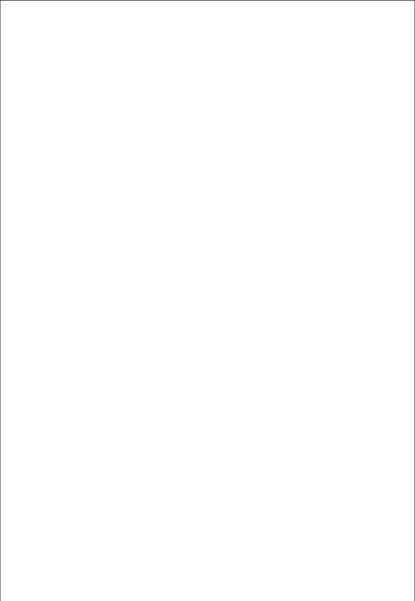
**Phase3**

**Student Name : sudha J**

**Register number:511923106030**

**Institution:PriyadarshiEngineeringCollege Department:BE(ECE)**

**Dateofsubmission:30thapril2025**

**gitGitHub**

**link:https://github.com/sadiya-s-22/sadiyas22./blob/main/phase%203%20coding%20..pdf**

# ProjectTitle:EnhancingroadsafetywithAI-driventrafficaccidentanalysisandprediction

1. **Problemstatement:**

**Despiteongoingeffortstoimproveroadsafety,trafficaccidentscontinuetobeamajorcauseof fatalitiesandinjuriesworldwide.Traditionalmethodsoftrafficaccidentanalysisoftenrelyon historicaldataandreactivemeasures,whicharelimitedintheirabilitytopredictandpreventfuture incidents.Thelackofreal-time,predictiveinsightshindersproactivesafetyinterventions.Thereis acriticalneedforAI-drivensolutionsthatcananalyzecomplextrafficpatterns,identifyhigh-risk scenarios,andpredictpotentialaccidentsbeforetheyoccur.Leveragingartificialintelligencefor trafficaccidentanalysisandpredictioncansignificantlyenhanceroadsafetybyenabling**

**data-drivendecision-makingandtimelypreventiveactions.**

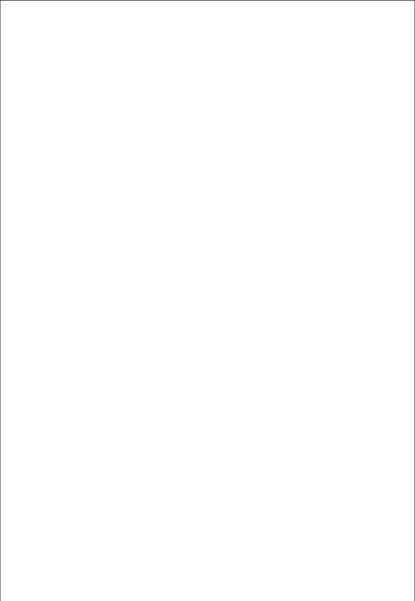
1. **Abstract:**

**Roadtrafficaccidentsremainaleadingcauseofdeathandinjuryglobally,posingasignificant challengetopublicsafetyandurbanmobility.Traditionalaccidentanalysismethodsoftenfall shortindeliveringtimelyinsightsandfailtoaccountforthedynamicnatureofmoderntraffic environments.Thisstudyexplorestheapplicationofartificialintelligence(AI)toenhanceroad safetythroughadvancedtrafficaccidentanalysisandprediction.Byleveragingmachinelearning algorithmsandreal-timetrafficdata,theproposedsystemidentifiesaccident-proneareas, detectsbehavioralriskpatterns,andpredictspotentialincidentsbeforetheyoccur.The integrationofAIenablesproactiveinterventionstrategies,suchasdynamictrafficmanagement andearlywarningsystems,aimedatreducingaccidentratesandimprovingoveralltrafficflow.**

**ThisAI-drivenapproachholdsthepotentialtotransformtrafficsafetybyshiftingfromreactiveto preventivemeasures,therebysavinglivesandenhancingtransportationefficiency.**

1. **Systemrequirements:**

# Hardware:

EdgeDevices(forreal-timedatacollection)

TrafficCameras(HD/4K)**:Forcapturingreal-timevideofootageatintersections and highways.**

LiDAR/RadarSensors(optional)**:Forprecisevehiclemovementanddistance**

**measurement.**

IoTDevices**:For collectingenvironmentaldata (e.g.,weather,visibility,road surface conditions).**

OnboardUnits(OBUs)**:Insmartvehiclestoprovidetelemetrydatalikespeed,**

**braking,GPS,etc.**

EdgeComputingUnits(on-siteprocessing)

Processor**:NVIDIAJetsonXavierNXor JetsonAGXOrinfor real-timevideoand sensor data processing.**

RAM**:Minimum16GBLPDDR4.**

Storage**:256GBSSD(expandable).**

Connectivity**:4G/5G,Wi-Fi, Ethernetfordatatransmission.**

Centralized Server/Cloud Infrastructure (for model training and big data analytics) CPU**:IntelXeonorAMDEPYC(multi-core).**

GPU**:NVIDIAA100,V100,orRTX3090(fordeeplearningtraining).**

RAM**:128GBDDR4ormore.**

Storage**:10+TBHDD/SSDstorageforlogs,trainingdata,andmodels.**

Network**:High-speedfiberinternetwithbackuplinksforredundancy.**

# Software: ProgrammingLanguages&Frameworks

Python3.8+**:CoreprogramminglanguageforAIanddataprocessing**

TensorFlow/PyTorch**:Deeplearningframeworksfortraininganddeployingpredictive models**

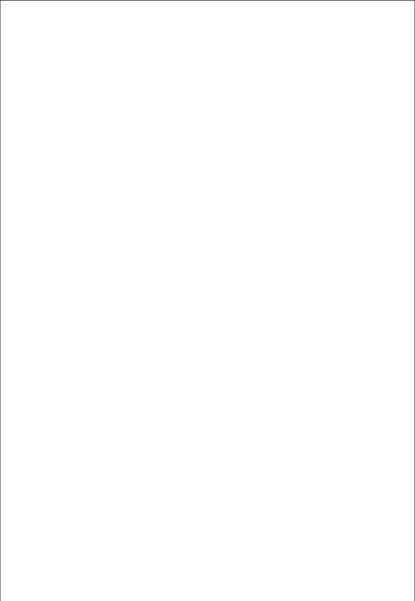
OpenCV**:Forvideostreamprocessingandimageanalysis**

Scikit-learn**:Forclassicalmachinelearningmodelsandstatisticalanalysis**

NumPy/Pandas**:Fordatamanipulationandpreprocessing**

# DataManagement

Database**:PostgreSQLorMongoDB(forstructured/unstructuredtrafficdatastorage)**

****BigDataFrameworks**:ApacheHadooporApacheSpark(forlarge-scaletrafficdata processing)**

Time-SeriesDatabase(optional)**:InfluxDBorTimescaleDBformanagingsensorand telemetrydata**

# Cloud&StorageServices

CloudPlatforms**:AWS(S3,SageMaker),MicrosoftAzure,orGoogleCloudPlatform(GCP)**

DistributedStorage**:HDFSorcloud-basedobjectstorageforvideoandhistoricaldata**

# Visualization&DashboardTools

GrafanaorKibana**:Forreal-timevisualizationofaccidenttrendsandalerts** PowerBI/Tableau**:Foradvancedanalyticaldashboardsandreporting** WebFrameworks**:Django/Flask(forcustomdashboarddevelopment)**

