Model Development Phase Template

Date	15 March 2024
Team ID	740139
Project Title	Acoustic Fire Extinguishing prediction
Maximum Marks	6 Marks

Model Selection Report

In the forthcoming Model Selection Report, various models will be outlined, detailing their descriptions, hyperparameters, and performance metrics, including Accuracy or F1 Score. This comprehensive report will provide insights into the chosen models and their effectiveness.

Model Selection Report:

Model	Description	Hyper parameters	Performance Metric (e.g., Accuracy, F1 Score)
K Nearest Neighbors Model	A variable is created with name knn which has KNeighborsClassifier() algorithm initialised in it. The knn model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance.	-	Training accuracy:94 Testing accuracy:91
SVM Model	A variable is created with name Svm which has SVC() algorithm initialised in it. The svm model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance.	-	Training accuracy:88 Testing accuracy:89

Naïve Bayes	A variable is created with name gnb which has GaussianNB() algorithm initialised in it. The gnb model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance	-	Training accuracy:86 Testing accuracy:87
Logistic Regression	A variable is created with name lr which has LogisticRegression() algorithm initialised in it. The lr model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance	-	Training accuracy:87 Testing accuracy:87
Decision Tree Model	A variable is created with name dt which has DecisionTreeClassifier() algorithm initialised in it with a parameter max_depth set to 11. The dt model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance.	-	Training accuracy:98 Testing accuracy:94
Random Forest Model	Random Forest Classifier is a Bagging model which utilises multiple decision trees and takes their aggregate to give a prediction. A variable is created with name rf which has RandomForestClassifier() algorithm initialised in it with a parameter max_depth set to 11. The rf model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance.	-	Training accuracy:98 Testing accuracy:95
Gradient Boosting Model	Random Forest Classifier is a Bagging model which utilises multiple decision trees and takes their aggregate to give a prediction. A variable is created with name rf which has RandomForestClassifier() algorithm initialised in it with a parameter max_depth set to 11. The rf model is trained using the .fit() function. The model is trained on the X_train and y_train data that is the training features and training target variables. This model is then given to the model_evaluation function to check its performance.	-	Training accuracy:95 Testing accuracy:94