Web Content Extraction using Machine Learning

In [116…

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import heapq
import pylcs as LCS

from collections import Counter

import numpy as np

from sklearn.preprocessing import StandardScaler
from sklearn.cluster import DBSCAN
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import f1_score,confusion_matrix

from sklearn.decomposition import PCA
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
import matplotlib.pyplot as plt

import seaborn as sns
```

Data Retrieval

In [117…

```
URL = "https://www.ndtv.com/india-news/ndtv-news-on-oxygen-supply-cited-by-delhi
page = requests.get(URL)
soup = BeautifulSoup(page.content, 'html5lib')
print(page.content[0:1000])
```

```
b'<!doctype html><html xmlns="http://www.w3.org/1999/xhtml" itemscope itemtype
="http://schema.org/NewsArticle"><head><title>At Delhi Oxygen Hearing, &quot;Bro
ther Judge&quot; Sends NDTV News On WhatsApp</title><meta name="news_keywords" c
ontent="Coronavirus,Delhi High Court,NDTV" itemprop="keywords"/><meta name="desc
ription" content="The full horror of an oxygen shortage in Delhi&#039;s Covid sp
iral sank in before the High Court today when NDTV&#039;s reports on top hospita
ls running out of oxygen were flagged by one of the judges." itemprop="descripti
on"/><meta name="section" content="india" itemprop="articleSection"/><meta name
="url" content="https://www.ndtv.com/india-news/ndtv-news-on-oxygen-supply-cited
-by-delhi-high-court-2418022" itemprop="url"/><link href="https://www.ndtv.com/i
ndia-news/ndtv-news-on-oxygen-supply-cited-by-delhi-high-court-2418022?amp=1&aka
mai-rum=off" rel="amphtml" ><link href="https://plus.google.com/+NDTV" rel="publ
isher" ><link href="android-app://com.july.nd'
```

Extracting the Meta Content

In [118…

```
meta = []
for tag in soup.findAll(True):
    if tag.name == "meta":
        meta.append(tag.attrs)
```

```
metaContent = []
for dic in meta:
    metaContent.append(dic["content"])

metaContent = []
for dic in meta:
    metaContent.append(dic["content"])
```

Removing Noises in the Meta Content

In [119…

```
noise = ["https", ".com", "com.", "www", "@", ":", "=", "#"]
metaContentFinal = []
for content in metaContent:
    if any(i in content for i in noise):
        pass
    else:
        metaContentFinal.append(content)

metaContentStr = ""

for content in metaContentFinal:
    metaContentStr = metaContentStr + content
```

In [120…

