Week 4 : task

STEP 1 : IMPORTING ALL REQUIRED LIBRARY

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score,
f1_score, confusion_matrix, classification_report
```

STEP 2 : Loading the dataset

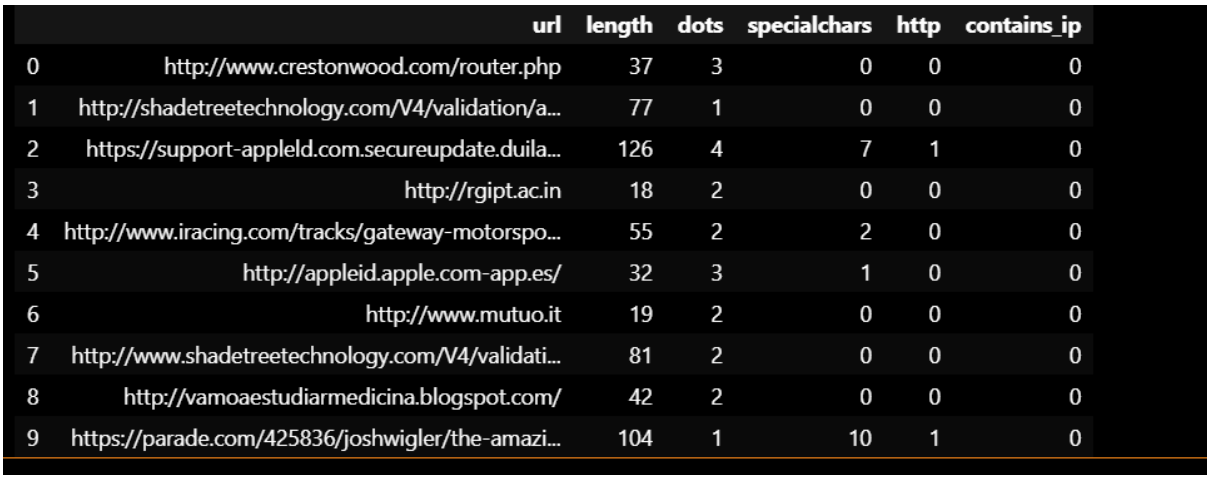
```
df = pd.read_csv("dataset_phishing.csv")
```

STEP 3 : FEATURE SELECTION

```
# Feature 1: Length of URL
df['length'] = df['url'].apply(len)

# Feature 2: Number of dots in URL
df['dots'] = df['url'].apply(lambda x: x.count('.'))

# Feature 3: Count of basic special characters
```

```
df['specialchars'] = df['url'].apply(lambda x: x.count('-') + x.count('@') +
x.count('&') + x.count('%') + x.count('?') + x.count('='))


# Feature 4: Check if 'https' is present

df['http'] = df['url'].apply(lambda x: 1 if 'https' in x else 0)


# Feature 5: Check if the URL contains numbers and dots in domain (very basic
IP pattern)

def feature(url):

    parts = url.split('/')

    if len(parts) > 2:

        domain = parts[2]

        return 1 if all(c.isdigit() or c == '.' for c in domain) and domain.count('.')
== 3 else 0

    return 0


df['contains_ip'] = df['url'].apply(feature)