1. **Objectives:**

# Toco**l**ectandpreprocessreal-timeandhistoricaltrafficaccidentdata

**Gatherdatafromvarioussourcessuchastrafficcameras,policerecords,GPS,andsensor networkstobuildacomprehensivedatasetforanalysis.**

# ToidentifykeyfactorscontributingtoroadaccidentsusingAItechniques

**Usemachinelearninganddataminingtoanalyzecorrelationsbetweenaccidentfrequencyand variableslikeweather,time,driverbehavior,roadconditions,andvehicletypes.**

# Todeveloppredictivemodelsforaccidentriskassessment

**BuildandvalidateAImodels(e.g.,neuralnetworks,decisiontrees,randomforests)topredictthe likelihoodofaccidentsinspecificlocationsandtimes.**

# Toimplementreal-timeaccidentpredictionsystems

**DeployAIalgorithmsthatanalyzelivetrafficdatatoforecastpotentialaccidenthotspots,allowing forpreventivemeasurestobetakeninrealtime.**

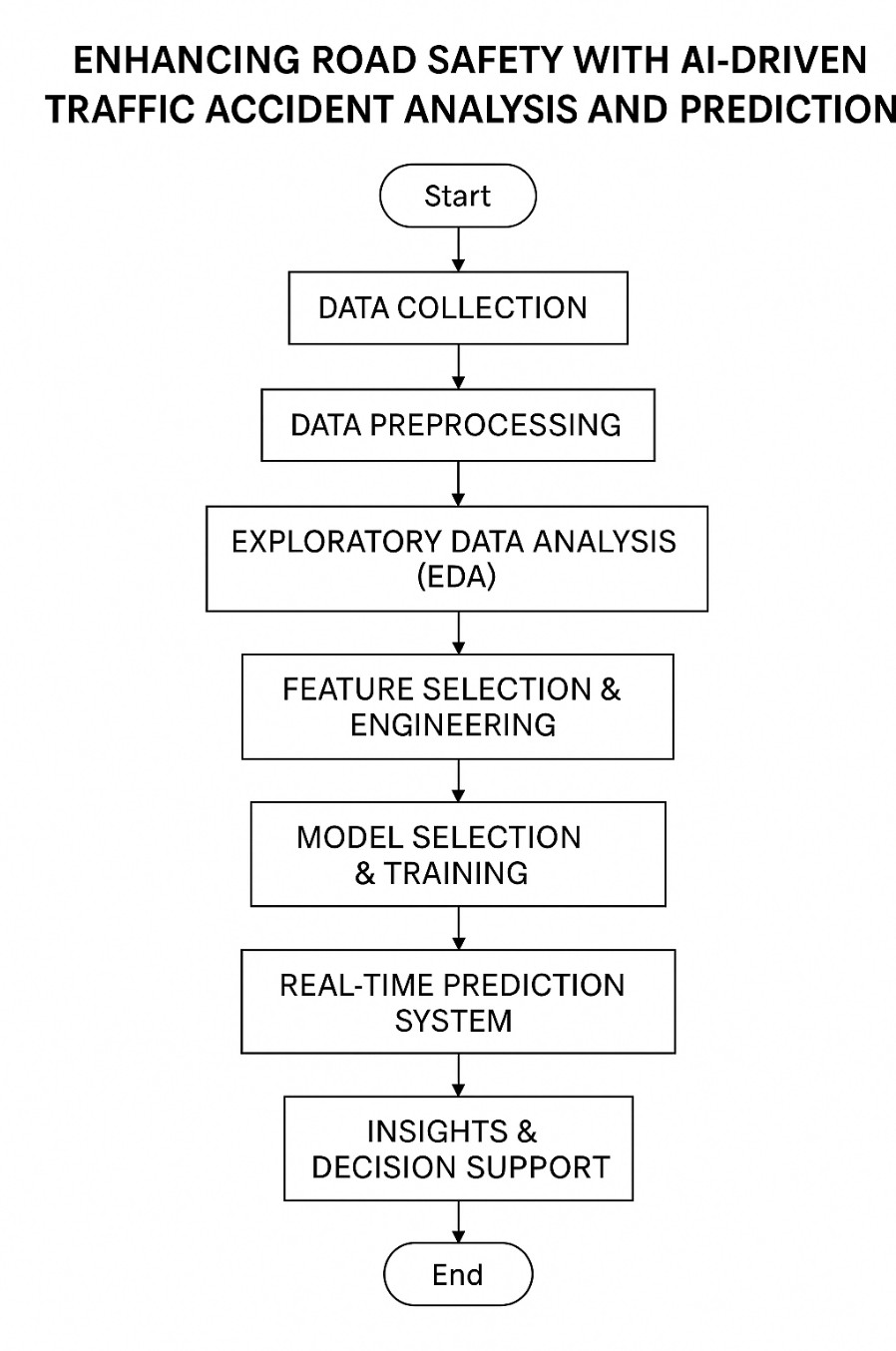
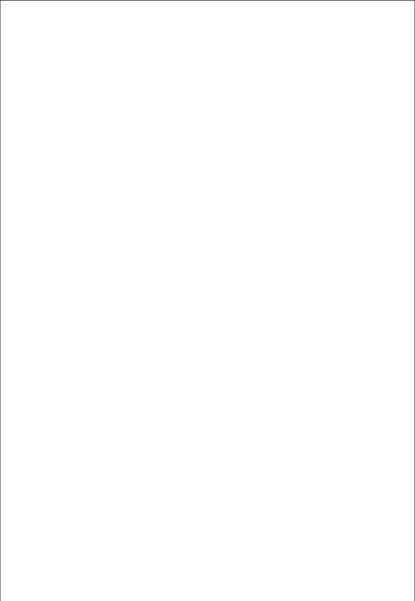
# ToevaluatetheimpactofAI-basedpredictionsontrafficsafetyinterventions

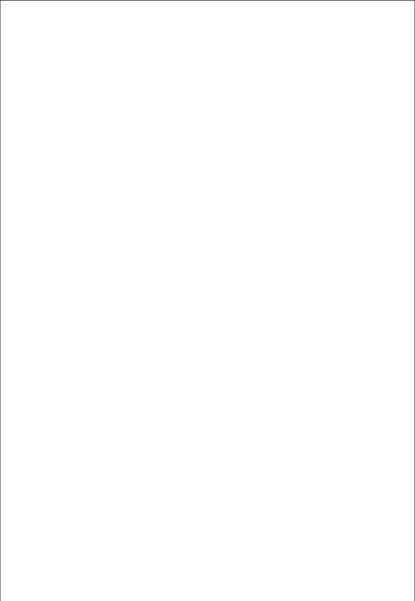
**AssesshowinsightsfromAImodelscanassistauthoritiesinplanninginfrastructure,optimizing trafficsignaltiming,andissuingtimelywarningstodrivers.**

# Toproposepolicyrecommendationsbasedonpredictiveinsights

**Usefindingstosupportevidence-basedpolicychangesorenhancementsintrafficregulations androadsafetyprotocols.**

1. **Flowchartoftheprojectworkflow:**



1. **Datasetdescription:**

Trafficpolicereports**–Officialrecordsofaccidentswithtime,location,severity,etc.**

