

NANDHA ENGINEERING COLLEGE

ERODE – 638052

(Autonomous)

(Affiliated to Anna University, Chennai)



**DEPARTMENT
OF**

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SPAM DETECTION USING MACHINE LEARNING

PBL REPORT

(17AIC02 – INTRODUCTION TO ARTIFICIAL INTELLIGENCE)

Submitted by

GOWTHAM RAJ .B (21AI012)

JAGADEESWARAN.VP (21AI016)

JASSIM USMAN.M (21AI017)

In partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

NANDHA ENGINEERING COLLEGE

(Autonomous)

(Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

**Certified that this Report titled “SPAM DETECTION USING
MACHINE LEARNING” is the bonafide work of GOWTHAM RAJ.B
(21AI012), JAGADEESWARAN.VP (21AI016), JASSIM USMAN.M
(21AI017), who carried out the work under my supervision.**

Signature of the HOD

**Mrs.M.PARVATHI M.E.,
M.E.,**

Associate Professor & Head

Department of AI&DS

Nandha Engineering College

Erode– 638052.

Signature of the Supervisor

Mrs.M.SENTHAMARAI

Assistant Professor

Department of AI&DS

Nandha Engineering College

Erode– 638052.

Submitted for End Semester PBL Review Examination held on _____

ACKNOWLEDGEMENT

I express my thanks to our beloved Chairman of Sri Nandha Educational Trust **Thiru. V. Shanmugan** and our beloved Secretaries, **Thiru S. Nandhakumar Pradeep** of Sri Nandha Educational Trust and **Thiru. S. Thirumoorthi** of Nandha Educational Institutions for providing me all the basic amenities to complete the course successfully.

I specially thank **Dr. S. Arumugam**, Chief Executive Officer of Nandha Educational Institutions for his affection and support in all aspects have made me to complete the course successfully.

I wish to convey my earnest gratefulness to our cherished Principal of Nandha Engineering College, **Dr. N. Rengarajan, ME., Ph.D.**, for his constant support in my successful completion of my project work.

I articulate my genuine and sincere thanks to our dear hearted Head of the Department of Artificial Intelligence and Data Science **Mrs. Parvathi.**, who has been the key spring of motivation to me throughout the completion of my course and my project work.

I wish to convey my hearty thanks to my beloved Project Supervisor **Mrs. SENTHAMARAI.M, M.E.**, Assistant Professor, Department of Artificial Intelligence and Data Science for his continuous monitoring for the project work.

I am very much gratified to all teaching and non-teaching staff of our department who were direct and indirect stroke throughout my progress. I would like to acknowledge my heartfelt thanks to my parents and my friends who have supported me with their unconditional love and encouragement. Finally, I would like to thank the Almighty for his blessings.

TABLE OF CONTENTS		
CHAPTER NO.	TITLE	PAGENO.
	ABSTRACT	5
1	INTRODUCTION	6
2	SYSTEM SPECIFICATION	7
3	SOFTWARE DESCRIPTION	8
4	PROJECT DESCRIPTION	10
5	FUTURE ENHANCEMENT	12
6	CONCLUSION	13
7	APPENDIX	14
	7.1 SOURCE CODE	14
	7.2 SCREEN SHOTS	25
8	REFERENCE	27

ABSTRACT

Email is the worldwide use of communication application. It is because of the ease of use and faster than other communication application. However, its inability to detect whether the mail content is either spam or ham degrade its performance. Nowadays, lot of cases have been reported regarding stealing of personal information or phishing activities via email from the user. This project will discuss how machine learning help in spam detection. Machine learning is an artificial intelligence application that provides the ability to automatically learn and improve data without being explicitly programmed. Binary classifier will be used to classify the text into two different categories; spam and ham. The algorithm will predict the score more accurately. The objective of developing this project is to detect whether the message is spam or not faster and accurately. The trained dataset model is used to detect the spam mails using multinomial naive bayes classifier which an package from the python library scikit-learn(sklearn).

CHAPTER 1

INTRODUCTION

It takes the following steps:

- The system generates an IP address using flask library after the app.py get runned.
- Then copy the IP address and paste it in any web browser and it will show the webpage
- Then the user are allowed to enter the message which they want to check it as spam or not.
- After it will compare the message with the trained dataset which trained in jupyter notebook and using sklearn library.
- After clicking the check button , whether the message is not spam the webpage showing the message as green color and playing safe sound or it is a spam the webpage showing the message as red color and playing a warning sound.
- After the whole process completed turn off the python code by clicking “ctrl+q” button.