# Display the first few rows to verify

df[['url', 'length', 'dots', 'specialchars', 'http', 'contains_ip']].head(10)
```

OUTPUT:

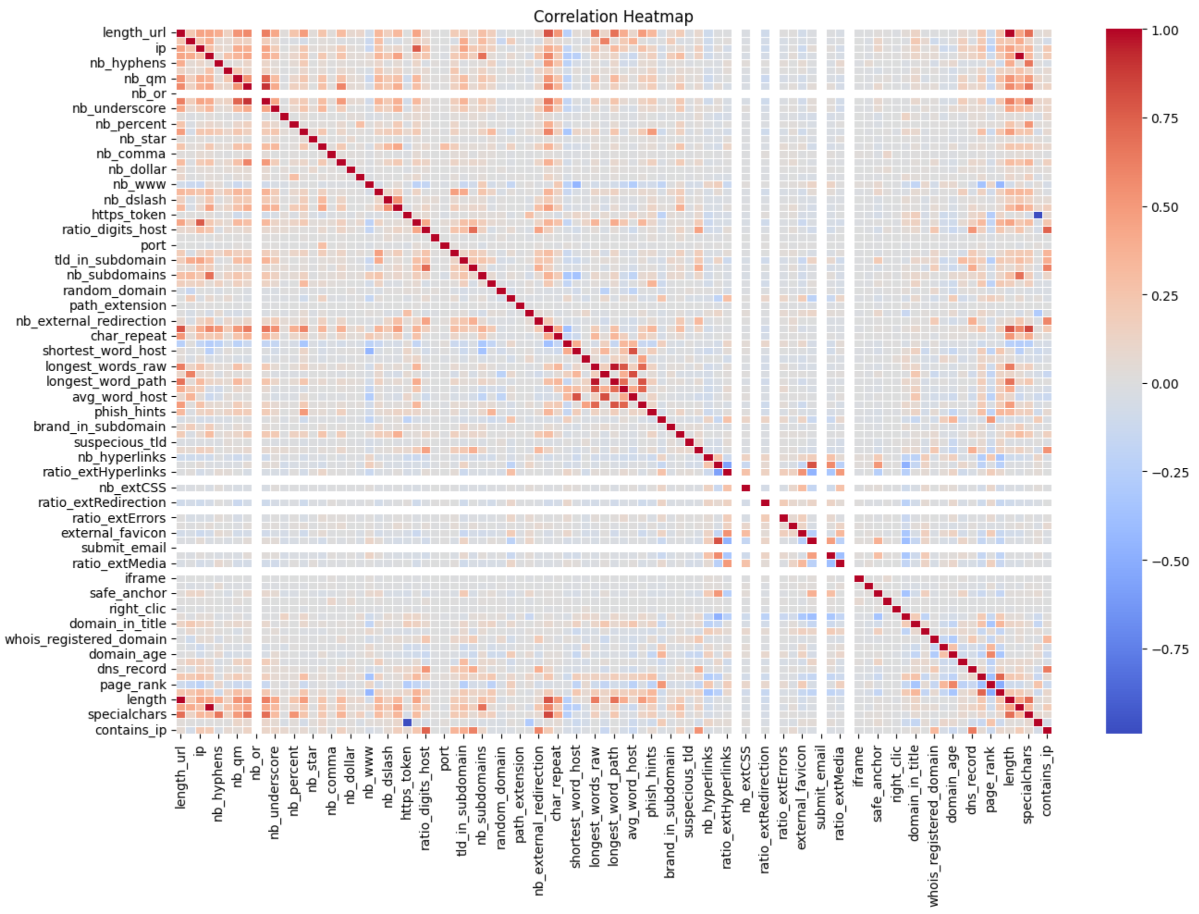


	url	length	dots	specialchars	http	contains_ip
0	http://www.crestonwood.com/router.php	37	3	0	0	0
1	http://shadetreetechnology.com/V4/validation/a...	77	1	0	0	0
2	https://support-appleld.com.secureupdate.duila...	126	4	7	1	0
3	http://rgipt.ac.in	18	2	0	0	0
4	http://www.iracing.com/tracks/gateway-motorspo...	55	2	2	0	0
5	http://appleid.apple.com-app.es/	32	3	1	0	0
6	http://www.mutuo.it	19	2	0	0	0
7	http://www.shadetreetechnology.com/V4/validati...	81	2	0	0	0
8	http://vamoaestudiarmedicina.blogspot.com/	42	2	0	0	0
9	https://parade.com/425836/joshwigler/the-amazi...	104	1	10	1	0

STEP 4 : CORRELATION

```
# Create correlation matrix
correlation_matrix = df.corr()