```
print(metaContentStr)
```

```
Coronavirus,Delhi High Court,NDTVThe full horror of an oxygen shortage in Delh
i's Covid spiral sank in before the High Court today when NDTV's reports on top
hospitals running out of oxygen were flagged by one of the judges.indiaCoronavir
us,Delhi High Court,NDTVAt Delhi Oxygen Hearing, "Brother Judge" Sends NDTV News
On WhatsApparticle630473The full horror of an oxygen shortage in Delhi's Covid s
piral sank in before the High Court today when NDTV's reports on top hospitals r
unning out of oxygen were flagged by one of the judges.213741912058651NDTV377869
410NDTV390847563NDTV HDsummary_large_imageAt Delhi Oxygen Hearing, "Brother Judg
e" Sends NDTV News On WhatsAppThe full horror of an oxygen shortage in Delhi's C
ovid spiral sank in before the High Court today when NDTV's reports on top hospi
tals running out of oxygen were flagged by one of the judges.newsAt Delhi Oxygen
Hearing, "Brother Judge" Sends NDTV News On WhatsAppNDTV377869410NDTV HDNDTV1003
0NDTV123At Delhi Oxygen Hearing, "Brother Judge" Sends NDTV News On WhatsAppAt D
elhi Oxygen Hearing, "Brother Judge" Sends NDTV News On WhatsAppAt Delhi Oxygen
Hearing, "Brother Judge" Sends NDTV News On WhatsAppT1M20S1200886
```

Feature Extraction

Tags, Texts, Attributes Extraction

In [121…

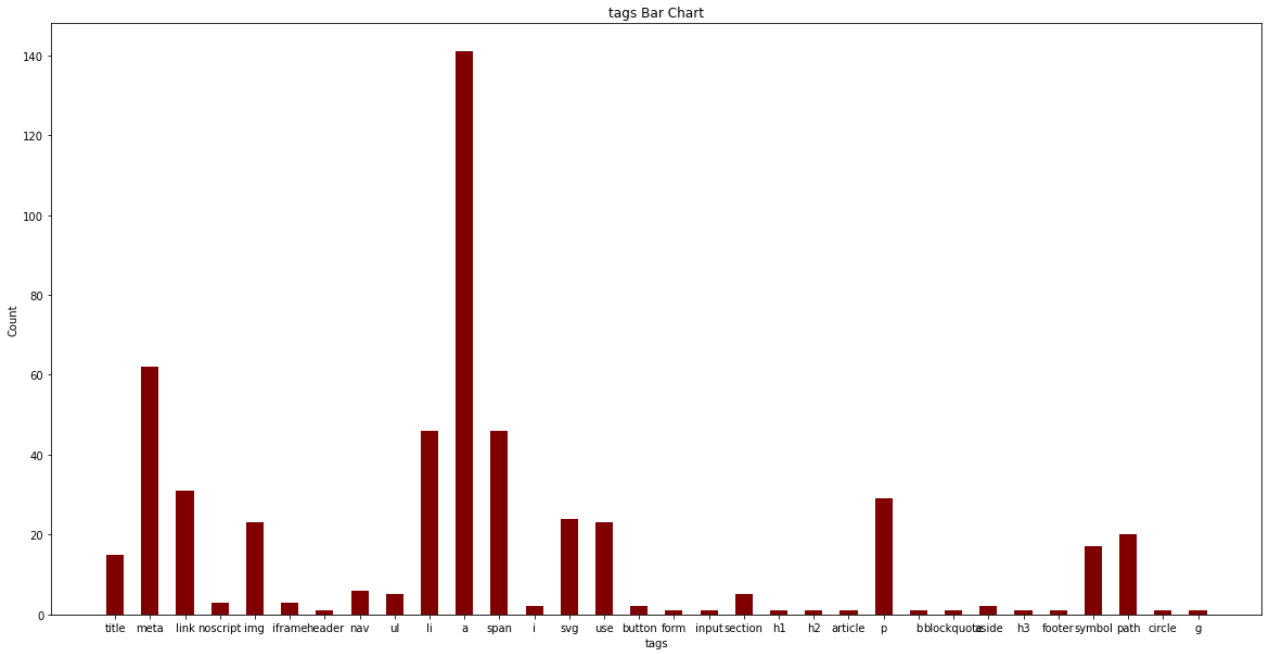
```
tags = []
texts = []
attrs = []

for tag in soup.findAll(True):
    if (tag.name == "style") or (tag.name == "script") or (tag.name == "body") c
        continue
    else:
        tags.append(tag.name)
        texts.append(tag.text)
        attrs.append(tag.attrs)
```

In [122…
```
tags[0:10]
```

Out[122…
```
['title',
 'meta',
 'meta',
 'meta',
 'meta',
 'link',
 'link',
 'link',
 'link',
 'link']
```

In [123…
```
tagsCount = Counter(tags)
uniqueTags = list(tagsCount.keys())
countValues = list(tagsCount.values())
```

In [124…
```
fig = plt.figure(figsize = (20, 10))
plt.bar(uniqueTags, countValues, color ='maroon',width = 0.5)
plt.xlabel("tags")
plt.ylabel("Count")
plt.title("tags Bar Chart")
plt.show()
```



In [125…
```
texts[0:5]
```

Out[125…
```
['At Delhi Oxygen Hearing, "Brother Judge" Sends NDTV News On WhatsApp',
 '',
 '',
 '',
 '']
```

In [126…
```
attrs[0:5]
```

Out[126…
```
[{},
 {'name': 'news_keywords',
  'content': 'Coronavirus,Delhi High Court,NDTV',
  'itemprop': 'keywords'},
 {'name': 'description',
  'content': "The full horror of an oxygen shortage in Delhi's Covid spiral sank
in before the High Court today when NDTV's reports on top hospitals running out
of oxygen were flagged by one of the judges.",
  'itemprop': 'description'},
 {'name': 'section', 'content': 'india', 'itemprop': 'articleSection'},
 {'name': 'url',
  'content': 'https://www.ndtv.com/india-news/ndtv-news-on-oxygen-supply-cited-b
y-delhi-high-court-2418022',
  'itemprop': 'url'}]
```

Feature selection

In [127…
```
texts = pd.Series(texts)
column = list(set(tags))

data = []

for i in range(0, len(tags)):
    data1 = []
    for j in range(0, len(column)):
        if tags[i] == column[j]:
            data1.append(1)
        else:
            data1.append(0)
    data.append(data1)

