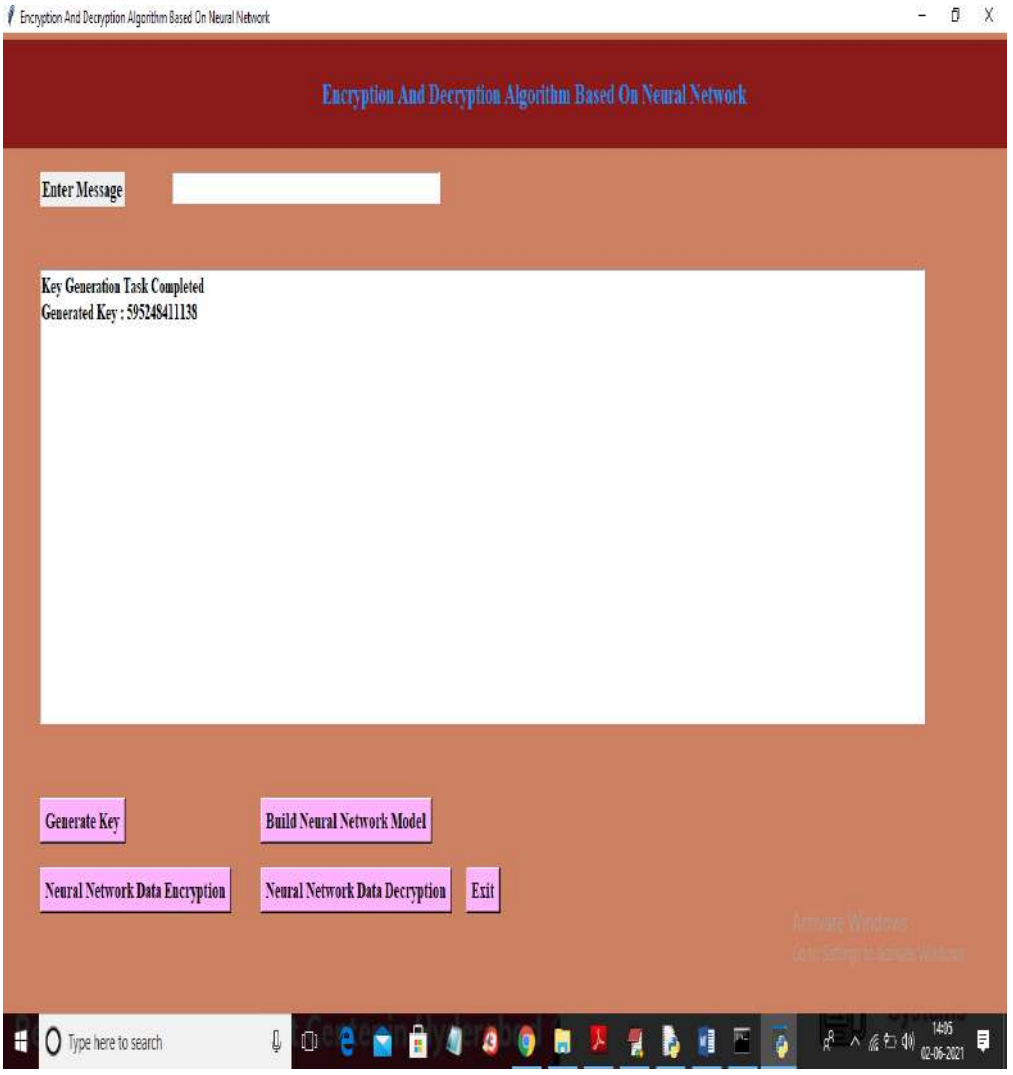
For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