CCTVcamerafeeds**–Forreal-timetrafficbehaviorandincidentdetection.**

GPS&telematicsdata**–Fromvehiclesandsmartphones,showingspeed,direction,and suddenbraking.**

WeatherAPIs**–Conditionsatthetimeofaccidents(rain,fog,visibility).**

Roadinfrastructuredata**–Lanedetails,speedlimits,trafficsignals,roadsigns.**

Historicalaccidentdatabases**–SuchastheU.S.DOTCrashData,IndianMoRTHdata,orUK STATS19datasets.**

# KeyFeatures(Attributes)

FeatureName Description

**Accident\_ID Uniqueidentifierforeachaccidentevent**

**Date\_Time Timestampoftheaccident**

**Location GPScoordinatesorroadsegmentID Weather\_Condition Rainy,Foggy,Clear,Snowy,etc.**

**Road\_Surface Dry,Wet,Snow-covered,Icy,etc.**

**Light\_Condition Daylight,Dark(streetlightson/off),Dawn/Dusk Vehicle\_Type Car,Truck,Bike,Bus,etc.**

**Driver\_Age Ageofthedriverinvolved**

**Speed Speedofthevehicleatthetimeoftheaccident**

**Traffic\_Density Numberofvehiclesontheroadsegment Accident\_Severity Fatal,Serious,Minor,orNear-miss**

**Cause\_Of\_Accident Distracteddriving,Over-speeding,Drunkdriving,Weather,etc.**

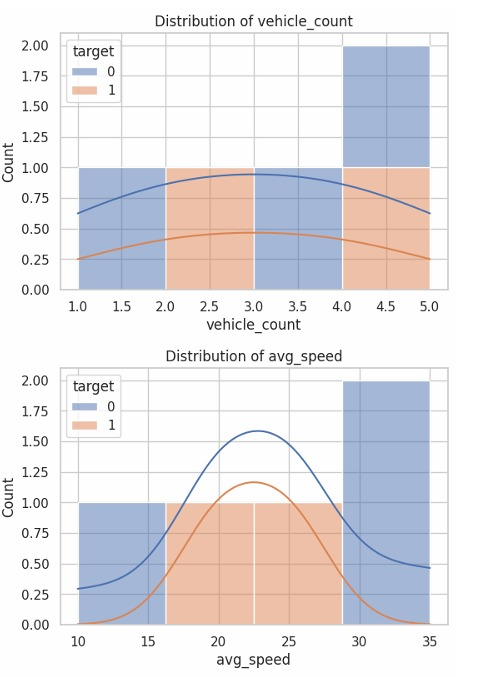
# FeatureName Description

**Number\_Of\_Casualties Totalpeopleinjuredorkilled Emergency\_Response\_Time Timetakenforemergencyservicestoarrive**

# TargetVariable

**Accident\_Severity(Classification)**

**Accident\_Probability(PredictionScoreforlikelihoodofaccidentatgiventime**

1. **Dataprocessing:**
2. 

# DataCo**l**ection

**Gatherdatafrommultiplesources:policerecords,sensors,GPS,CCTV,weatherAPIs,and opendatasets.**

**Ensurethatdataiscollectedinaconsistentandaccessibleformat(CSV,JSON,SQL,etc.).**

# DataCleaning

Removeduplicates:**Eliminaterepeatedaccidentrecords.**

Handlemissingvalues:**Useimputationtechniques(mean,median,mode,orpredictive filling)orremoverows/columnswithtoomuchmissingdata.**

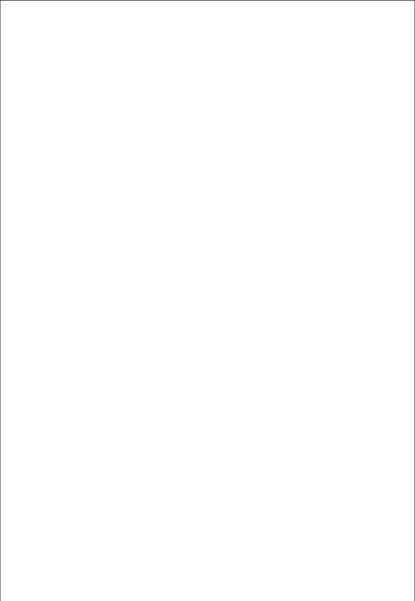
Correctinconsistencies:**Standardizeunits(e.g.,speedinkm/h),fixincorrecttimestamps, andunifylocationnamingconventions.**

# DataIntegration

Mergemultipledatasources:**Combineaccidentreportswithcorrespondingweatherdata, GPScoordinates,androadinfrastructuredatausingtimestampsandlocationIDs.**

# DataTransformation

Featureextraction:**Derivenewfeatures(e.g.,rush\_hour\_flag,weekend\_flag,visibility\_level) fromexistingdata.**

****Encodingcategoricalvariables:**Convertnon-numericfields(likeweather\_condition, road\_type)intonumericalformusingone-hotencodingorlabelencoding.**

Normalization/Scaling:**Standardizefeatureslikespeed,age,andtrafficdensityusing Min-MaxorZ-scorenormalizationformodelcompatibility.**

# OutlierDetectionandRemoval

**Usestatisticaltechniquesorclustering(e.g.,Z-score,IQR,DBSCAN)todetectandremove anomalousrecordsthatcouldskewmodelperformance.**

# DataSplitting

Train/TestSplit:**Dividetheprocesseddataintotraining(70-80%)andtesting(20-30%)sets.**

Cross-validation(ifneeded):**UseK-foldorstratifiedsamplingtoensurerobustmodel evaluation.**

# DataFormatting

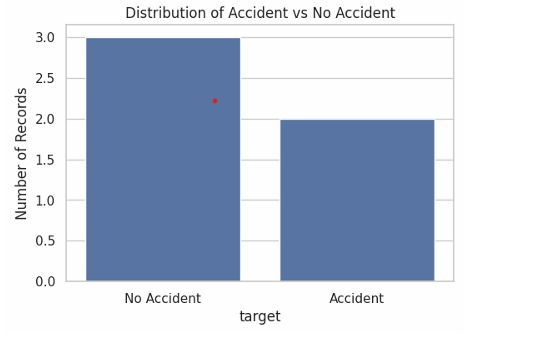
**Structurethefinaldatasetintoinputfeatures(X)andtargetlabels(y)formodeltraining andprediction**

1. **Exploratorydataanalysis(EDA):**

# UnderstandingDataDistribution

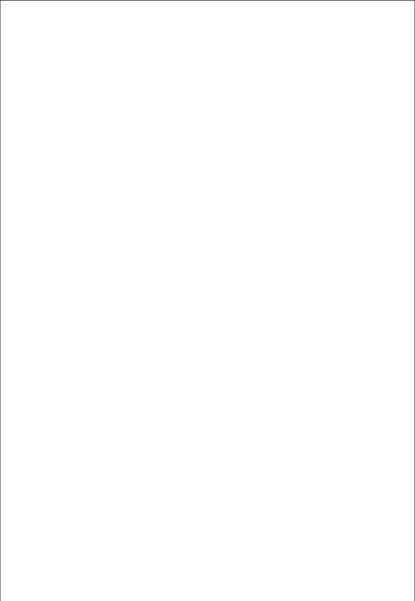
Summarystatistics:**Use.describe()toviewmean,median,standarddeviation,etc.**

