Model Optimization and Tuning Phase Template

Date	11-07-2024
Team ID	740047
Project Title	SMOKE DETECTION USING IOT DATASET
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (6 Marks):

Model	Tuned Hyperparameters	Optimal Values
Model 1	------	----
Model 2	-----	-----
----	-----	-----

Performance Metrics Comparison Report (2 Marks):

Model	Optimized Metric

Decision Tree Classifier	**Model Building**
	
	``python
	from sklearn.tree import DecisionTreeClassifier
	from sklearn.metrics import accuracy_score,classification_report
	clf = DecisionTreeClassifier()
	``
	
	``python
	# 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
	
	           0 1.00 1.00 1.00 3605
	           1 1.00 1.00 1.00 8921
	
	    accuracy 1.00 12526
	   macro avg 1.00 1.00 1.00 12526
	weighted avg 1.00 1.00 1.00 12526
	``

Logistic Regression
 	
K-Nearest Neighbors
	``python
	from sklearn.linear_model import LogisticRegression
	from sklearn.metrics import accuracy_score,classification_report
	``
	
	``python
	#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 0.94 0.95 0.95 3605
	           1 0.98 0.98 0.98 8921
	
	    accuracy 0.97 12526
	   macro avg 0.96 0.96 0.96 12526
	weighted avg 0.97 0.97 0.97 12526
	``
	
	KNN classifier
	
	``python
	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
	
	           0 1.00 1.00 1.00 3605
	           1 1.00 1.00 1.00 8921
	
	    accuracy 1.00 12526
	   macro avg 1.00 1.00 1.00 12526
	weighted avg 1.00 1.00 1.00 12526
	``
	
	``python
	confusion_matrix(y_test,knn_pred_test)
	``
	
	``
	array([[3599, 6],
	       [3, 8918]])
	``

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
K-Nearest Neighbors	KNN was selected as the final model for its high accuracy and exact prediction of the target outcome and it can be able to handle complex data.