# Plot the heatmap
plt.figure(figsize=(14, 10))
sns.heatmap(correlation_matrix, cmap='coolwarm', annot=False, linewidths=0.5)
plt.title("Correlation Heatmap")
plt.tight_layout()
plt.show()
```

OUTPUT:

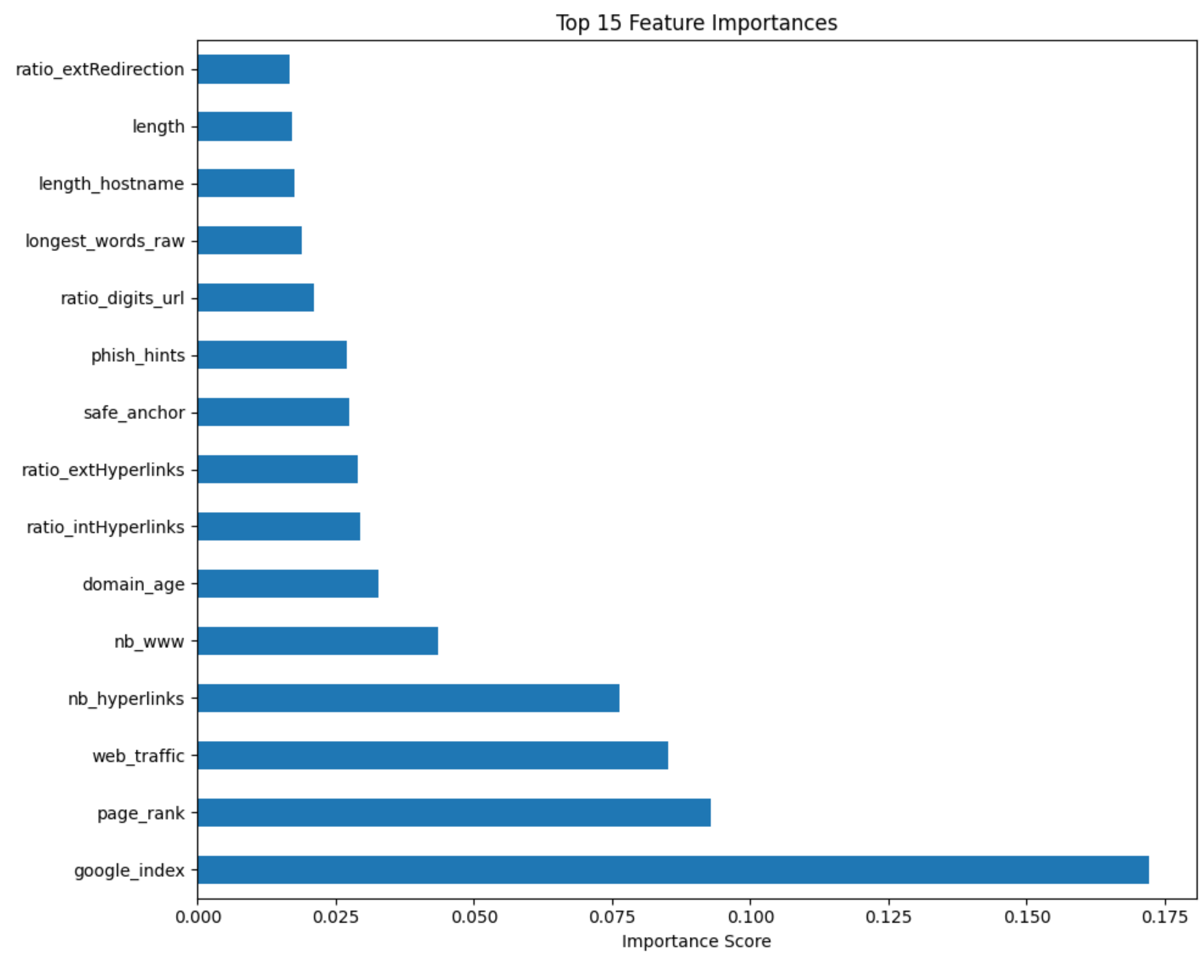


STEP 5 : SELECTING TOP FETAURE

```
# Prepare data

X = df.drop(['status', 'url'], axis=1, errors='ignore')

y = df['status']


# Train Random Forest model

model = RandomForestClassifier(random_state=42)

model.fit(X, y)


# Get and plot feature importances

importances = pd.Series(model.feature_importances_, index=X.columns)

importances.nlargest(15).plot(kind='barh', figsize=(10, 8), title="Top 15 Feature
Importances")
```