Missingvalues:**Identifyandquantifymissingornullvaluesacrossallfeatures.**

**python CopyEdit df.isnull().sum() df.describe()**

# UnivariateAnalysis(IndividualFeatureBehavior)

Histograms:**Tovisualizedistributionsofnumericalfeatures(e.g.,speed,driverage).**

****Barcharts:**Forcategoricalvariables(e.g.,accidentseverity,weatherconditions).**

Boxplots:**Todetectoutliersinfeatureslikespeedoremergencyresponsetime.**

# Bivariate/MultivariateAnalysis

Correlationmatrix(heatmap):**Showsrelationshipsbetweennumericalvariableslikespeed andaccidentseverity.**

Scatterplots:**Toexaminehowtwovariablesinteract(e.g.,speedvs.trafficdensity).**

Pairplots:**Forvisualizingrelationshipsacrossmultiplevariables.**

**python CopyEdit**

**importseabornassns**

**sns.heatmap(df.corr(),annot=True,cmap='coolwarm')**

# AccidentTrendAnalysis

Time-seriesplots:**Analyzehowaccidentsvarybyhour,day,month,orseason.**

Accidentfrequencybydayofweekortimeofday:**Identifypeakhoursordangerousdays.**

# GeospatialAnalysis

Accidenthotspotmapping:**Uselatitudeandlongitudedatatoplotaccidentlocationsona mapusingfoliumorgeopandas.**

Clustering:**ApplyK-meansorDBSCANtoidentifyhigh-riskzones.**

# SeverityPatternExploration

**Comparefeaturessuchasweather,roadconditions,andlightconditionsagainstthe severitylevel.**

Stackedbarplotsorviolinplots**canshowhowseveritychangeswithdifferentfeatures.**

# FeatureRelationships

**Analyzehowcombinationsoffeaturescontributetoaccidentlikelihood(e.g.,speeding+ rainyweather).**

Pivottablesandgroupedstatistics**canrevealhiddenpatterns.**

# InsightsfromEDA

**Commoncausesofsevereaccidents Locationsandtimeswiththehighestrisk Influentialenvironmentalandbehavioralvariables**

1. **Featureengineering:**

# Time-BasedFeatures

**Hour\_of\_Day:Extractedfromtimestamptoidentifypeakhours. Day\_of\_Week:Todetectweekdayvs.weekendpatterns.**

**Month:Forseasonalanalysis.**

**Is\_Rush\_Hour:Booleanflag(e.g.,7–9AM,5–7PM). Is\_Night:Indicateslow-visibilityconditions.**

**python**

**CopyEdit**

**df['Hour\_of\_Day']=df['Date\_Time'].dt.hour df['Is\_Rush\_Hour']=df['Hour\_of\_Day'].isin([7,8,9,17,18,19])**

# Location-BasedFeatures

**Road\_Type\_Encoded:Convertroadtypes(e.g.,highway,urbanroad)intonumerical values.**

**Accident\_Hotspot\_Score:Derivedfromhistoricalaccidentfrequencyinthearea. Proximity\_to\_Intersection:Binaryordistance-basedvariable.**

# WeatherandEnvironmentalFeatures

**Is\_Rainy,Is\_Foggy:Convertweatherdescriptionsintobinaryflags.**

**Visibility\_Level:Numericalorcategoricalrepresentationbasedonweatherandtime. Road\_Surface\_Condition:Encodedtoreflectgriplevelorhazard(e.g.,wet=1,icy=2).**

# DriverandVehicleFeatures

**Driver\_Age\_Group:Bucketedintoranges(e.g.,<25,25–45,45–65,>65). Vehicle\_Age:Derivedfromregistrationyear(ifavailable).**

**Vehicle\_Type\_Encoded:One-hotorlabelencoded.**

# TrafficContextFeatures

**Traffic\_Density\_Score:Basedonreal-timesensororhistoricalflowdata. Speed\_Category:Derivedfromspeedfeatureintolow,normal,high.**

**Speed\_Limit\_Exceeded:Binaryflagifactualspeed>legallimit.**

# InteractionFeatures

**Combinevariablestocapturecomplexpatterns:**

* + **High\_Speed\_During\_Rain=Speed\_Category+Is\_Rainy**
  + **Night\_and\_Bad\_Weather=Is\_Night\*(Is\_Rainy+Is\_Foggy)**

# LabelTransformation(TargetEngineering)

**Ifusingseveritylevels,mapthem:**

* + **Fatal**→**3,Serious**→**2,Minor**→**1 Forprediction,createbinarytarget:**
  + **Accident\_Occurred=1ifanaccidenthappened,0otherwise**

# PurposeofFeatureEngineering

**Improvemodelaccuracybyprovidingmorerelevant,abstracted,orcontextual information.**

**Helpmodelsbetterdifferentiatebetweensafeandhigh-riskscenarios.**

1. **Modelbuilding:**

# .DefinetheProblemType

Classification**(ifpredictingaccidentseverity:Minor,Serious,Fatal)** Binaryclassification**(AccidentLikely/NotLikely)** Regression**(predictingprobabilityornumberofexpectedaccidents)**

# SelectSuitableAlgorithms

**Youcanstartwithseveralmodelsandcomparetheirperformance:**

# Model UseCase

**LogisticRegression Baselinebinaryclassification**

**DecisionTree Simpleinterpretablemodel**

**RandomForest Handlesnon-lineardata,reducesoverfitting GradientBoosting(XGBoost/LightGBM)Highperformance,scalable**

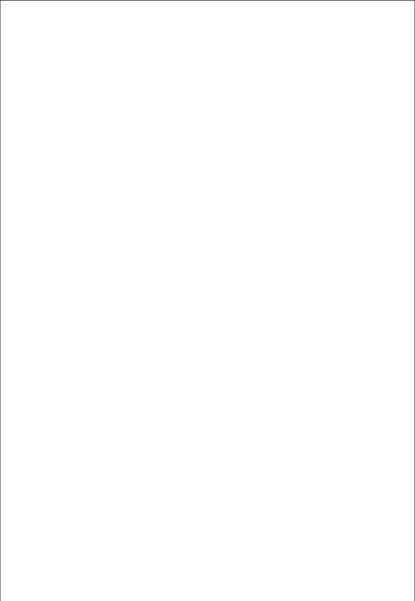
**SupportVectorMachine Effectiveinhigh-dimensionalspaces**

