



Model Development Phase Template

Date	11-07-2024	
Team ID	739736	
Project Title	SMOKE DETECTION USING IOT DATASET	
Maximum Marks	4 Marks	

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

Model Building [] from sklearn.tree import DecisionTreeClassifier from sklearn.metrics import accuracy_score,classification_report clf = DecisionTreeClassifier() # Train the classifier on the training data $clf.fit(x_train, y_train)$ # Make predictions on the testing data y_pred = clf.predict(x_test) # Evaluate the classifier report = classification_report(y_test, y_pred) print("Classification Report:\n", report) → Classification Report: precision recall f1-score support 1.00 1.00 1.00 1.00 1.00 1.00 0 3605 1 8921 12526 12526 1.00 accuracy macro avg 1.00 1.00 1.00 weighted avg 1.00 1.00 1.00 12526





```
[ ] from sklearn.linear_model import LogisticRegression
    from sklearn.metrics import accuracy score, classification report
#Initializing the model
    model_lr = LogisticRegression()
    model_lr.fit(x_train,y_train)
    lr_pred_test=model_lr.predict(x_test)
    lr\_pred\_train=model\_lr.predict(x\_train)
    test_acc_lr=accuracy_score(y_test,lr_pred_test)
    train_acc_lr=accuracy_score(y_train,lr_pred_train)
    print('Logistic Regression test accuracy: ',test_acc_lr)
    print(classification_report(y_test,lr_pred_test))
→ Logistic Regression test accuracy: 0.9691840970780776
                precision recall f1-score support
                   0.94 0.95
0.98 0.98
                                    0.95
0.98
                                              3605
                                     0.97
                                             12526
       accuracy
      macro avg 0.96 0.96 0.96 ighted avg 0.97 0.97 0.97
                                             12526
    weighted avg
                                             12526
  KNN classifier
       from sklearn.neighbors import KNeighborsClassifier
        knn=KNeighborsClassifier()
        knn.fit(x_train,y_train)
        knn_pred_test=knn.predict(x_test)
        knn_pred_train=knn.predict(x_train)
        test_acc_knn=accuracy_score(y_test,knn_pred_test)
        train_acc_knn=accuracy_score(y_train,knn_pred_train)
        print('KNN Test Accuracy is: ',test_acc_knn)
        print(classification_report(y_test,knn_pred_test))
   → KNN Test Accuracy is: 0.9992814944914578
                        precision
                                   recall f1-score
                                                          support
                             1.00
                                       1.00
                                                    1.00
                    Θ
                                                               3605
                             1.00
                                       1.00
                                                    1.00
                                                               8921
            accuracy
                                                    1.00
                                                              12526
                             1.00
                                       1.00
                                                    1.00
                                                            12526
           macro avg
```

Model Validation and Evaluation Report:

1.00

1.00

1.00

12526

weighted avg





Model	Classification Report	Accuracy	Confusion Matrix
Decision tree classifier	Tlansification Report: precision recall flucture separt # 1.00 1.00 1.00 1005 1 1.00 1.00 1.00 1001 accuracy 1.00 1.00 1.00 1002 excuracy 1.00 1.00 1.00 1002 excuracy 1.00 1.00 1.00 1002 excuracy 1.00 1.00 1.00 1002	0.99	<pre>contuston_entrix(y_test,y_pres) ref errey([[labs, v],</pre>
Logistic Regression	20 Inglittle Segretion test accomply 4.50150000000000000000000000000000000000	0.96	(3.88) kunfusion_matrix(y_test,lr_pred_test) (a) errey([[343], 188],
K-Nearest Neighbors	### Test descript in: 8.2002314044214378 #### preclaim: resull fl.edom support ### 3.00 3.00 1.00 1.00 1006 % 3.00 3.00 1.00 0.00 ###############################	0.99	Confusion matrix() test,knn ared test() array(([3599, 6],