

Model Development Phase Template

Date	1 October 2024
Team ID	LTVIP2024TMID24838
Project Title	Detection of Phishing Websites from URLs
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

Initial Model Training Code:

1. RANDOM FOREST

```
model = RandomForestClassifier(random_state=0)
```

```
model.fit(x_tr,y_tr)
```

```
RandomForestClassifier
RandomForestClassifier(random_state=0)
```

```
y_p=model.predict(x_t)
y_p1=model.predict(x_tr)
```

```
rf1=accuracy_score(y_t, y_p)
rf2=accuracy_score(y_tr, y_p1)
```

```
print("Accuracy:", rf1)
print("Accuracy:",rf2)
```

```
Accuracy: 0.9642695612844867
Accuracy: 1.0
```

2.KNN CLASSIFICATION:

```
from sklearn.neighbors import KNeighborsClassifier
kmodel=KNeighborsClassifier()
```

```
kmodel.fit(x_tr,y_tr)
```

```
KNeighborsClassifier
KNeighborsClassifier()
```

```
y_predk = kmodel.predict(x_t)
y_pred_train = kmodel.predict(x_tr)
```

```
knn=accuracy_score(y_t,y_predk)
knn1=accuracy_score(y_tr,y_pred_train)
print("Accuracy score for testing data: ",knn)
print("Accuracy score for training data: ",knn1)
```

```
Accuracy score for testing data: 0.6142017186793306
Accuracy score for training data: 0.7800768882858435
```

3.LOGISTIC REGRESSION

```
from sklearn.linear_model import LogisticRegression  
lr=LogisticRegression(max_iter=1000)
```

```
lr.fit(x_tr,y_tr)
```

```
▼ LogisticRegression ⓘ ⓘ  
LogisticRegression()
```

```
y_plr=lr.predict(x_t)  
y_plr1=lr.predict(x_tr)
```

```
log_reg=accuracy_score(y_t,y_plr)  
log_reg1=accuracy_score(y_tr,y_plr1)  
print(log_reg)  
print(log_reg1)
```

```
0.9181365897783809  
0.9314789687924017
```

4.DECISIONTREE CLASSIFIER:

```
from sklearn.tree import DecisionTreeClassifier  
dt=DecisionTreeClassifier()  
dt.fit(x_tr,y_tr)
```

```
▼ DecisionTreeClassifier ⓘ ⓘ  
DecisionTreeClassifier()
```

```
y_pred6=dt.predict(x_t)  
y_pred62=dt.predict(x_tr)
```

```
dec_tree=accuracy_score(y_t,y_pred6)  
print(dec_tree)  
dec_tree1=accuracy_score(y_tr,y_pred62)  
print(dec_tree1)
```

```
0.9479873360470376  
1.0
```

Model Validation and Evaluation Report:

Model	Classification Report	Acc ura cy	Confusion Matrix																														
Random Forest	<pre>print(classification_report(y_t, y_p))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>-1</td><td>0.97</td><td>0.95</td><td>0.96</td><td>1014</td></tr><tr><td>1</td><td>0.96</td><td>0.98</td><td>0.97</td><td>1197</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.96</td><td>2211</td></tr><tr><td>macro avg</td><td>0.97</td><td>0.96</td><td>0.96</td><td>2211</td></tr><tr><td>weighted avg</td><td>0.96</td><td>0.96</td><td>0.96</td><td>2211</td></tr></tbody></table>		precision	recall	f1-score	support	-1	0.97	0.95	0.96	1014	1	0.96	0.98	0.97	1197	accuracy			0.96	2211	macro avg	0.97	0.96	0.96	2211	weighted avg	0.96	0.96	0.96	2211	96	<pre>confusion_matrix(y_t,y_p)</pre> <pre>array([[962, 52], [27, 1170]])</pre>
	precision	recall	f1-score	support																													
-1	0.97	0.95	0.96	1014																													
1	0.96	0.98	0.97	1197																													
accuracy			0.96	2211																													
macro avg	0.97	0.96	0.96	2211																													
weighted avg	0.96	0.96	0.96	2211																													
KNN	<pre>print(classification_report(y_t,y_predk))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>-1</td><td>0.59</td><td>0.54</td><td>0.56</td><td>1014</td></tr><tr><td>1</td><td>0.63</td><td>0.68</td><td>0.66</td><td>1197</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.61</td><td>2211</td></tr><tr><td>macro avg</td><td>0.61</td><td>0.61</td><td>0.61</td><td>2211</td></tr><tr><td>weighted avg</td><td>0.61</td><td>0.61</td><td>0.61</td><td>2211</td></tr></tbody></table>		precision	recall	f1-score	support	-1	0.59	0.54	0.56	1014	1	0.63	0.68	0.66	1197	accuracy			0.61	2211	macro avg	0.61	0.61	0.61	2211	weighted avg	0.61	0.61	0.61	2211	61	<pre>confusion_matrix(y_t,y_predk)</pre> <pre>array([[544, 470], [383, 814]])</pre>
	precision	recall	f1-score	support																													
-1	0.59	0.54	0.56	1014																													
1	0.63	0.68	0.66	1197																													
accuracy			0.61	2211																													
macro avg	0.61	0.61	0.61	2211																													
weighted avg	0.61	0.61	0.61	2211																													
Logistic Regression	<pre>print(classification_report(y_t,y_plr))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>-1</td><td>0.93</td><td>0.89</td><td>0.91</td><td>1014</td></tr><tr><td>1</td><td>0.91</td><td>0.94</td><td>0.93</td><td>1197</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.92</td><td>2211</td></tr><tr><td>macro avg</td><td>0.92</td><td>0.92</td><td>0.92</td><td>2211</td></tr><tr><td>weighted avg</td><td>0.92</td><td>0.92</td><td>0.92</td><td>2211</td></tr></tbody></table>		precision	recall	f1-score	support	-1	0.93	0.89	0.91	1014	1	0.91	0.94	0.93	1197	accuracy			0.92	2211	macro avg	0.92	0.92	0.92	2211	weighted avg	0.92	0.92	0.92	2211	92	<pre>confusion_matrix(y_t,y_plr)</pre> <pre>array([[906, 108], [73, 1124]])</pre>
	precision	recall	f1-score	support																													
-1	0.93	0.89	0.91	1014																													
1	0.91	0.94	0.93	1197																													
accuracy			0.92	2211																													
macro avg	0.92	0.92	0.92	2211																													
weighted avg	0.92	0.92	0.92	2211																													
Decision tree classifier	<pre>print(classification_report(y_t,y_pred6))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>-1</td><td>0.95</td><td>0.94</td><td>0.94</td><td>1014</td></tr><tr><td>1</td><td>0.95</td><td>0.96</td><td>0.95</td><td>1197</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.95</td><td>2211</td></tr><tr><td>macro avg</td><td>0.95</td><td>0.95</td><td>0.95</td><td>2211</td></tr><tr><td>weighted avg</td><td>0.95</td><td>0.95</td><td>0.95</td><td>2211</td></tr></tbody></table>		precision	recall	f1-score	support	-1	0.95	0.94	0.94	1014	1	0.95	0.96	0.95	1197	accuracy			0.95	2211	macro avg	0.95	0.95	0.95	2211	weighted avg	0.95	0.95	0.95	2211	95	<pre>confusion_matrix(y_t,y_pred6)</pre> <pre>array([[951, 63], [52, 1145]])</pre>
	precision	recall	f1-score	support																													
-1	0.95	0.94	0.94	1014																													
1	0.95	0.96	0.95	1197																													
accuracy			0.95	2211																													
macro avg	0.95	0.95	0.95	2211																													
weighted avg	0.95	0.95	0.95	2211																													