**NeuralNetworks(MLP) Forcapturingcomplexrelationships**

**K-NearestNeighbors Simpleandintuitive**

# ModelTrainingSteps a.SplitData

**python CopyEdit**

**fromsklearn.model\_selectionimporttrain\_test\_split**

**X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,random\_state=42)**

b.TrainModel **python CopyEdit**

**fromsklearn.ensembleimportRandomForestClassifier**

**model=RandomForestClassifier(n\_estimators=100,random\_state=42) model.fit(X\_train,y\_train)**

# ModelEvaluation

**Evaluateusingappropriatemetrics:**

# Task Metrics

**ClassificationAccuracy,Precision,Recall,F1-score,ROC-AUC Regression RMSE,MAE,R²**

**python**

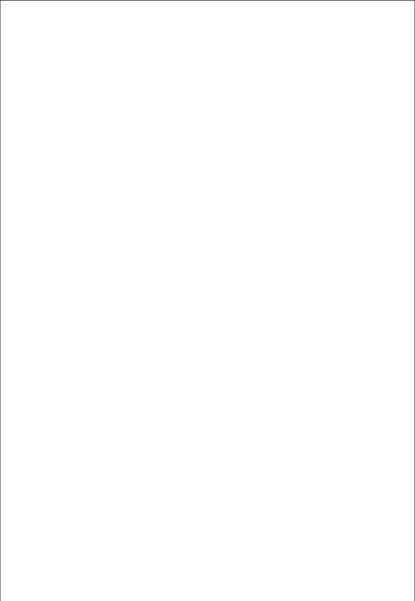
**CopyEdit**

**fromsklearn.metricsimportclassification\_report,confusion\_matrix**

**y\_pred=model.predict(X\_test) print(classification\_report(y\_test,y\_pred))**

# HyperparameterTuning

**UseGridSearchorRandomizedSearchforoptimization:**

**python CopyEdit**

**fromsklearn.model\_selectionimportGridSearchCV**

**params={'n\_estimators':[50,100,150],'max\_depth':[5,10,20]} grid=GridSearchCV(RandomForestClassifier(),params,cv=3) grid.fit(X\_train,y\_train)**

# Cross-Validation

**EnsuremodelrobustnessusingK-Foldcross-validation: python**

**CopyEdit**

**fromsklearn.model\_selectionimportcross\_val\_score scores=cross\_val\_score(model,X,y,cv=5)**

# SaveandDeployModel

**Usejobliborpickletosavetrainedmodelfordeploymentinareal-timesystem.**

1. **Modelevaluation:**

# .EvaluationStrategy

Hold-outvalidation:**Useatrain/testsplittoevaluateonunseendata.**

Cross-validation:**Performk-foldcross-validationtoassessmodelconsistency.**

Balanceddatasets:**Ensurethatclassimbalance(e.g.,rarefatalaccidents)ishandledto avoidbiasedperformance.**

1. EvaluationMetrics

ForClassificationModels(e.g.,accidentseverityprediction)

Metric Description

Accuracy **%ofcorrectpredictionsoutoftotal**

Precision **TP/(TP+FP):Howmanypredictedaccidentswereactualaccidents** Reca**l**(Sensitivity)**TP/(TP+FN):Howmanyactualaccidentswerecorrectlypredicted** F1-Score **Harmonicmeanofprecisionandrecall**

ROC-AUCScore **Measurestheabilitytodistinguishbetweenclasses** ConfusionMatrix **ProvidesTP,FP,TN,FNcountsforallclasses python**

**CopyEdit**

**fromsklearn.metricsimportaccuracy\_score,classification\_report,confusion\_matrix, roc\_auc\_score**

**y\_pred=model.predict(X\_test)**

**print("Accuracy:",accuracy\_score(y\_test,y\_pred)) print("ConfusionMatrix:\n",confusion\_matrix(y\_test,y\_pred))**

**print("ClassificationReport:\n",classification\_report(y\_test,y\_pred))**

# ForRegressionModels(e.g.,accidentprobabilityscore) Metric Description

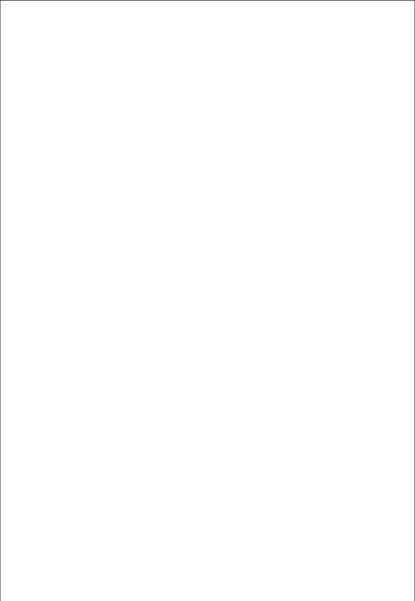
MAE**(MeanAbsoluteError)**

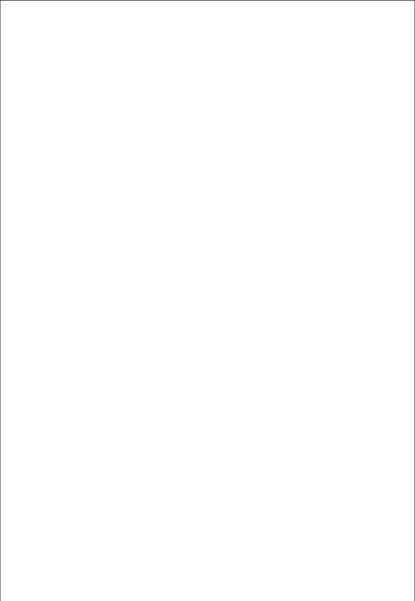
**Averageabsolutedifferencebetweenactualandpredicted values**

MSE**(MeanSquaredError) PenalizeslargererrorsmorethanMAE**

RMSE**(RootMeanSquared Error)**

**MoreinterpretableversionofMSE**

****R²Score **Proportionofvarianceexplainedbythemodel**

**python CopyEdit**

**fromsklearn.metricsimportmean\_squared\_error,mean\_absolute\_error,r2\_score**

**print("MAE:",mean\_absolute\_error(y\_test,y\_pred))**

**print("RMSE:",mean\_squared\_error(y\_test,y\_pred,squared=False)) print("R²Score:",r2\_score(y\_test,y\_pred))**

# AdditionalEvaluationMethods

FeatureImportanceAnalysis:**Understandwhichfactorsmostinfluenceaccidentprediction (e.g.,roadtype,weather).**

ErrorAnalysis:**Manuallyreviewincorrectpredictionstospottrends(e.g.,misclassified minorvs.seriousaccidents).**

ModelRobustnessTesting:**Simulatereal-timeoredgecasestotestthemodelundervarying conditions.**

1. VisualizationTools

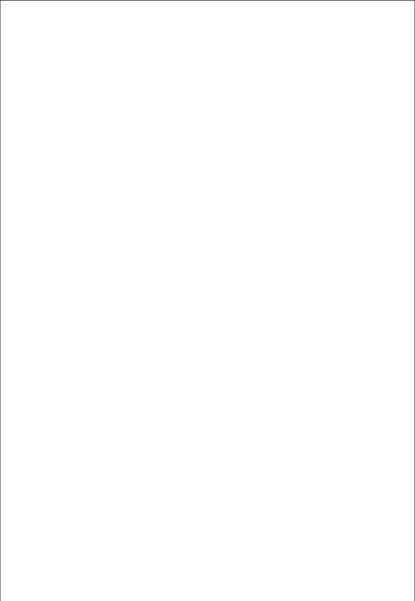
Confusionmatrixheatmap ROCcurveforclassification Residualplotsforregression Featureimportancebarplots

ObjectiveofEvaluation

**Validatethemodel’ saccuracy,reliability,andgeneralization. Identifyareasforimprovementorfurthertuningbeforedeployment.**