```
plt.xlabel("Importance Score")
plt.tight_layout()
plt.show()
```

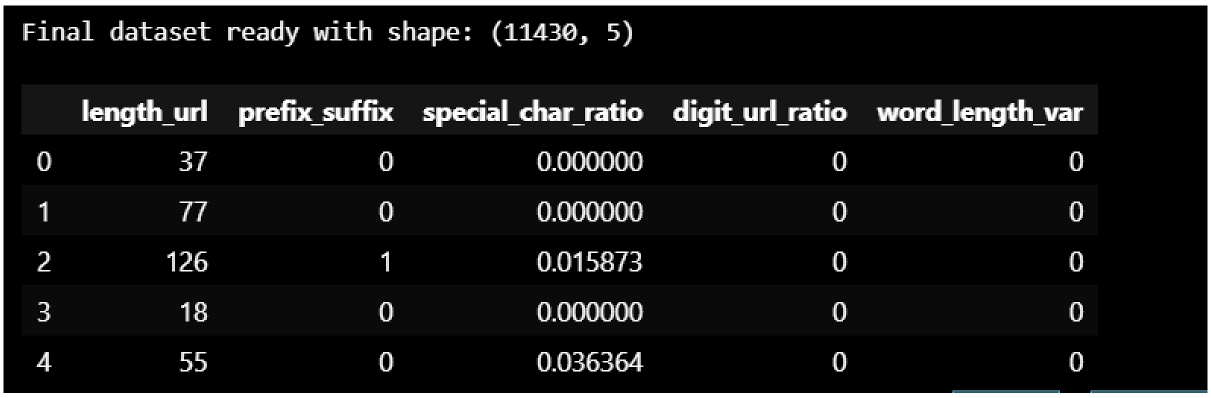
OUTPUT:



STEP 6 : REMOVING UNESSARY

```
# Define the list of special character features
all_special_char_cols = [
    'nb_hyphens', 'nb_at', 'nb_qm', 'nb_and', 'nb_or', 'nb_eq', 'nb_underscore',
    'nb_tilde', 'nb_percent', 'nb_slash', 'nb_star', 'nb_colon', 'nb_comma',
    'nb_semicolumn', 'nb_dollar', 'nb_space'
```

```
]

# Select only available columns from df
available_special_chars = [col for col in all_special_char_cols if col in df.columns]

# Safely create engineered features
if available_special_chars:
    df['special_char_ratio'] = df[available_special_chars].sum(axis=1) / df['length_url']
else:
    df['special_char_ratio'] = 0  # or drop this if not wanted

# Digit-to-length ratio (already present as 'ratio_digits_url')
df['digit_url_ratio'] = df['ratio_digits_url'] if 'ratio_digits_url' in df.columns else 0

# Word length variance
word_cols = ['longest_words_raw', 'shortest_words_raw', 'avg_words_raw']
existing_word_cols = [col for col in word_cols if col in df.columns]
df['word_length_var'] = df[existing_word_cols].std(axis=1) if len(existing_word_cols) >= 2
else 0

# Final selected features
selected_features = [
    col for col in [
        'length_url',
        'nb_dots',
        'http',
        'prefix_suffix',
        'random_domain',
        'shortening_service',
        'char_repeat',
        'domain_in_brand',
        'suspecious_tld',
        'web_traffic',
        'dns_record',
        'google_index',
        'page_rank',
        'domain_age',
        'domain_registration_length',
        'special_char_ratio',
        'digit_url_ratio',
        'word_length_var',
        'status'
    ] if col in df.columns
]

df_final = df[selected_features]

# Output
print("Final dataset ready with shape:", df_final.shape)
display(df_final.head())
```



```
Final dataset ready with shape: (11430, 5)

   length_url  prefix_suffix  special_char_ratio  digit_url_ratio  word_length_var
0          37              0            0.000000                0                0
1          77              0            0.000000                0                0
2         126              1            0.015873                0                0
3          18              0            0.000000                0                0
4          55              0            0.036364                0                0
```

```
# Check if 'status' column is present
if 'status' not in df.columns:
    print("'status' column (target) is missing from the original dataset. Please ensure it is loaded.")
else:
    df_final['status'] = df['status']
```

```
OUTPUT: 'status' column (target) is missing from the original dataset. Please ensure it is loaded.
```

STEP 7 :

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix

# Load the dataset
df = pd.read_csv('dataset_phishing.csv')

# Define useful special character columns
all_special_char_cols = [
    'nb_hyphens', 'nb_at', 'nb_qm', 'nb_and', 'nb_or', 'nb_eq', 'nb_underscore',
    'nb_tilde', 'nb_percent', 'nb_slash', 'nb_star', 'nb_colon', 'nb_comma',
    'nb_semicolumn', 'nb_dollar', 'nb_space'
]
available_special_chars = [col for col in all_special_char_cols if col in df.columns]

#  Feature engineering
if available_special_chars:
    df['special_char_ratio'] = df[available_special_chars].sum(axis=1) / df['length_url']
else:
    df['special_char_ratio'] = 0

df['digit_url_ratio'] = df['ratio_digits_url'] if 'ratio_digits_url' in df.columns else 0

word_cols = ['longest_words_raw', 'shortest_words_raw', 'avg_words_raw']
existing_word_cols = [col for col in word_cols if col in df.columns]
df['word_length_var'] = df[existing_word_cols].std(axis=1) if len(existing_word_cols) >= 2
else 0

# Select final features + status
final_cols = [
    'length_url', 'prefix_suffix', 'special_char_ratio',
    'digit_url_ratio', 'word_length_var', 'status'
]
df_final = df[[col for col in final_cols if col in df.columns]].copy()

# Checking if target column does exists
if 'status' not in df_final.columns:
    raise ValueError("'status' column is missing. Please verify your data source contains
labels.")
```