initialDf = pd.DataFrame(columns=column, data=data)
```

In [128…
```
initialDf.head()
```

Out[128…

	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	h1	article	h3	nav	footer	hea
0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
1	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	
2	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	
3	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	
4	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	

5 rows × 32 columns

Normalization

In [129…
```
def normalize(df):
    std = StandardScaler()

    df = std.fit_transform(df)

    return pd.DataFrame(df)
```

DBSCAN Clustering

In [130…

```
def clusteringDB(df):
    clustering = DBSCAN().fit(df)

    uniqueClusters = set(clustering.labels_)

    df["cluster"] =[i+1 for i in clustering.labels_]

    return uniqueClusters,clustering,df
```

Relavance Score

In [131…

```
def relevanceScore(uniqueClusters,df):
    score = [0 for i in range(len(uniqueClusters))]

    for i in range(0, df.shape[0]):
        score[int(df.loc[i]["cluster"])] = score[int(df.loc[i]["cluster"])] + LC

    return score
```

Finding the cluster with Maximum Score

In [132…

```
def highScore(score):
    maxScoreClusters = heapq.nlargest(2, range(len(score)), key=score.__getitem_

    return maxScoreClusters
```

Marking the Label

In [133…

```
def labelMarking(maxScoreClusters,df):
    label = []

    for i in range(0,df.shape[0]):
        if (int(df.loc[i]["cluster"]) == maxScoreClusters[0]) or (int(df.loc[i][
            label.append(1)
        else:
            label.append(0)

    df["label"] = label

    return df
```

Train and Test Data split

In [134…

```
def trainTestSplit(df):
    df1 = df.drop(columns="cluster")
```

```
    X = df1.drop(columns="label")
    y = df1["label"]

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, ra

    return X_train, X_test, y_train, y_test
```

SVM Classification and Prediction

In [135…

```
def svmModel(X_train,y_train,X_test):
    svmModel = SVC()
    svmModel.fit(X_train,y_train)

    prediction = svmModel.predict(X_test)

    return prediction
```

F1 Score and Confusion Matrix

In [136…

```
def performance(y_test,prediction):
    f1Score = f1_score(y_test,prediction)
    cf_matrix = confusion_matrix(y_test,prediction)

    print("F1 Score of SVM Model")
    print(f1Score)

    print('Confusion Matrix')
    sns.heatmap(cf_matrix, annot=True)
```

In [ ]:

Text Length as Feature

In [137…

```
textSize = []

for text in texts:
    textSize.append(len(text))

initialDf["textSize"] = textSize
```

In [138…

```
initialDf.head()
```

Out[138…

	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	article	h3	nav	footer	header
0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0

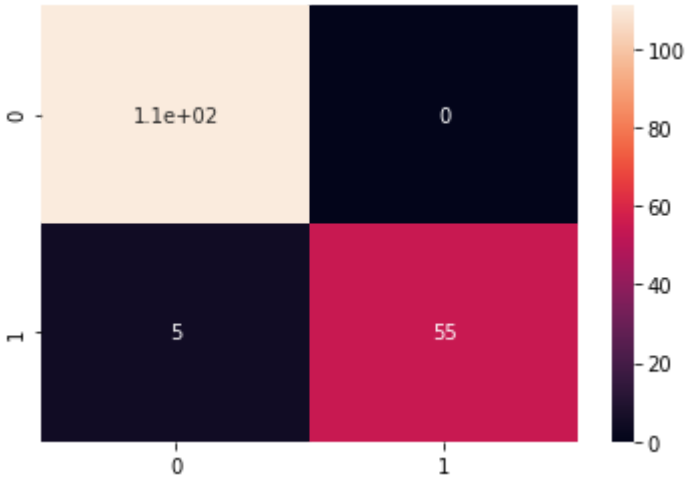
	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	article	h3	nav	footer	header
3	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0

5 rows × 33 columns

In [139…

```
df = normalize(df)
uniqueClusters,clustering,df = clusteringDB(df)

score = relevanceScore(uniqueClusters,df)

maxScoreClusters = highScore(score)

df = labelMarking(maxScoreClusters,df)

X_train, X_test, y_train, y_test = trainTestSplit(df)

prediction = svmModel(X_train,y_train,X_test)

performance(y_test,prediction)
```

```
F1 Score of SVM Model
0.9565217391304348
Confusion Matrix
```



HTML Tags Attribute as Feature

In [140…