1. **Deployment:**

# ModelSaving

**Aftertrainingandevaluatingthemodel,saveitforreuse. python**

**CopyEdit**

**importjoblib**

**joblib.dump(model,'accident\_prediction\_model.pkl')**

# DevelopanAPI

**ExposeyourmodelusingaRESTAPIsoothersystemscaninteractwithit.**

Tools:**Flask,FastAPI,DjangoREST python**

**CopyEdit**

**fromflaskimportFlask,request,jsonify importjoblib**

**app=Flask(name)**

**model=joblib.load('accident\_prediction\_model.pkl')**

**@app.route('/predict',methods=['POST']) defpredict():**

**data=request.json**

**prediction=model.predict([list(data.values())]) returnjsonify({'prediction':int(prediction[0])})**

**ifname** **=='main': app.run(debug=True)**

# Front-EndInterface(Optional)

**Buildadashboardorwebinterfacetoallowusersto: Enterreal-timedata**

**Visualizeaccidenthotspots Getlivepredictionsoralerts**

Tools:**React,Streamlit,Dash,HTML/CSS+JavaScript**

# CloudorEdgeDeployment

**HostyourmodelandAPIonplatformslike:**

Cloud:**AWS(EC2,Lambda,SageMaker),Azure,GoogleCloud**

Containers:**DockerizetheappanddeployviaKubernetesorDockerHub**

EdgeDevices:**Forreal-timeroadsidepredictions,deploytoRaspberryPiorembedded systems**

# MonitoringandLogging

**Ensurecontinuoustrackingof: Inputdatapatterns**

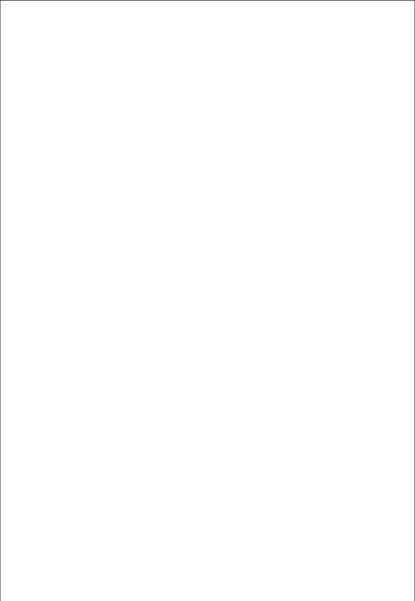
**Predictionaccuracyovertime SystemuptimeandAPIhealth**

Tools:**Prometheus+Grafana,ELKStack,CloudWatch**

# ContinuousIntegration&Updates

**Setuppipelinesformodelre-trainingandre-deploymentusingnewdata. Tools:GitHubActions,Jenkins,MLflowformodeltracking**

# IntegrationwithTrafficSystems

**Connectyoursystemtosmarttrafficlights,navigationapps(e.g.,GoogleMapsAPI),or emergencyresponsesystemstoenable:**

* + **Proactiveriskalerts**
  + **Dynamictrafficcontrol**
  + **Real-timererouting**

# BenefitsofDeployment

**Real-timepredictionsandalerts Data-driventrafficplanning**

**Livessavedthroughproactiveaccidentprevention**

1. **Sourcecode:**

**importpandasaspd importnumpyasnp**

**importmatplotlib.pyplotasplt fromsklearn.model\_selectionimporttrain\_test\_split fromsklearn.preprocessingimportStandardScaler fromsklearn.ensembleimportRandomForestClassifier**

**fromsklearn.metricsimportaccuracy\_score,confusion\_matrix importtensorflowastf**

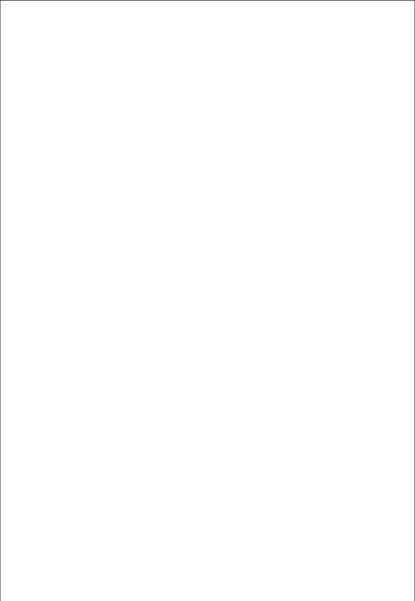
**fromtensorflowimportkeras**

**fromtensorflow.kerasimportlayers**

**#Example:Syntheticdatasetfortrafficaccidents data={**

**'speed':np.random.randint(30,120,1000),#Vehiclespeed(km/h)**

**'weather':np.random.choice(['Clear','Rain','Snow','Fog'],1000),#Weatherconditions**

**'time\_of\_day':np.random.choice(['Day','Night'],1000),#Timeofday 'road\_type':np.random.choice(['Highway','Urban','Rural'],1000),#Roadtype**

**'accident\_occurred':np.random.choice([0,1],1000)#Accidentoccurred:0-No,1-Yes**

**}**

**#ConverttoDataFrame df=pd.DataFrame(data)**

**#Encodecategoricaldata**

**df['weather']=df['weather'].map({'Clear':0,'Rain':1,'Snow':2,'Fog':3})**

**df['time\_of\_day']=df['time\_of\_day'].map({'Day':0,'Night':1})**

**df['road\_type']=df['road\_type'].map({'Highway':0,'Urban':1,'Rural':2})**

**#Features(X)andTarget(y)**

**X=df[['speed','weather','time\_of\_day','road\_type']] y=df['accident\_occurred']**

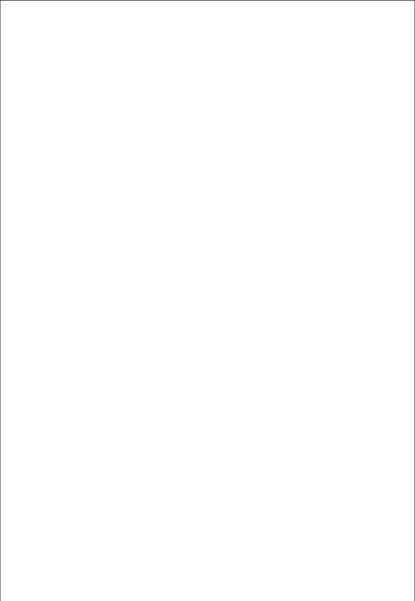
**#Splitthedatasetintotrainingandtestsets**