CHAPTER 2

SYSTEM REQUIREMENTS

HARDWARE CONFIGURATION:

System : HP

Processor : INTEL CORE i3 11th GEN

RAM : 8GB RAM

Hard Disk Capacity : 521 GB

SOFTWARE REQUIREMENTS:

Operating System : Windows XP/ Windows 7/8/8.1/10/11

Front end : HTML&CSS

Back end : PYTHON, CSV, IPYTOHN(JUPYTER)

IDE Used : VS CODE, JUPYTER NOTEBOOK

CHAPTER 3

SOFTWARE DESCRIPTION

3.1:PYTHON MODULES USED IN OUR PROJECT :-

3.1.1:FLASK:-

- ❖ Flask is a web framework that provides libraries to build lightweight web applications in python. It is developed by Armin Ronacher who leads an international group of python enthusiasts (POCCO). It is based on web server gateway(WSGI) toolkit and jinja2 template engine. Flask is considered as a micro framework.

3.1.2: SCIKIT-LEARN(SKLEARN):-

- ❖ Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistent interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

3.1.3: PANDAS(PYTHON DATA ANALYSIS LIBRARY):-

- ❖ Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008. Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science.

3.1.4:JOBLIB:-

- ❖ You can use the sklearn joblib integration to distribute certain sklearn tasks over all the cores in your machine for a faster runtime. You can connect joblib to the Dask backend to scale out to a remote cluster for even faster processing times.

3.1.5:PICKLE:-

- ❖ Pickle in Python is primarily used in serializing and deserializing a Python object structure. In other words, it's the process of converting a Python object into a byte stream to store it in a file/database, maintain program state across sessions, or transport data over the network. The pickled byte stream can be used to re-create the original object hierarchy by unpickling the stream.

3.2:HTML & CSS :-

- ❖ HTML (the Hypertext Markup Language) and CSS (Cascading Style Sheets) are two of the core technologies for building Web pages. HTML provides the structure of the page, CSS the (visual and aural) layout, for a variety of devices. Along with graphics and scripting, HTML and CSS are the basis of building Web pages and Web Applications.

3.3:JUPYTER NOTEBOOK:-

- ❖ Jupyter Notebook is an open-source, web-based interactive environment, which allows you to create and share documents that contain live code, mathematical equations, graphics, maps, plots, visualizations, and narrative text. Jupyter Notebook mainly used for Python because Python is used with Artificial Intelligence (AI), Machine learning, as well as Deep learning.

CHAPTER-4

PROJECT DESCRIPTION

4.1DESCRIPTION OF MODULES :-

4.1.1:DATA SET TRAINING MODULES:

- ❖ It is an Backend Module.In this module,First we take the spam message dataset from kaggle.com which is an data science company who have N number of datasets.
- ❖ After the dataset get downloaded open jupyter notebook and set the project folder directory and create a new ipython file.
- ❖ In the ipython file first import pandas library to manipulate the dataset. And read the csv file .
- ❖ After readed the dataset import the sklearn library to train the dataset ,from sklearn use MultinomialNB naive bayes package to classifiers the spam and not spam messages.
- ❖ After the dataset get trained dump the model into pickle file like(spam_model.pkl)using Joblib library and dump() method.

4.2: WEBPAGE DESIGNING MODULE:-

- ❖ It is an Frontend Module . In this module we create an web page using HTML & CSS.
- ❖ Initially the top heading is created and an text field is to read the message form the user.
- ❖ After we create an button to run spam detection module.
- ❖ And we added two different sounds for spam and not spam messages.
- ❖ In CSS we add style to the each and every part of our webpage for heading, text field, button, images.

4.3:BACKEND FLASK MODULE:-

- ❖ Initially imported the flask module to connect the python source to webpage.
- ❖ After imported the pandas and numpy libraries to analyze and manipulate the dataset. Then import the joblib library to access the trained model .
- ❖ Next check posted message with dataset and using if-else statements to print the message is spam or not.
- ❖ After the code get completed save the python code as “app.py”.

4.4:TESTING MODULE:-

- ❖ First run the app.py it will generate an IP address.
- ❖ Then copy the IP address and paste it in any web browser.
- ❖ After paste the IP address it show the spam detection webpage.
- ❖ Then paste the message in the text field that you want to check it.
- ❖ After it check the message with the dataset and report it is spam or not and also play different sounds.
- ❖ After the spam detection process completed come to app.py shell and press “ctrl+q” Quit the webpage.

CHAPTER 5

SCOPE FOR FUTURE ENHANCEMENT

There are several areas in which the effectiveness of spam detection using machine learning could be improved in the future. Some potential areas of focus include:

1. **Incorporating additional features:** Currently, spam detection algorithms often rely on features such as the presence of certain keywords or the structure of the email. However, there may be additional features that could be useful in identifying spam emails, such as the sender's reputation or the presence of unusual formatting.
2. **Developing more advanced machine learning algorithms:** There are a variety of machine learning algorithms that could be used for spam detection, including decision trees, support vector machines, and neural networks. Continuing to research and develop more advanced algorithms may lead to more effective spam detection.
3. **Leveraging unstructured data:** Many spam emails contain unstructured data, such as images or links, that may not be easily analyzed using traditional machine learning techniques. Developing methods for effectively analyzing and incorporating this type of data into spam detection algorithms could lead to further improvements.
4. **Improving the handling of rare or novel spam techniques:** Spammers are constantly evolving their techniques, and it can be challenging for machine learning algorithms to identify new or unusual spam techniques. Developing methods for detecting and adapting to these types of spam could improve the effectiveness of spam detection.

CHAPTER 6

CONCLUSION

Detection of spam is important for securing message and e-mail communication. The accurate detection of spam is a big issue, and many detection methods have been proposed by various researchers. However, these methods have a lack of capability to detect the spam accurately and efficiently. To solve this issue, we have proposed a method for spam detection using machine learning predictive models. The method is applied for the purpose of detection of spam. The experimental results obtained show that the proposed method has a high capability to detect spam. The proposed method achieved 99% accuracy which is high as compared with the other existing methods. Thus, the results suggest that the proposed method is more reliable for accurate and on-time detection of spam, and it will secure the communication systems of messages and e-mails.

CHAPTER 7

APPENDIX

7.1 SOURCE CODE :

7.1.1: JUPYTER NOTEBOOK (IPYTHON SOURCE CODE):-

```
### PANDAS PART
```

```
import pandas as pd
```

```
df = pd.read_csv('spam.csv')
```

```
df.head(5)#it will print the first dataset
```

```
out[1]: Category      Message
0      ham      Go until jurong point, crazy.. Available only ...
1      ham      Ok lar... Joking wif u oni...
2      spam      Free entry in 2 a wkly comp to win FA Cup fina...
3      ham      U dun say so early hor... U c already then say...
4      ham      Nah I don't think he goes to usf, he lives aro...
```

```
df.Category.unique()
```

```
out[2]: array(['ham', 'spam'], dtype=object)
```

```
df['spam'] = df['Category'].apply(lambda x: 1 if x=='spam' else 0)
```

```
df.head(5)
```

```
out[3]:Category      Message      spam
0      ham      Go until jurong point, crazy.. Available only ...      0
1      ham      Ok lar... Joking wif u oni...                      0
2      spam      Free entry in 2 a wkly comp to win FA Cup fina...      1
3      ham      U dun say so early hor... U c already then say...      0
4      ham      Nah I don't think he goes to usf, he lives aro...      0
```

```
### SKLEARN TRAIN_TEST_SPLIT PART
```

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test =
train_test_split(df.Message,df.spam,test_size=0.2,random_state=42)
```

```
len(x_train)
```

```
out[4]: 4457
```

```
len(x_test)
```

```
out[5]: 1115
```

```
### COUNT VECTORIZER PART
```

```
from sklearn.feature_extraction.text import CountVectorizer
v = CountVectorizer()
cv_messages = v.fit_transform(x_train.values)
cv_messages.toarray()[0:5]
```

```
out[6]:array([[0, 0, 0, ..., 0, 0, 0],
              [0, 0, 0, ..., 0, 0, 0],
              [0, 0, 0, ..., 0, 0, 0],
              [0, 0, 0, ..., 0, 0, 0],
              [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

```
### NAIVE BAYES PART
```

```
from sklearn.naive_bayes import MultinomialNB
model = MultinomialNB()
```

```
model.fit(cv_messages,y_train)
```

```
out[7]: MultinomialNB()
```

```
email = [
```

```

    'Upto 30% discount on parking, exclusive offer just for you. Dont miss thi reward!',
    'Ok lar...joking wif u oni...'
]
email_count = v.transform(email)
model.predict(email_count)

out[8]:array([1, 0], dtype=int64)

x_test_count = v.transform(x_test)
model.score(x_test_count,y_test)

out[9]: 0.9919282511210762

### SKLEARN PIPELINE PART

from sklearn.pipeline import Pipeline
clf = Pipeline([
    ('vectorizer',CountVectorizer()),
    ('nb',MultinomialNB())
])
clf.fit(x_train,y_train)

out[10]: Pipeline(steps=[('vectorizer', CountVectorizer()), ('nb', MultinomialNB())])

email = [
    'Upto 30% discount on parking, exclusive offer just for you. Dont miss thi reward!',
    'Ok lar...joking wif u oni...'
]
clf.predict(email)

out[11]: array([1, 0], dtype=int64)

clf.score(x_test,y_test)

out[12]: 0.9919282511210762