STEP 8: CTEATING NEW DATASET WITH ADDITIONAL COLUMN

```
# Save the updated dataset for modeling
df.to_csv("new_dataset.csv", index=False)
print("Refined dataset saved as 'new_dataset.csv'")
```

STEP 9 : OUTLINE OF NEW DATASET

```
data = pd.read_csv("new_dataset.csv")
print(data.shape)
print(data.info())
print(data.describe())
```

```
(11430, 92)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11430 entries, 0 to 11429
Data columns (total 92 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   url                         11430 non-null  object
 1   length_url                  11430 non-null  int64
 2   length_hostname             11430 non-null  int64
 3   ip                          11430 non-null  int64
 4   nb_dots                     11430 non-null  int64
 5   nb_hyphens                  11430 non-null  int64
 6   nb_at                       11430 non-null  int64
 7   nb_qm                       11430 non-null  int64
 8   nb_and                      11430 non-null  int64
 9   nb_or                       11430 non-null  int64
 10  nb_eq                       11430 non-null  int64
 11  nb_underscore               11430 non-null  int64
 12  nb_tilde                    11430 non-null  int64
 13  nb_percent                  11430 non-null  int64
 14  nb_slash                    11430 non-null  int64
 15  nb_star                     11430 non-null  int64
 16  nb_colon                    11430 non-null  int64
```

```
17  nb_comma                  11430 non-null  int64

18  nb_semicolumn             11430 non-null  int64

19  nb_dollar                 11430 non-null  int64

20  nb_space                  11430 non-null  int64

21  nb_www                    11430 non-null  int64

22  nb_com                    11430 non-null  int64

23  nb_dslash                 11430 non-null  int64

24  http_in_path              11430 non-null  int64

25  https_token               11430 non-null  int64

26  ratio_digits_url          11430 non-null  float64

27  ratio_digits_host         11430 non-null  float64

28  punycode                  11430 non-null  int64

29  port                      11430 non-null  int64

30  tld_in_path               11430 non-null  int64

31  tld_in_subdomain          11430 non-null  int64

32  abnormal_subdomain        11430 non-null  int64

33  nb_subdomains             11430 non-null  int64

34  prefix_suffix             11430 non-null  int64

35  random_domain             11430 non-null  int64

36  shortening_service        11430 non-null  int64

37  path_extension            11430 non-null  int64

38  nb_redirection            11430 non-null  int64

39  nb_external_redirection   11430 non-null  int64

40  length_words_raw          11430 non-null  int64

41  char_repeat               11430 non-null  int64

42  shortest_words_raw        11430 non-null  int64

43  shortest_word_host        11430 non-null  int64

44  shortest_word_path        11430 non-null  int64

45  longest_words_raw         11430 non-null  int64

46  longest_word_host         11430 non-null  int64
```

```
47  longest_word_path          11430 non-null  int64
48  avg_words_raw              11430 non-null  float64
49  avg_word_host              11430 non-null  float64
50  avg_word_path              11430 non-null  float64
51  phish_hints                11430 non-null  int64
52  domain_in_brand            11430 non-null  int64
53  brand_in_subdomain         11430 non-null  int64
54  brand_in_path              11430 non-null  int64
55  suspecious_tld             11430 non-null  int64
56  statistical_report         11430 non-null  int64
57  nb_hyperlinks              11430 non-null  int64
58  ratio_intHyperlinks        11430 non-null  float64
59  ratio_extHyperlinks        11430 non-null  float64
60  ratio_nullHyperlinks       11430 non-null  int64
61  nb_extCSS                  11430 non-null  int64
62  ratio_intRedirection       11430 non-null  int64
63  ratio_extRedirection       11430 non-null  float64
64  ratio_intErrors            11430 non-null  int64
65  ratio_extErrors            11430 non-null  float64
66  login_form                 11430 non-null  int64
67  external_favicon           11430 non-null  int64
68  links_in_tags              11430 non-null  float64
69  submit_email               11430 non-null  int64