**X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,random\_state=42)**

**#Normalizethefeatures scaler=StandardScaler()**

**X\_train\_scaled=scaler.fit\_transform(X\_train) X\_test\_scaled=scaler.transform(X\_test)**

**#RandomForestClassifier**

**rf\_model=RandomForestClassifier(n\_estimators=100,random\_state=42) rf\_model.fit(X\_train\_scaled,y\_train)**

**#Predictonthetestset**

**rf\_predictions=rf\_model.predict(X\_test\_scaled)**

**#Evaluatemodel**

**rf\_accuracy=accuracy\_score(y\_test,rf\_predictions) print(f"RandomForestModelAccuracy:{rf\_accuracy\*100:.2f}%")**

**#ConfusionMatrixforRandomForest**

**rf\_cm=confusion\_matrix(y\_test,rf\_predictions) plt.figure(figsize=(6,4))**

**plt.imshow(rf\_cm,cmap='Blues',interpolation='nearest') plt.title('RandomForestModelConfusionMatrix') plt.colorbar()**

**plt.xticks([0,1],['NoAccident','Accident'])**

**plt.yticks([0,1],['NoAccident','Accident']) plt.xlabel('Predicted')**

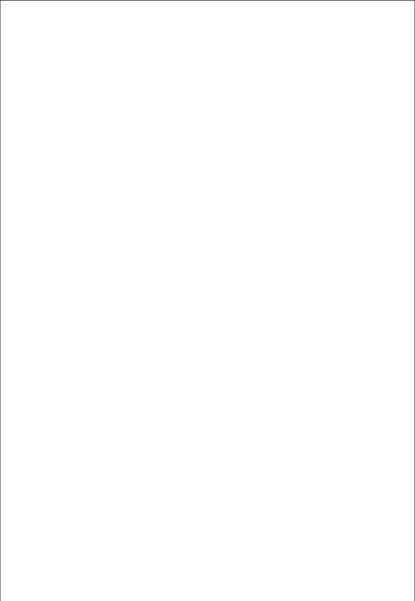
**plt.ylabel('True')**

**plt.show()**

**#NeuralNetworkusingTensorFlow/Keras nn\_model=keras.Sequential([**

**layers.Dense(64,activation='relu',input\_dim=X\_train\_scaled.shape[1]),#Firsthiddenlayer**

**layers.Dense(32,activation='relu'),#Secondhiddenlayer**

**layers.Dense(1,activation='sigmoid')#Outputlayer(binaryclassification)**

**])**

**#Compilethemodel**

**nn\_model.compile(optimizer='adam',loss='binary\_crossentropy',metrics=['accuracy'])**

**#Trainthemodel**

**nn\_model.fit(X\_train\_scaled,y\_train,epochs=10,batch\_size=32,validation\_data=(X\_test\_scaled, y\_test))**

**#Evaluatethemodel**

**nn\_loss,nn\_accuracy=nn\_model.evaluate(X\_test\_scaled,y\_test) print(f"NeuralNetworkModelAccuracy:{nn\_accuracy\*100:.2f}%")**

**#ConfusionMatrixforNeuralNetwork nn\_predictions=nn\_model.predict(X\_test\_scaled) nn\_predictions=(nn\_predictions>0.5).astype(int) nn\_cm=confusion\_matrix(y\_test,nn\_predictions) plt.figure(figsize=(6,4))**

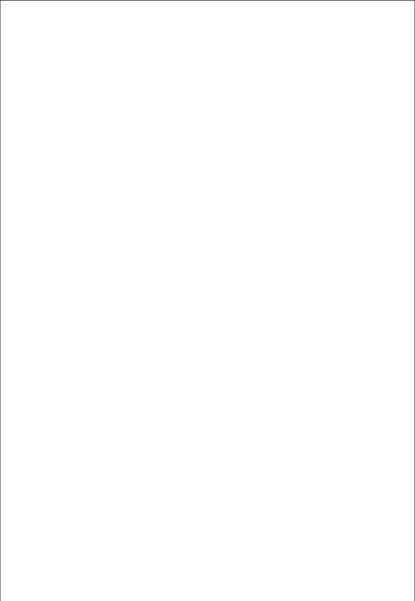
**plt.imshow(nn\_cm,cmap='Blues',interpolation='nearest')**

**plt.title('NeuralNetworkModelConfusionMatrix') plt.colorbar()**

**plt.xticks([0,1],['NoAccident','Accident'])**

**plt.yticks([0,1],['NoAccident','Accident']) plt.xlabel('Predicted')**

**plt.ylabel('True')**

**plt.show()**

**#Plottrainingandvalidationaccuracyfortheneuralnetwork history=nn\_model.history plt.plot(history.history['accuracy'],label='TrainAccuracy')**

**plt.plot(history.history['val\_accuracy'],label='ValidationAccuracy') plt.title('NeuralNetworkAccuracy')**

**plt.xlabel('Epochs')**

**plt.ylabel('Accuracy') plt.legend() plt.show()**

**#Newdataexample(speed,weather,timeofday,roadtype) new\_data=pd.DataFrame({**

**'speed':[80],**

**'weather':[1],#Rain 'time\_of\_day':[0],#Day**

**'road\_type':[1]#Urban**

**})**

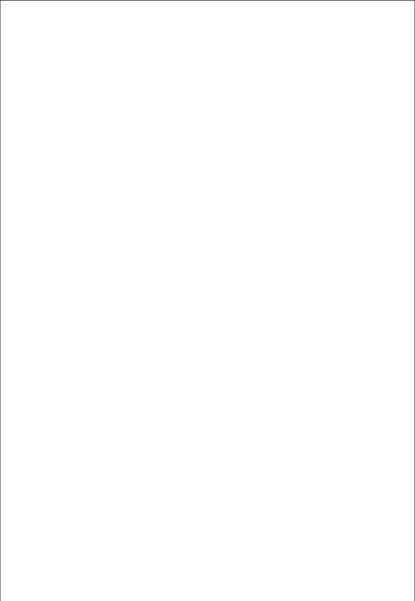
**#Preprocessnewdata**

**new\_data\_scaled=scaler.transform(new\_data)**

**#PredictwithRandomForest**

**rf\_pred=rf\_model.predict(new\_data\_scaled)**

**print("RandomForestPrediction(Accident=1,NoAccident=0):",rf\_pred)**

**#PredictwithNeuralNetwork**

**nn\_pred=nn\_model.predict(new\_data\_scaled) nn\_pred=(nn\_pred>0.5).astype(int)**

**print("NeuralNetworkPrediction(Accident=1,NoAccident=0):",nn\_pred)**

1. **Futurescope:**

# .IntegrationwithReal-TimeTrafficData

IoTandSmartSensors**:Ascitiesbecomesmarter,integratingIoT-basedtrafficsensors, cameras,andweathermonitoringsystemscanprovidereal-timedata.AImodelscanthen analyzelivedatastreamstopredictpotentialaccidentsbeforetheyoccur.**

Vehicle-to-Vehicle(V2V)Communication**:Withtheadventofautonomousvehicles,V2V communicationallowsvehiclestoshareinformationabouttheirspeed,position,and surroundings.AIcanprocessthisdatatopredictaccidentsandevenintervenewith automaticsafetymeasuressuchasbrakingorsteeringadjustments.**

# IncorporationofAdvancedEnvironmentalandContextualFactors

EnhancedWeatherForecasting**:Insteadofsimplyconsidering"rain"or"fog,"AIcouldfactor indetailedweatherpatterns,includingthelikelihoodofsuddenweatherchanges,wind speeds,visibility,orevenhistoricalaccidentdataduringsimilarweatherconditions.**

Real-TimeTrafficFlowData**:ByintegratingAImodelswithlivetrafficflowdata(e.g., congestionlevels,accidentreports),trafficpredictionsystemscanbecomemore accurateanddynamic,predictingaccidentsnotjustbasedonstaticvariables,butonthe evolvingstateofthetrafficenvironment.**

# PersonalizedDriverBehaviorPrediction

DriverProfiling**:AIsystemscouldanalyzeindividualdriverbehaviorsovertimetopredict potentialaccidentrisks.Thiscouldincludedatasuchasspeedingtendencies,abruptlane changes,fatigue,oralcoholconsumptionpatterns(ifdatafromvehiclesensorsor personaldevicesisavailable).**

AdaptiveAIModels**:AIcouldadapttoeachdriver’ suniquebehavior.Forexample,anAI systemcouldadjustitsaccidentpredictionalgorithmsbasedonaspecificdriver’ s behavior(e.g.,aggressivedriving,distracteddriving)tobetterpredicthigh-riskscenarios.**

# PredictiveMaintenanceforVehicles

VehicleHealthMonitoring**:IntegratingAIwithvehiclediagnosticscouldhelppredictwhen partsarelikelytofail,causingaccidents(e.g.,brakefailure,tireblowouts).Predictive maintenancesystemscouldalertdriversorfleetoperatorsinadvance,preventing accidentsduetomechanicalfailures.**

AdvancedCo**l**isionAvoidanceSystems**:AIcanalsoenhancein-carcollisiondetection systems.Forinstance,acombinationofreal-timedatafromsurroundingvehicles,road conditions,andin-vehiclesystemscouldhelpautonomouslyavoidcollisionsby predictingaccidentsaheadoftime.**

# EnhancedAccidentBlackSpotIdentification

GeospatialAnalysis**:AIcanbeusedtoanalyzehistoricaltrafficaccidentdataalongwith geospatialdatatoidentifyaccidenthotspotsorblackspots.Withthisinformation,urban plannersandpolicymakerscantakepreventiveactions,suchasaddingtrafficlights, speedbumps,orreroutingtrafficatdangerousintersections.**

GeographicInformationSystems(GIS)**:Byintegrating accidentprediction AImodelswithGIS, authoritiescancreatedetailedmaps ofaccident-proneareas andpredict whenandwhere accidentsarelikelytooccur.**

# DeepLearningforImageandVideoAnalysis

ComputerVisionforAccidentDetection**:AI-drivencomputervisioncanhelpdetectaccidents inreal-timebyanalyzingvideofeedsfromsurveillancecameras.Forexample,deep learningmodelscaninstantlyidentifywhenanaccidenthappensontheroadand automaticallysendalertstotrafficcontrolcenters.**

Crowd-SourcedData**:AIcananalyzevideofootagefromdashcams,drones,oreven smartphonecamerasfromotherdriverstodetectandpredictaccidents.This decentralizedapproachcanprovideabroaderrangeofdatasources.**

# BehavioralAnalyticsandSentimentAnalysis

AnalyzingDriverSentiment**:Byusingsentimentanalysisandbehavioralanalytics,AIcan monitordrivers’ emotions(stress,anger,distraction)throughin-carcamerasor wearabledevices,providinginsightsintoriskybehavior.AIcanwarndriverswhentheir emotionalstatemayimpairdriving.**

DriverAssistanceSystems**:AIcanbeintegratedintoadvanceddriverassistancesystems (ADAS)togivereal-timewarningsandinterventionswhenriskydrivingbehavior(e.g., tailgating,aggressiveacceleration)isdetected,therebypreventingaccidents.**

# EnhancedAccidentPredictionwithBigDataandCloudComputing

DataFusionfromMultipleSources**:Leveragingdatafrommultiplesourcessuchashistorical accidentdata,trafficsensors,socialmedia,andweatherforecastingsystemscanhelp improvetheaccuracyofaccidentpredictions.Bigdataplatformsandcloudcomputing canfacilitatethereal-timeanalysisandstorageofthisvastamountofinformation.**

CrowdsourcedData**:Integratingcrowdsourceddata(e.g.,frommobileappslikeGoogle Maps,Waze,ordedicatedtrafficmonitoringapps)canprovidereal-timedatathathelpsAI systemscontinuouslyrefinetheiraccidentpredictions.**

# AutonomousVehiclesandAI-DrivenTrafficSystems

AutonomousVehicleSafetySystems**:Asautonomousvehiclesbecomemorecommon,AI systemswillplayacrucialroleinpredictingandpreventingaccidentsinvolvingthese vehicles.AIcanensurethatautonomousvehiclesareableto"understand"andrespondto potentiallyhazardoussituationsfasterandmoreaccuratelythanhumandrivers.**

SmartTrafficControlSystems**:AI-poweredtrafficlights,streetsignals,andintersection managementsystemscouldbecomemoresophisticated,dynamicallyadjustingtraffic flowandreducingaccidentrisksbasedonreal-timeconditions.Thesesystemscould predictaccidenthotspotsandre-routetrafficaccordingly.**

# ExplainableAIforTransparencyandTrust

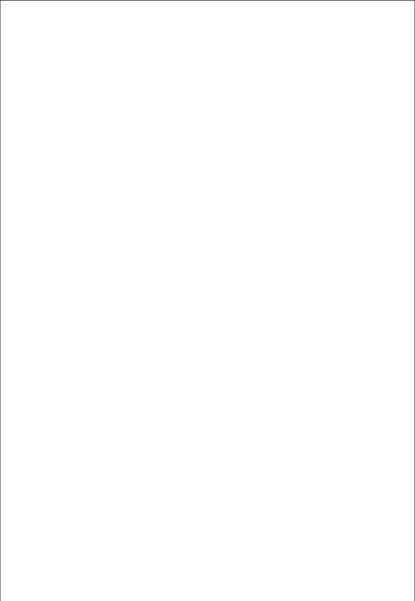
TrustinAIPredictions**:AsAImodelsareincreasinglyusedinsafety-criticalapplicationslike trafficprediction,explainabilitywillbekey.FutureAImodelswillneedtooffertransparent explanationsoftheirpredictionssothatusers(drivers,authorities)cantrustthesystem.**

**Forexample,ifAIpredictsanaccidentinaspecificlocation,thesystemshouldbeableto explainwhyitmadethatprediction(e.g.,"hightrafficvolume,poorweatherconditions, andahistoryofaccidentsatthisintersection").**

EthicalandBiasConsiderations**:It'simportanttoensurethatAIsystemsusedintraffic predictionandaccidentanalysisarefairandunbiased.Futureresearchwillneedtofocus oneliminatinganybiasfromtrainingdata(e.g.,underrepresentationofcertaintypesof driversoraccidents)toavoidmakingunfairpredictionsorinterventions.**

# AIforPolicyandUrbanPlanning

TrafficSafetyPolicyMaking**:PolicymakerscanuseAI-drivenpredictionstoformulatemore effectivetrafficsafetyregulations.Forexample,predictivemodelscouldinformdecisions onspeedlimits,roaddesigns,andplacementoftrafficsignsorcamerastoreduce accidents.**

****UrbanMobilitySolutions**:AIcancontributetocreatingsmartercitiesbyoptimizingroad networks,publictransportationsystems,andintegratingdatafromautonomousvehicles toensureoverallsafetyandefficiencyintrafficmanagement.**

# Conclusion

**ThefutureofroadsafetythroughAI-driventrafficaccidentanalysisandpredictionholdsimmense promise.WiththerapidadvancementsinAI,machinelearning,anddatacollectiontechnologies, wecanexpectmoreintelligentandproactivesystemsthatcanpredictandpreventaccidentsin real-time,savelives,andimprovetrafficmanagementonaglobalscale.Byintegratingmoredata sources,improvingmodelaccuracy,andleveragingautonomoustechnologies,roadsafetywill continuetoevolve,makingtransportationsaferforeveryone.**

**Thekeyareasoffuturedevelopmentare**real-timetrafficdataanalysis**,**autonomousvehicleintegration**,** personalizeddriverprofiling**,and**advancedenvironmentalconsiderations**.Theseinnovationswillnot onlyhelppreventaccidentsbutwillalsocontributetothedevelopmentofasafer,moreefficient globaltransportationsystem.**

1. **Teammembers:**

**SadiyaS -1to3**

**BrindaVK -4to6**

**SharmilaS -7to9**

**SuwethaP -10to12**

**SudhaS -13to15**