```
attrsColumn = []
for i in attrs:
    attrsColumn.append(list(i.keys()))

attrsColumn = sum(attrsColumn, [])

attrsColumn = list(set(attrsColumn))

data = []
for i in attrs:
    idx = []
    for j in i.keys():
```

```
        for k in range(0, len(attrsColumn)):
            if j == attrsColumn[k]:
                idx.append(k)

    data.append([1 if i in idx else 0 for i in range(0, len(attrsColumn))])

dummyDf = pd.DataFrame(columns=attrsColumn, data=data)

finalDf = pd.concat([initialDf, dummyDf], axis=1, join='inner')

df = pd.concat([initialDf, dummyDf], axis=1, join='inner')
```

In [141…

```
df.head()
```

Out[141…

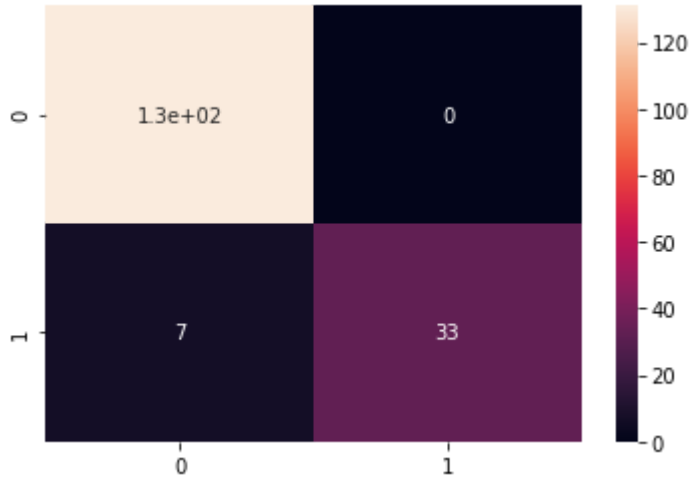
	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	itemscope	dir	target	technolo
0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
4	0	0	1	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 91 columns

In [142…

```
df = normalize(df)
uniqueClusters,clustering,df = clusteringDB(df)

score = relevanceScore(uniqueClusters,df)

maxScoreClusters = highScore(score)

df = labelMarking(maxScoreClusters,df)

X_train, X_test, y_train, y_test = trainTestSplit(df)

prediction = svmModel(X_train,y_train,X_test)

performance(y_test,prediction)
```

```
F1 Score of SVM Model
0.9041095890410958
Confusion Matrix
```



JavaScript Keywords as Feature

In [143…

```
keywords = ["await", "break", "case", "catch", "class", "const", "continue", "de
            "default", "delete", "do", "else", "enum", "export", "extends", "fal
            "finally", "for", "function", "if", "implements", "import", "in", "i
            "let", "new", "null", "package", "private", "protected", "public", "
            "static", "throw", "try", "true", "typeof", "var", "void", "while",
            "(", ")", "{", "}", "]", "[", ";", ".", "\"", "function", "console",
            "window", "href", "\'", "return"]


data=[]
for i in str(texts):
    data1=[]
    for j in keywords:
        c = i.count(j)
        data1.append(c)
    data.append(data1)


dummyDf = pd.DataFrame(columns=keywords, data=data)

finalDf = pd.concat([finalDf, dummyDf], axis=1, join='inner')

df = pd.concat([initialDf, dummyDf], axis=1, join='inner')
```

In [144…

```
df.head()
```

Out[144…

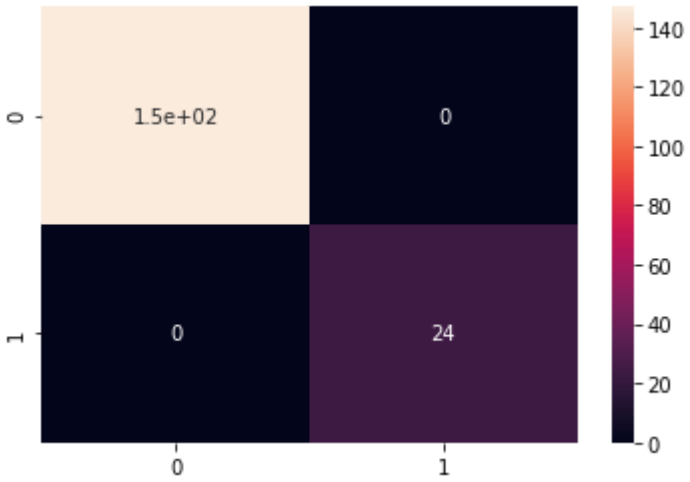
	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	"	function	console	cmd	disp
0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	0	
1	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	
2	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	
3	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	
4	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	

5 rows × 96 columns

In [145…

```
df = normalize(df)
uniqueClusters,clustering,df = clusteringDB(df)

score = relevanceScore(uniqueClusters,df)

maxScoreClusters = highScore(score)

df = labelMarking(maxScoreClusters,df)