```



```
import joblib
joblib.dump(clf, 'spam_model.pkl')'''After this block executed it will generate a python pickle file
like("spam_model.pkl").'''
```

```
out[13]: ['spam_model.pkl']
```

7.1.2: PYTHON SOURCE CODE:-

```
from flask import Flask,render_template,request,jsonify
import pandas as pd
import numpy as np
import joblib

app = Flask(__name__)

model = joblib.load('spam_model.pkl')

@app.route('/',methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        message = request.form.get('message')
        output = model.predict([message])
        if output == [0]:
            result = "This Message is Not a SPAM Message."
        else:
            result = "This Message is a SPAM Message."
        return render_template('index.html', result=result,message=message)

    else:
        return render_template('index.html')

if __name__ == '__main__':
    app.run(debug=True)
```

7.1.3:HTML & CSS SOURCE CODE:-

7.1.3.1 : HTML CODE:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons"
rel="stylesheet">
  <link rel="shortcut icon" href="static/img/logo-w.png"/>
  <title>SPAM Detector Website</title>
</head>
<body>
  <div class="container">

    <div class="logo-heading">
      
    </div>
    <div class="container-data">
      <form action="/" method="post">
        <textarea class="text-box" id="sentence" name="message"
placeholder="Enter a message. Example: Upto 30% off on sale. Buy now!"
onsubmit="return checkforblank()" ></textarea>
        <button type="submit" class="btn">CHECK</button>
      </form>
    </div>
    <br>
    {% if message %}
    <div class="show-result">
      <div class="output"><h3 style="color:black; font-size:22px; font-
weight:bold; font-family:'Courier New', Courier, monospace; text-transform:
uppercase;">{{message}}</h3></div>
      <br>
      {% if result=="This Message is Not a SPAM Message." %}
      <div class="output-logo">
        
      </div>
      {% endif %}
      {% if result=="This Message is Not a SPAM Message." %}
```

```

<div class="alert">
<audio controls autoplay>
  <source src="{{url_for('static', filename='audio/safe.mpeg')}}"
type="audio/mp3">
</audio>
</div>
{% endif %}
{% if result=='This Message is a SPAM Message.' %}
<div class="output-logo">

</div>
{% endif %}
{% if result=='This Message is Not a SPAM Message.' %}
<div class="output-not">{{result}}</div>
{% endif %}
{% if result=='This Message is a SPAM Message.' %}
<div class="output">{{result}}</div>
{% endif %}
{% if result=='This Message is a SPAM Message.' %}
<div class="alert">
<audio controls autoplay>
  <source src="{{url_for('static', filename='audio/warning.mpeg')}}"
type="audio/mp3">
</audio>
</div>
{% endif %}
{% else %}
<h3 style="color:red; padding:30px;font-size:35px; font-weight:bold; font-
family:'Courier New', Courier, monospace; text-transform: uppercase;">Enter A
Message To Check The Message is SPAM or NOT-SPAM...</h3>
{% endif %}
</body>
</html>

```

7.1.3.2:CSS CODE:-

```

* {
border: 0;
box-sizing: border-box;
margin: 0;
}

```

```

.container{
  height: content;
  width: 100%;
  justify-content: center;
  align-items: center;
  display: flex;
  flex-flow: column;
}
.head-nav {
  height:30px;
  width:100%;
  background-color: green;
  display:flex;
  justify-content: center;
  align-items: center;
  padding: 8px;
}
.logo-heading {
  height: 250px;
  width: 250px;
  margin-top: -20px;
  justify-content: center;
  align-items: center;
  display: flex;
}
.logo-heading img {
  height: 100%;
  width: 100%;
}
.container-data {
  height: 200px;
  width: 100%;
  justify-content: center;
  display: flex;
  align-items: center;
  background-color:rgb(236, 214, 214);
  flex-flow: column;
  margin-top: -40px;
  padding-left: 10px;
  padding-right: 10px;
}
.text-box {
  height: 65px;
  width:90%;
  border-radius: 5px;
  border: 2px solid green;
  background-color: rgb(252, 248, 248);

```

```

    margin-top: 15px;
    font-size: 14px;
    font-weight: bold;
}
.btn {
    height: 35px;
    width: 100px;
    border-radius: 5px;
    border: 2px solid black;
    background-color: green;
    color: white;
    margin-top: 15px;
}

.btn:hover {
    color: white;
    background-color: red;
    cursor: pointer;
}

.show-result {
    height: content;
    width: 100%;
    display: flex;
    justify-content: center;
    flex-flow: column wrap;
    background-color: rgb(252, 243, 243);
    margin-bottom: 50px;
    margin-top: -30px;
    padding: 35px;
}

.output {
    height: content;
    width: 100%;
    display: flex;
    justify-content: center;
    align-items: center;
    font-size: 19px;
    font-weight: 300;
    font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida
Sans Unicode', Geneva, Verdana, sans-serif;
    color: red;
    margin-top: 5px;
    padding: 30px;
}

.output-not {
    height: content;
    width: 100%;

```

```

display: flex;
justify-content: center;
align-items: center;
font-size: 19px;
font-weight: 300;
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;
color: green;
margin-top: 3px;
padding: 30px;
}
.output-logo {
height: 100px;
width: 100%;
margin-top: 20px;
justify-content: center;
display: flex;
align-items: center;
}
.output-logo img {
height: 100%;
width: 100%;
}

.head{
height: 40px;
width: 100%;
justify-content: center;
align-items: center;
border-top: 2px solid black;
background-color: yellow;
display: flex;
flex-flow: column;
}
.head h2 {
font-size: 15px;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
color: black;
}
.head-git{
height: 40px;
width: 100%;
justify-content: center;
align-items: center;
background-color: rgb(250, 244, 244);
display: flex;
flex-flow: row;

```

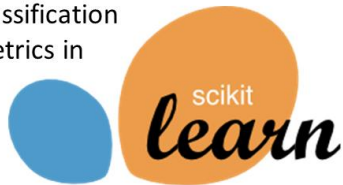
```
    margin-bottom: 5px;
}
.head-git h2 {
    font-size: 18px;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
    color: black;
}
.head-git h2 a {
    text-decoration: none;
    color: blue;
    font-size: 23px;
}
.head-git h2 a:hover {
    color: rgb(211, 47, 41);
    background-color: beige;
}
.alert {
    visibility: hidden;
}
```

7.1.4: DATESET USED IN OUR PROJECT :

URL: <http://bit.ly/datasetfiles> (For easy access we used cloud storage)

7.1.5:IMAGES USED IN THE PROJECT:-

Classification
Metrics in



7.2 SCREEN SHOTS

Classification Metrics in




Enter a message. Example: Upto 30% off on sale. Buy now!

CHECK

ENTER A MESSAGE TO CHECK THE MESSAGE IS SPAM OR NOT-SPAM...


Classification Metrics in



Enter a message. Example: Upto 30% off on sale. Buy now!

CHECK

SORRY TO BE A PAIN. IS IT OK IF WE MEET ANOTHER NIGHT? I SPENT LATE AFTERNOON IN CASUALTY AND THAT MEANS I HAVEN'T DONE ANY OF Y E SHEETS AND THAT. SORRY. STUFF42MORO AND THAT INCLUDES ALL MY TIM





Enter a message. Example: Upto 30% off on sale. Buy now!

CHECK

FREE ENTRY IN 2 A WKLY COMP TO WIN FA CUP FINAL TKTS 21ST MAY 2005. TEXT FA TO 87121 TO RECEIVE ENTRY QUESTION(STD TXT RATE)T&C'S APPLY 08452810075OVER18'S



This Message is a SPAM Message.

CHAPTER 8

REFERENCES

- ✧ For Python: <https://www.w3schools.com/python/>
- ✧ For Dataset: <https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset>
- ✧ For jupyter notebook : <https://www.javatpoint.com/jupyter-notebook>
- ✧ For HTML: <https://www.w3schools.com/html/>
- ✧ For CSS: <https://www.w3schools.com/css/default.asp>
- ✧ For Flask: <https://www.javatpoint.com/flask-tutorial>
- ✧ For pandas: <https://www.javatpoint.com/python-pandas>
- ✧ For Numpy: https://www.w3schools.com/python/numpy/numpy_intro.asp
- ✧ For scikit-learn: <https://scikit-learn.org/stable/tutorial/index.html>
- ✧ For Joblib: <https://joblib.readthedocs.io/en/latest/>