**Scikit – learn**

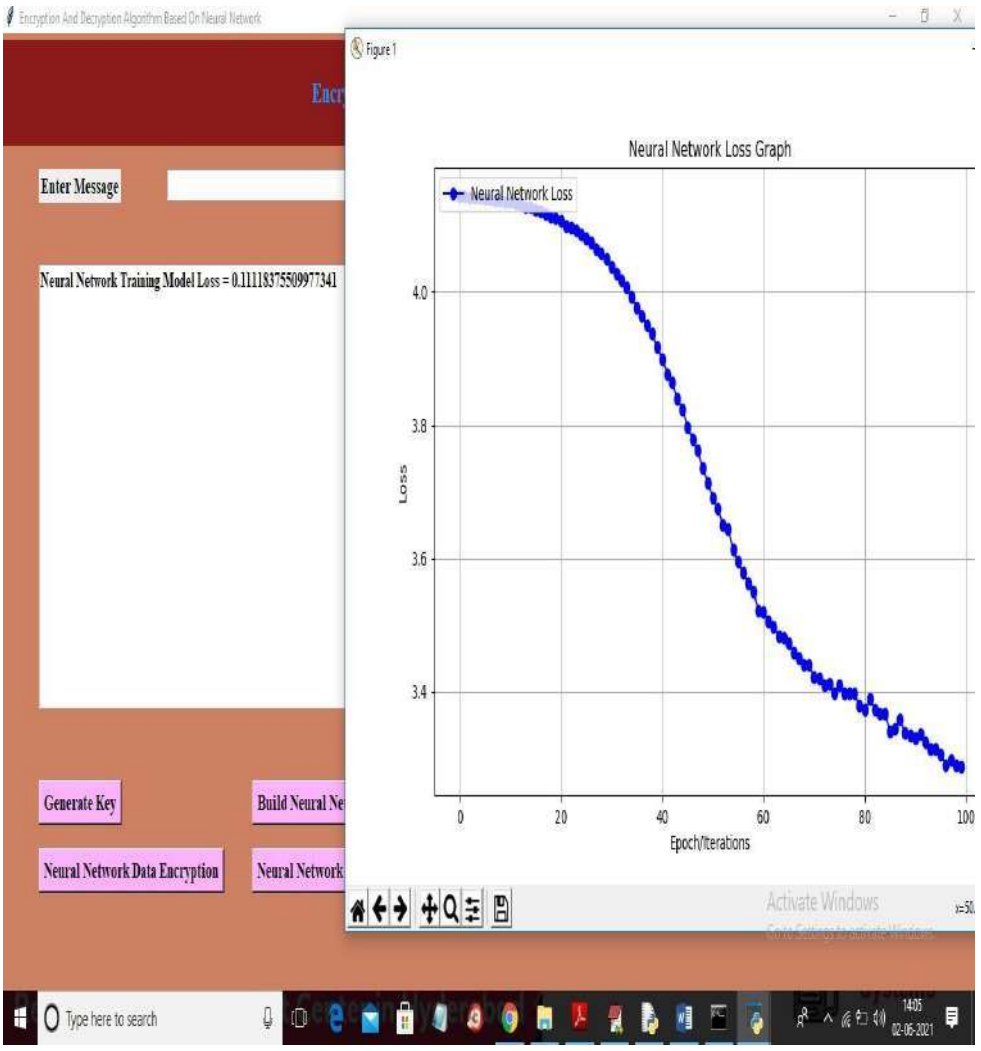
Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use.