X_train, X_test, y_train, y_test = trainTestSplit(df)

prediction = svmModel(X_train,y_train,X_test)

performance(y_test,prediction)
```

```
F1 Score of SVM Model
1.0
Confusion Matrix
```



HTML tags CSS Class as attribute

In [146…

```
tagsClass = []

for attr in attrs:
    tagsClass.append(attr.get("class"))

classList = []
for i in tagsClass:
    if i:
        for j in i:
            classList.append(j)

uniqueClass = list(set(classList))

data = []
zeroes = [0 for i in range(len(uniqueClass))]
for iClass in tagsClass:
    row=[]
    if iClass:
        for j in uniqueClass:
            if j in iClass:
                row.append(1)
```

```
            else:
                row.append(0)
        data.append(row)
    else:
        data.append(zeroes)


dummyDf = pd.DataFrame(columns=uniqueClass,data=data)

finalDf = pd.concat([finalDf, dummyDf], axis=1, join='inner')

df = pd.concat([initialDf, dummyDf], axis=1, join='inner')
```

In [147…

```
df.head()
```

Out[147…

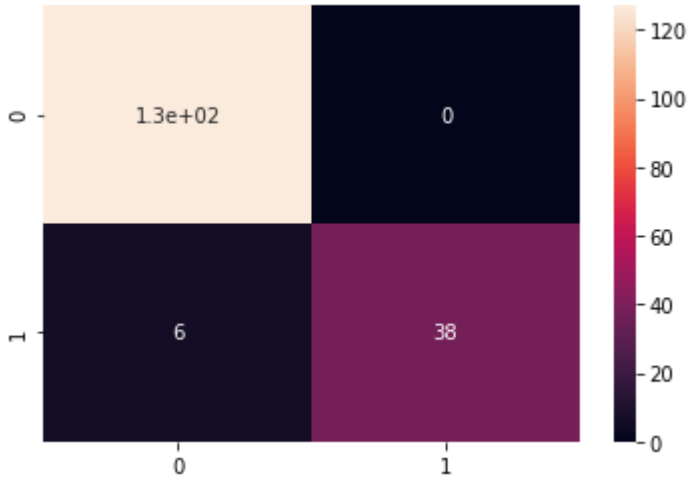
	svg	title	meta	form	use	noscript	li	blockquote	p	link	...	hid_sml-dvc	imgbrd	reddit	adc sectio
0	0	1	0	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
4	0	0	1	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 121 columns

In [148…

```
df = normalize(df)
uniqueClusters,clustering,df = clusteringDB(df)

score = relevanceScore(uniqueClusters,df)

maxScoreClusters = highScore(score)

df = labelMarking(maxScoreClusters,df)

X_train, X_test, y_train, y_test = trainTestSplit(df)

prediction = svmModel(X_train,y_train,X_test)

performance(y_test,prediction)
```

```
F1 Score of SVM Model
0.9268292682926829
Confusion Matrix
```

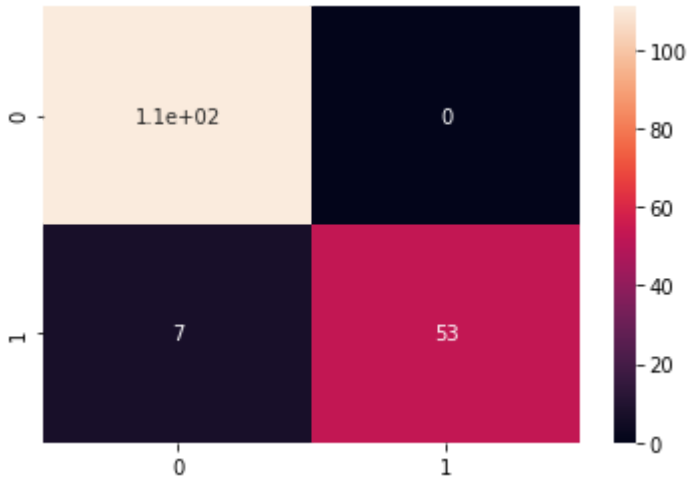


Combined DataSet

In [149…

```
df = normalize(finalDf)
uniqueClusters,clustering,df = clusteringDB(df)

score = relevanceScore(uniqueClusters,df)

maxScoreClusters = highScore(score)

df = labelMarking(maxScoreClusters,df)

X_train, X_test, y_train, y_test = trainTestSplit(df)

prediction = svmModel(X_train,y_train,X_test)

performance(y_test,prediction)
```

```
F1 Score of SVM Model
0.9380530973451328
Confusion Matrix
```



In [ ]:

In [ ]: