****

**IBM NAAN MUDHALVAN PROJECT**

**PROJECT NAME :-FAKE NEWS DETECTION USING NLP**

**TEAM MEMBERS:-**

|  |  |  |
| --- | --- | --- |
| **Serial no:** | **Team members name** | **Team members email id** |
| 1. | P.Udhaiyalakshmi | udayalaxmi2002@gmail.com |
| 2. | V.Swarnalakshmi | swarnalakshmi065@gmail.com |
| 3. | M.Sneha | keccse21097@kingsedu.ac.in |
| 4. | K.Srivarthini | srivarthinik007@gmail.com |
| 5. | H.Stephy graph | graphstephy171@gmail.com |

Abstract:-

Here we will talk about how one can create an NLP to detect whether the news is real or fake. Nowadays, fake news has become a common trend. Even trusted media houses are known to spread fake news and are losing their credibility. By using the dataset collected were cleaned and processed , however the punctuations and mistakes that existed in the fake news were kept in the text.

**Data preprocessing:-**

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data preprocessing task.

Why do we need Data Preprocessing?

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

It involves below steps:

* Getting the dataset
* Importing libraries
* Importing datasets
* Finding Missing Data
* Encoding Categorical Data
* Splitting dataset into training and test set
* Feature scaling

**Loading dataset:-**

With Python Standard Library, you will be using the module CSV and the function reader() to load your CSV files. Upon loading, the CSV data will be automatically converted to NumPy array which can be used for machine learning.

**Program for loading and preprocessing:**

**Import libraries**

# Modelling Algorithms

from sklearn.naive\_bayes import MultinomialNB

from sklearn.linear\_model import LogisticRegression

from sklearn.linear\_model import PassiveAggressiveClassifier

# Modelling Helpers

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn import metrics

# Computations

import itertools

# Visualization

import matplotlib.pyplot as plt

**Loading dataset**

train = pd.read\_csv("../input/fake-news/train.csv")

test = pd.read\_csv ("../input/fake-news/test.csv")

submit = pd.read\_csv ("../input/fake-news/submit.csv")

train.head()

print(f"Train Shape : **{**train.shape**}**")

print(f"Test Shape : **{**test.shape**}**")

print(f"Submit Shape : **{**submit.shape**}**")

output:

Train Shape : (20800, 5)

Test Shape : (5200, 4)

Submit Shape : (5200, 2)

**Data preprocessing**

train.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 20800 entries, 0 to 20799

Data columns (total 5 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 id 20800 non-null int64

1 title 20242 non-null object

2 author 18843 non-null object

3 text 20761 non-null object

4 label 20800 non-null int64

dtypes: int64(2), object(3)

memory usage: 812.6+ KB

train.isnull().sum()

Output:

id 0

title 558

author 1957

text 39

label 0

dtype: int64

train.dtypes.value\_counts()

Output:

object 3

int64 2

dtype: int64