70  ratio_intMedia             11430 non-null  float64
71  ratio_extMedia             11430 non-null  float64
72  sfh                        11430 non-null  int64
73  iframe                     11430 non-null  int64
74  popup_window               11430 non-null  int64
75  safe_anchor                11430 non-null  float64
76  onmouseover                11430 non-null  int64
```

```
 77  right_clic                 11430 non-null  int64
 78  empty_title                11430 non-null  int64
 79  domain_in_title            11430 non-null  int64
 80  domain_with_copyright      11430 non-null  int64
 81  whois_registered_domain    11430 non-null  int64
 82  domain_registration_length 11430 non-null  int64
 83  domain_age                 11430 non-null  int64
 84  web_traffic                11430 non-null  int64
 85  dns_record                 11430 non-null  int64
 86  google_index               11430 non-null  int64
 87  page_rank                  11430 non-null  int64
 88  status                     11430 non-null  int64
 89  special_char_ratio         11430 non-null  float64
 90  digit_url_ratio            11430 non-null  float64
 91  word_length_var            11430 non-null  float64
dtypes: float64(16), int64(75), object(1)
memory usage: 8.0+ MB
None
        length_url  length_hostname           ip       nb_dots  \
count  11430.000000     11430.000000  11430.000000  11430.000000
mean      61.126684        21.090289      0.150569      2.480752
std       55.297318        10.777171      0.357644      1.369686
min       12.000000         4.000000      0.000000      1.000000
25%       33.000000        15.000000      0.000000      2.000000
50%       47.000000        19.000000      0.000000      2.000000
75%       71.000000        24.000000      0.000000      3.000000
max     1641.000000       214.000000      1.000000     24.000000

        nb_hyphens         nb_at         nb_qm        nb_and     nb_or  \
count  11430.000000  11430.000000  11430.000000  11430.000000  11430.0
```

```
mean       0.997550      0.022222      0.141207      0.162292      0.0
std       2.087087      0.155500      0.364456      0.821337      0.0
min       0.000000      0.000000      0.000000      0.000000      0.0
25%       0.000000      0.000000      0.000000      0.000000      0.0
50%       0.000000      0.000000      0.000000      0.000000      0.0
75%       1.000000      0.000000      0.000000      0.000000      0.0
max      43.000000      4.000000      3.000000     19.000000      0.0

             nb_eq  ...  domain_registration_length    domain_age  \
count  11430.000000  ...                11430.000000  11430.000000
mean       0.293176  ...                  492.532196   4062.543745
std        0.998317  ...                  814.769415   3107.784600
min        0.000000  ...                   -1.000000    -12.000000
25%        0.000000  ...                   84.000000    972.250000
50%        0.000000  ...                  242.000000   3993.000000
75%        0.000000  ...                  449.000000   7026.750000
max       19.000000  ...                29829.000000  12874.000000

        web_traffic    dns_record  google_index     page_rank        status  \
count  1.143000e+04  11430.000000  11430.000000  11430.000000  11430.000000
mean   8.567566e+05      0.020122      0.533946      3.185739      0.500000
std    1.995606e+06      0.140425      0.498868      2.536955      0.500022
min    0.000000e+00      0.000000      0.000000      0.000000      0.000000
25%    0.000000e+00      0.000000      0.000000      1.000000      0.000000
50%    1.651000e+03      0.000000      1.000000      3.000000      0.500000
75%    3.738455e+05      0.000000      1.000000      5.000000      1.000000
max    1.076799e+07      1.000000      1.000000     10.000000      1.000000

       special_char_ratio  digit_url_ratio  word_length_var
count        11430.000000     11430.000000     11430.000000
```

```
mean       0.133989     0.053137     6.314628
std        0.035258     0.089363    12.038260
min        0.011385     0.000000     0.000000
25%        0.112150     0.000000     3.000000
50%        0.132075     0.000000     4.041452
75%        0.153846     0.079365     6.512061
max        0.452381     0.723881   456.132538

[8 rows x 91 columns]
```