# 5. SCREENSHOTS

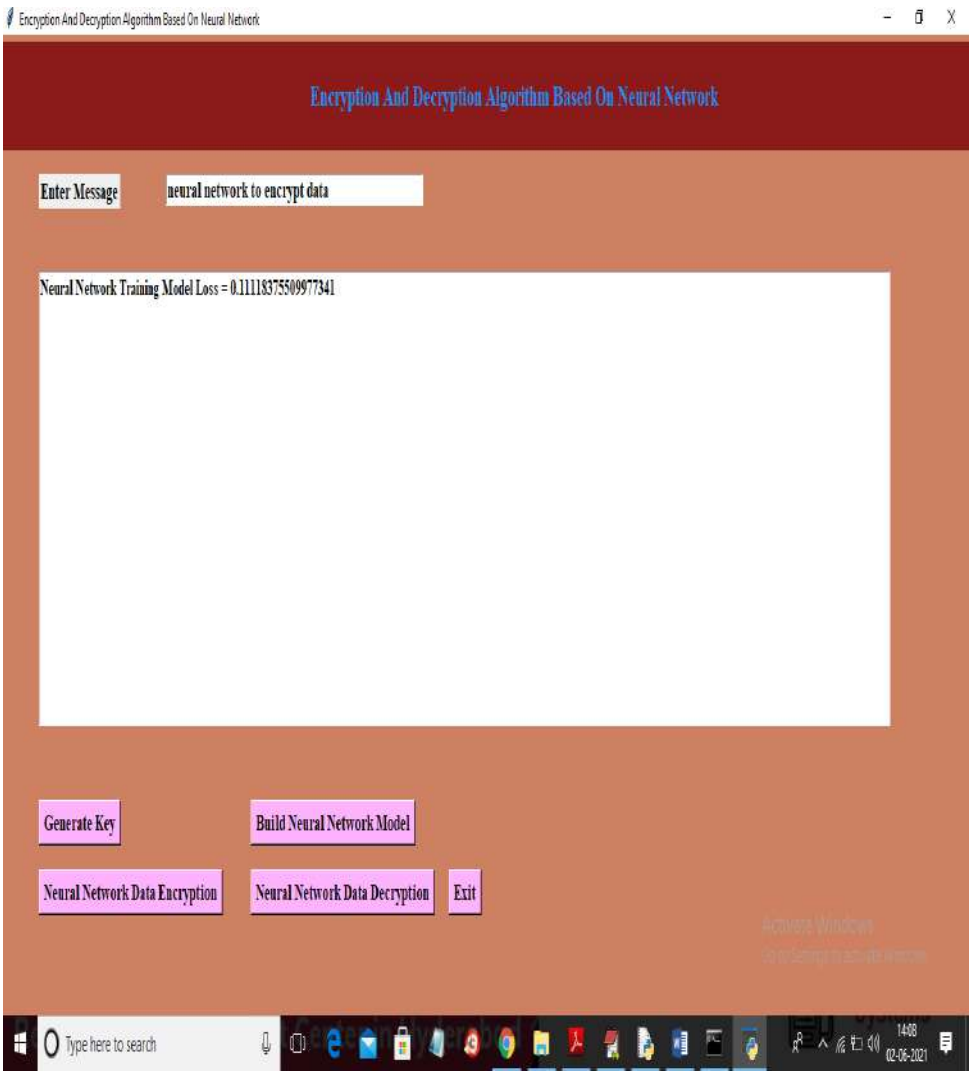

**SCREENSHOT 5.1 WEB PAGE FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**



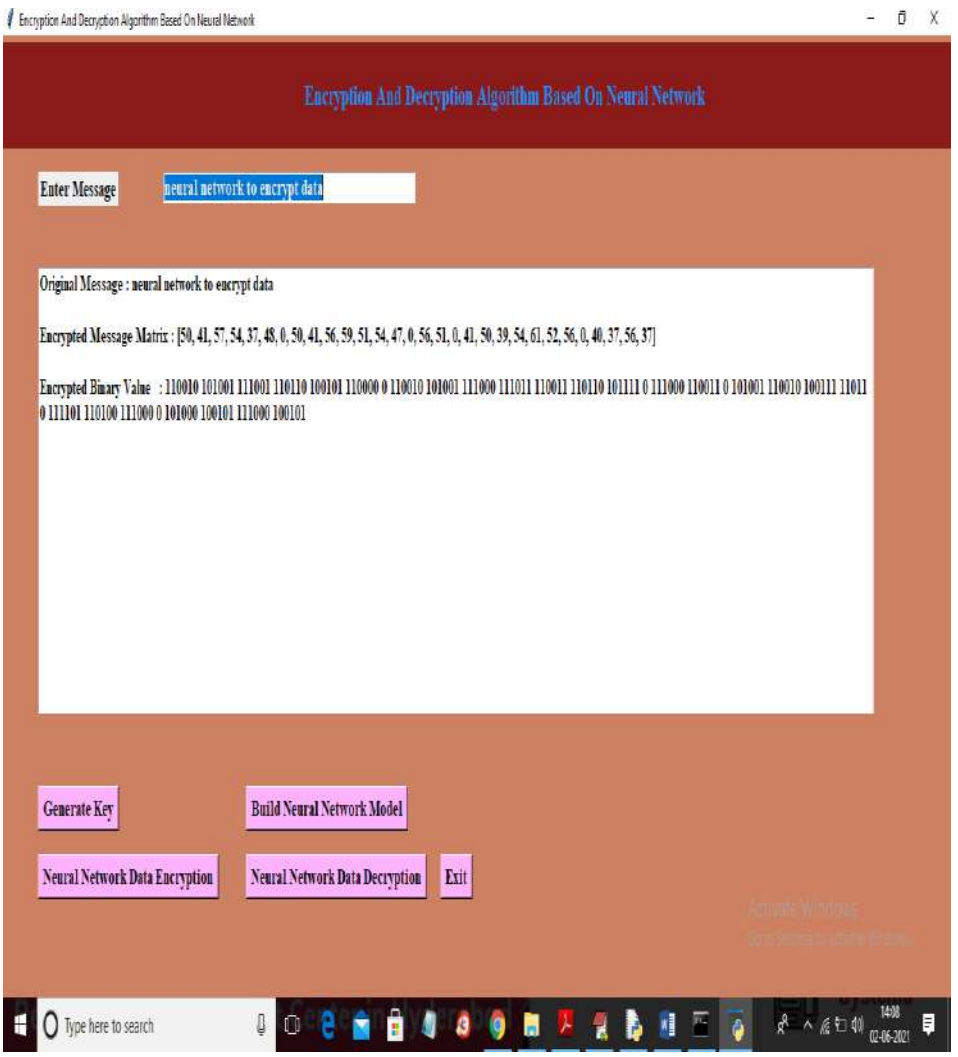

**SCREENSHOT 5.2 GENERATION OF KEYS FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**



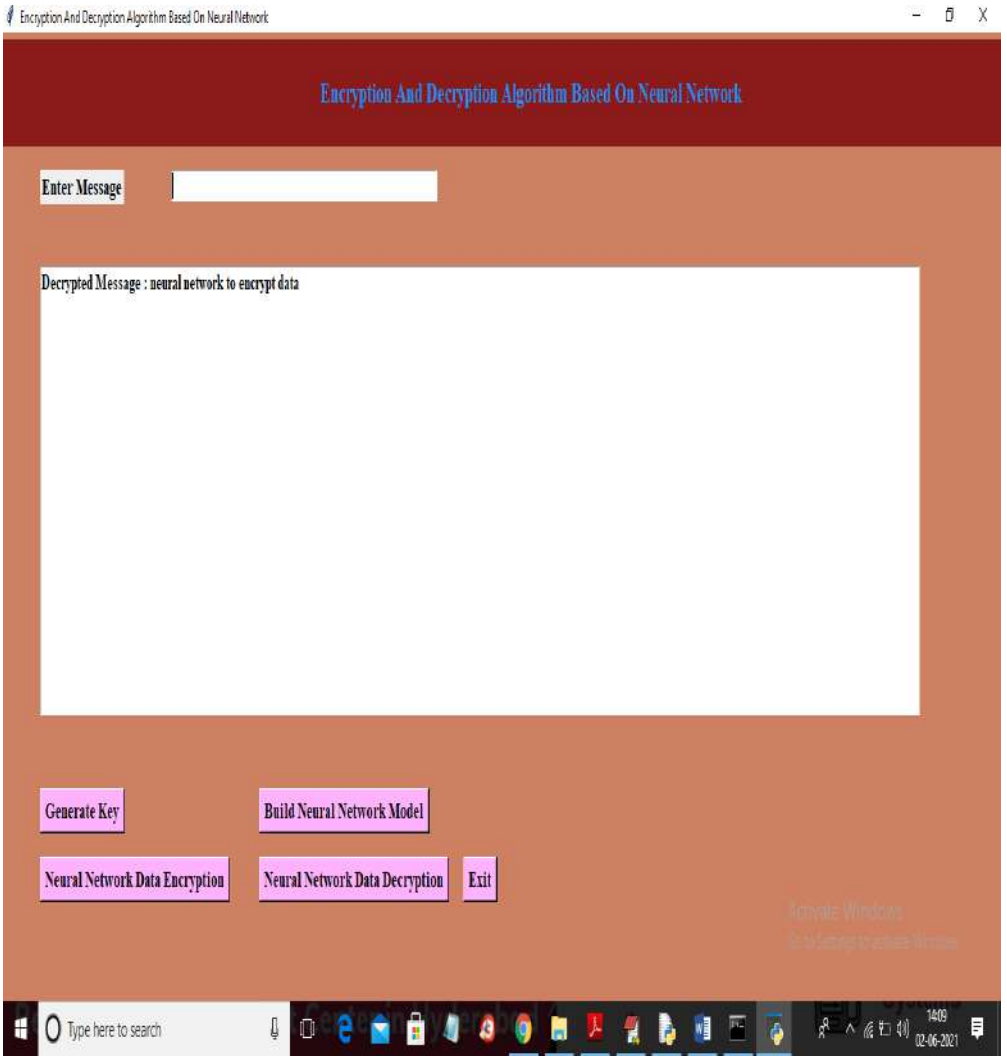

**SCREENSHOT 5.3 NEURAL NETWORK MODEL FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**




**SCREENSHOT 5.4 WEB PAGE WITH ENTERED MESSAGE FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**




**SCREENSHOT 5.5 NEURAL NETWORK DATA ENCRYPTION**
**FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**




**SCREENSHOT 5.6 NEURAL NETWORK DATA DECRYPTION**
**FOR ENCRYPTION AND DECRYPTION ALGORITHM BASED ON NEURAL NETWORK**

# 6. TESTING

# 6.TESTING

## 6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

## 6.2 TYPES OF TESTING

### 6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

### 6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determineif they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent.

### 6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.
Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.Functions :identified functions must be exercised.
Output : identified classes of application outputs must be exercised.Systems/Procedures : interfacingsystems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

## 6.3 TESTCASES

**USER REQUIREMENTS:**

**1.Home**

| | | |
|---|---|---|
| Use case ID | | Encryption and Decryption using Neural Network |
| Use case Name | | Home button |
| | Description | Display home page of application |
| | Primary actor | User |
| | Precondition | User must open application |
| | Post condition | Display the Home Page of an application |
| | Frequency of Use case | Many times |
| | Alternative use case | N/A |
| | Attachments | N/A |

# 7. CONCLUSION

# 7.CONCLUSION

## 7.1 PROJECT CONCLUSION

The concept of using neural networks in the field of cryptography is growing at a rapid pace. Various neuro-crypto algorithms proposed by researchers are available in literature. But most of them are limited to the key generation and cryptanalysis. In the research work auto associative memory network is utilized to encrypt the plain text into the form which is totally independent from the previous one. The algorithm is pretty simple to implement and has faster encryption and decryption speed. The algorithm is following the symmetric key system which makes it vulnerable to leakage of key. To overcome this, only trusted parties should be involved in communication ora trusted third party can be used as an authority to prevent the key leakage.

## 7.2 FUTURE SCOPE

This project aims at removing the need for the encoding to follow a general rule by using a neural network for decoding the cipher text. Hence introducing the randomness in coding making it so much more difficult to decode. We have also introduced the concept of including lies in the information transmitted to misguideany eavesdropper who manages to decipher the cipher text.

# 8. BIBILOGRAPHY

# 8.BIBILOGRAPHY

## 8.1 REFERENCES

[1] M. Hellman, "An overview of public key cryptography", IEEE CommunicationsMagazine, 2002,40(5): 42-49.

[2]Diffie W, Hellman M., "New Directions in Cryptography". IEEE Transactions on InformationTheory. 1976, 22(6):644-654.

[3]L. P. Yee and L. C. D. Silva. Application of multilayer per- ceptron networks in public keycryptography. Proceedings ofIJCNN02,2(Honolulu,HI,USA):1439–1443, May2002.

[4] Salomaa, Arto. Public-key cryptography. Springer Science & Business Media,2013.

[5]Law, Laurie, et al. "An efficient protocol for authenticated key agreement."Designs, Codes andCryptography 28.2 (2003): 119-134.

## 8.2 WEBSITES

[1] Code snippets for any errors http://stackoverflow.com/

[2] Android Development Guide https://www.udemy.com/android

[3] Xml and Layout Guide https://www.androidhive.com/

[4] Connecting to Firebase Docs https://firebase.google.com

[5] Software Testing http://en.wikipedia.org/wiki/Software_testing

[6] Manual Testing http://en.wikipedia.org/wiki/Manual_testing

[7] Performance Testing
http://en.wikipedia.org/wiki/Software_performance_testing

## 8.3 GITHUB LINK :

**https://github.com/CRAGHAVI/ENCRYPTION-AND-DECRYPTION-ALGORITHM-BASED-ON-NEURAL-NETWORKS.git**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 6 June 2022, pp: 795-798 www.ijaem.net ISSN: 2395-5252

# Encryption and Decryption Algorithm Based on Neural Network

Dr.M.Malyadri[1],C.Raghavi[2],P.Saivardhan Reddy[3], G.Shravan Kumar[4]

[1]*Associate Professor,Dept of CSE,CMR Technical campus,Kandlakoya,Medchal-501401.*
[2]*B.Tech Student,Dept of CSE,CMR Technical campus,Kandlakoya,Medchal-501401.*
[3]*B.Tech Student,Dept of CSE,CMR Technical campus,Kandlakoya,Medchal-501401.*
[4]*B.Tech Student,Dept of CSE,CMR Technical campus,Kandlakoya,Medchal-501401.*

Submitted: 01-06-2022 Revised: 10-06-2022 Accepted: 15-06-2022

**ABSTRACT:**The project elaborating Neural Network, its various characteristics and business applications. A Neural Network is a machine which is designed to work like brain. It has the capability to execute complex calculations easily. Cryptography is the interchange of data between the users without seep of data to others. Many public key cryptography are there which are based on numerical theory but it has the limitation of availability of large analytical power, trignometric and time utilization during creation of key. To overcome these limitations, we prepared neural network is the perfect way to create secret key. In this, we implemented a perfect approach in the study of cryptography. We are utilizing neural networks in the study of cryptography. In our article, we had knowledge of many other neural network architectures along with training algorithms. we use self associative neural network concept of soft computing in coordination with encryption technique to send information securely on communication network. The ground idea of cryptography is concealing of the information from unauthenticated users as they can misuse the data.

**KEYWORDS:**Keysgeneration,NeuralNetworkModel,Encryption,Decryption.

## I. INTRODUCTION

This paper aims at removing the necessity for the coding to follow a general rule by employing a neural network for cryptography the cipher text. Thus introducing the randomness in secret writing creating it most harder to decrypt. We've conjointly introduced the thought of as well as lies with in the data transmitted to misguide any listener who manages to decipher the ciphetext.

Security is one among the foremost vital desires in network communication. Cryptography may be a science that involves two techniques secret writing and decipherment and it essentially permits to send sensitive and confidential information over the unsecure network.

## II. LITERATURE REVIEW

Key-insulated symmetric key cryptography and mitigating attacks against cryptographic cloud software. **[1] Dodis, Yevgeniy, et al** Software-based attacks (e.g., malware) pose a big threat to cryptographic software because they can compromise the as-sociated cryptographic keys in their entirety. In this paper, we have a tendency to investigate key-insulated radially symmetrical key cryptography, which may mitigate the harm caused by perennial attacks against scientific discipline package.For example,the feasibleness of key-insulated radially symmetrical key cryptography, we have a tendency to additionally report a proof-of-concept implementation within the Kernel-based Vir-tual Machine (KVM) atmosphere.

An efficient protocol for authenticated key agreement. **[2] Law, Laurie, et al**, Authentication and key establishment are fundamental building blocks for securing electronic communication. Cryptographic rule for coding and integrity cannot perform their perform unless secure keys are established and therefore the users grasp that parties share such keys. It's essential that protocols for providing and key institution area unit fit their purpose. This paper proposes a replacement and economical key institution protocol with in the uneven (public key) setting that's supported MTI (Matsumoto, Takashima and Imai)-two pass key agreement protocol which consists of three phases; The Transfer and Verification Phase, and The Key

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 6 June 2022, pp: 795-798 www.ijaem.net ISSN: 2395-5252

Generation Phase. This potential attacks(Known-Key Security, Forward (Perfect)Secrecy,Key-Compromise Impersonation, Unknown Key-Share Attack, Small Subgroup Attack, and Man-in-the-Middle Attack) with low complexity (complexity is 4), also it provide authentication between the two entities before exchanging the session keys. On the impossibility of private key cryptography with weakly random keys. **[3] McInnes, James L., and Benny Pinkas**, The properties of weak sources of randomness have been investigated in many contexts and using several models of weakly random behaviour. For two such models, developed by Santha and Vazirani, and Chor and Goldreich, it is known that the output from one such source cannot be "compressed" to produce nearly random bits. At the sametime, however, a single source is sufficient to solve problems in the randomized complexity classes BPP and RP. It is natural to raise precisely that tasks is employing a single, weak supply of randomness and that cannot. This work begins to answer this question by establishing that one frail random supply of either model can not be accoustomed acquire a secure "one-time-pad" kind of cryptosystem. New Steganographic Technique using Neural Network. **[4] Phadke, Akshay, and Aditi Mayekar**, Steganographic technique is used to hide the information,a string of characters information, in a carrier image. The information is coded into individual rows of the constituent primaries of the carrier. Victimisation this system the neural network is utilized to find the presence of the message within the individual rows of the carrier image and to retrieve the contents of the message hidden in the carrier. This technique is able to maintain good visual quality of the carrier image. The results of this technique revealed high PSNR values and significantly less MSE values for the unmodified carrier and the steganographic image.

## III. PROPOSED SYSTEM

In this paper author is using neural network to encrypt and decrypt and this neural network will be trained with keys and plain text. While training neural network application calculate weight between keys and neural network and this weight will be consider as encrypted data.This encrypted data can be send to any receiver and then receiver willperform below steps to decrypt text .

### 1.1 PROJECT ARCHITECTURE

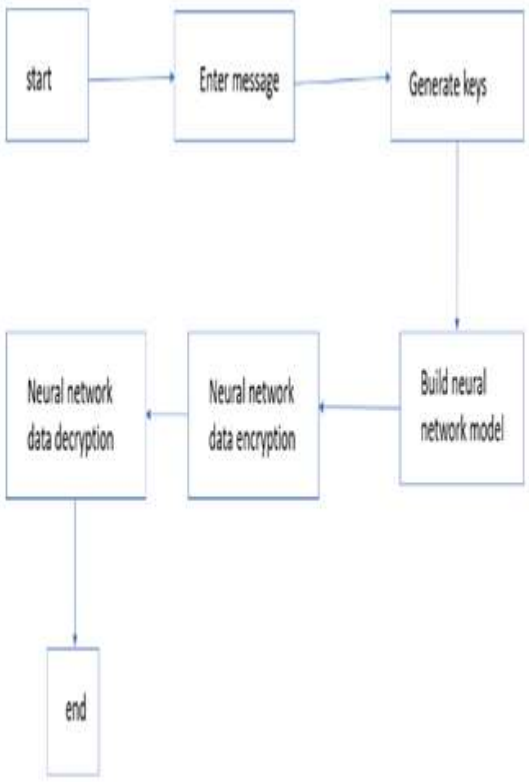



**Fig 1.** Architecture diagram of Encryption and decryption algorithm based on neural network

### 1.2 DATA FLOW DIAGRAM

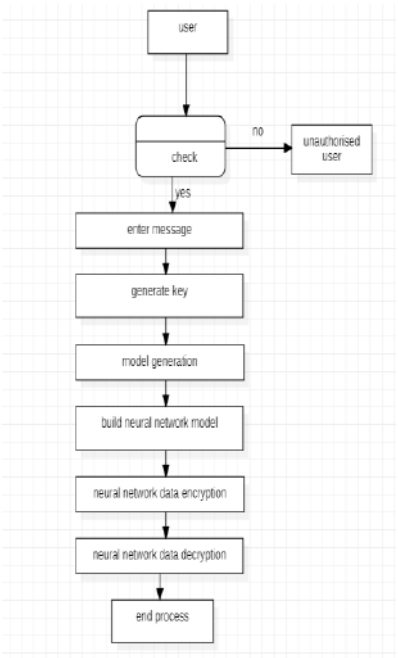



**Fig 2**. Dataflow diagram of Encryption and Decryption alogorithm based on neural network

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 6 June 2022, pp: 795-798 www.ijaem.net ISSN: 2395-5252

## IV. RESULTS AND DISCUSSION

In below screen click on 'Generate Key' button to generate keys.

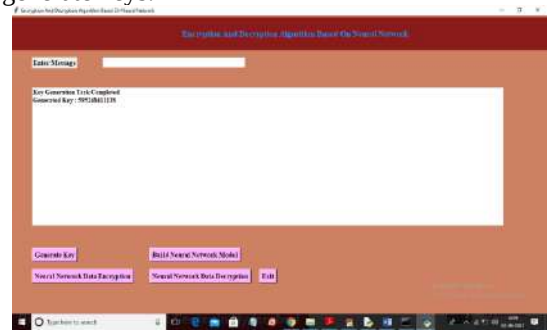


**Fig 3**. Generation of keys for Encryption and Decryption alogorithm based on neural networks

In above screen random key is generated and now click on 'Build Neural Network Model' button to generate neural network for encryption and decryption.

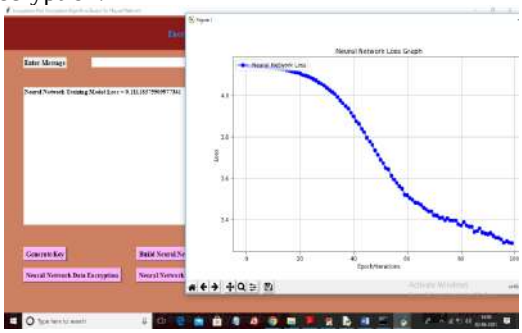


**Fig 4.**Neural Network Model for Encryption and Decryption alogorithm based on neural networks

In above screen neural network model is built using keys and plain test and we can see model loss reduce from 5.0 to 0.11 and in graph we can see x-axis represents EPOCH and y-axis represents loss value and we can see in above graph at each increasing epoch loss value is getting decreased and we can see loss value decrease from 5 to 0.1 and in any neural network can be consider as reliable if its loss value decrease to 0. Now model is build and enter some message in text field.

**Fig 5.**.Web page showing Model with zero loss for Encryption and Decryption alogorithm based on neural networks

In above screen in text field I entered message as 'neural network to encrypt data' and now click on 'Neural Network Data Encryption' button to encrypt message.

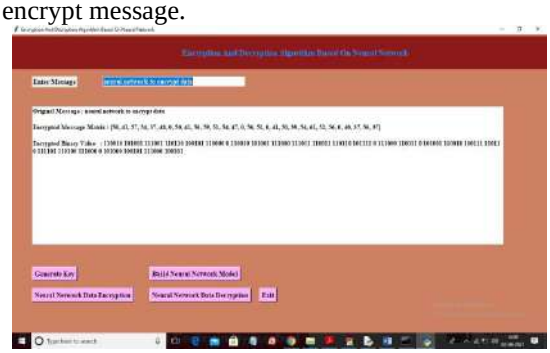


**Fig 6**. Encrypted matrix for Encryption and Decryption alogorithm based on neural networks

In above screen message is encrypted and we got encrypted matrix and binary numbers and now click on 'Neural Network Data Decryption' button to decrypt message.

**Fig 7**. Decrypted message for Encryption and Decryption alogorithm based on neural networks

In above screen in text area we can see message is decrypted successfully. Similarly you can enter any message and perform encryption and decryption.

Note: key generation and build neural network model button has to click only one time when application started and then u can perform encryption and decryption any number of times.

## V. CONCLUSION

The concept of using neural networks in the field of cryptography is growing at a rapid pace. Various neuro- crypto algorithms planned by researchers are offered in literature. But most of them are limited to the key generation and cryptanalysis. In the research work auto associative memory network is utilized to encrypt the plain text into the form which is totally independent from the previous one. The formula is pretty easy to implement and has quicker coding and decipherment speed. The algorithm is following the symmetric key system which makes it vulnerable to leakage of key. To overcome this, solely sure parties ought to be concerned in communication or a sure third party will be used as associate authority to forestall the key run.

## VI. ACKNOWLEDGEMENT

The success of this project includes help from our guide as well.We are grateful tour guide,Mr.M.Malyadri,Associate Professor,CMR Technical Campus,for his expertise that guided us in our research.

## REFERENCES

[1]. Dodis yevgeniy,et al. "Key-insulated symmetric key cryptography and mitigating attacks against cryptographic cloud software." Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security. ACM, 2012.

[2]. Law, lawrie,et al.”An efficient protocol for authenticated key agreement.”Designs, Codes and Cryptography 28.2 (2003): 119-134.

[3]. McInnes,James L..,and Benny pinkas.”On the impossibility of private key cryptography with weakly random keys”.Advances in cryptologyCRYPT0’90.Springer Berlin Heidelberg,1991.421-435.

[4]. Phadke,Akshay,and Aditi Mayekar.”New Steganographic Technique using Neural Network.”International Journal of Computer Applications 82.7 (2013) :39-42.

[5]. Nakano,Kaoru.”Association-a model of associative memor”.Systems,Man and cybernetics,IEEE Transactions on 3(1972):380-388.

[6]. Amari,S-I.”Neural theory of association and concept formation.”Biological cybernatics 26.3(1997):175-185.

[7]. Wang,Guofeng,and Yinhu Cui.”On line tool wear monitoring based on auto associative neural network.”Journal of Intelligent Manufacturing 24.6(2013):1085-1094.

[8]. Widrow,Bernard,Juan Carlos Aragon,and Brian Mitchell Percival.”Cognitive memory and auto-associative neural network based search engine for computer and network located images and photographs”.U.S.Patent No.7,991,714.2 Aug,2011.

[9]. Valentin,Dominique.Herve Abdi,and Alice J.O’TOOLE.”Categorization and identification of human face images by neural networks:A review of the linear auto associative and principal component approaches.”Journal of biological systems 2.03(1994):413-429.

[10]. M.Hellman,”An overview of public key cryptography”,IEEE CommunicationsMagazine,2002,40(5):42-49.

[11]. Diffie W,Hellman M.,”New Directions in Cryptography”.IEEE Transactions on Information Theory.1976,22(6):644-654.

[12]. L.P.Yee and L.C.D.Silva.Application of multilayer perceptron networks in public key cryptography.Proceedings of IJCNN02,2(Honolulu,HI,USA):1439-1443,May 2002.

# Certificate of Publication

This is to confirm that

C.Raghavi , P.Saivardhan Reddy , G.Shravan Kumar

Published following article

Encryption and Decryption Algorithm Based on Neural Network

Volume 4, Issue 6, pp: 795-798

**www.ijaem.net**

**A Peer Reviewed Journal**

**International journal of Advances in Engineering and Management (IJAEM)**

**ISSN: 2395-5252**

**Publication Head**