TRAFFIC MANAGEMENT USING IoT

# NAME:  AVIKA M

COLLEGE CODE:9530

NAN MUDHALVAN ID:  au953021106011

PHASE-3 PROJECT SUBMISSION

PROJECT TITLE:  TRAFFIC MANAGEMENT SYSTEM

PHASE 3:  DEVELOPEMENT PART 1

# INTRODUCTION:

PROBLEM STATEMENT:

Traffic congestion consists of incremental delay, consumption, pollution emission and stress that result from interference vehicles in traffic stream, particularly as traffic volumes approaches a road’s capacity.

DESCRIPTION:

Traffic management concerns the control, planning, and purchasing of transport services needed to physically move road vehicles.

COMPONENTS REQUIRED:

* ARDUINO UNO
* TRAFFIC SENSOR
* JUMPER WIRES
* BREAD BOARD
* LED

### SOFTWARE REQUIRED:

* Arduino IDE:

Used to connect Arduino UNO and upload the file into them for further processing.

* TINKERCAD:

Used for creating digital circuit and to simulate that.

# SOURCE  CODE

from \_future\_ import absolute\_import

from \_future\_ import print\_function

from sumolib import checkBinary

import os

import sys

import optparse

import subprocess

import random

import traci

import random

import numpy as np

import keras

import h5py

from collections import deque

from keras.layers import Input,Conv2D,Flatten

from keras.models import Model

class DQNAgent:

def \_init\_(self):

self.gamma = 0.95

self.epsilon = 0.1

self.learning\_rate = 0.0002

self.memory = deque(maxlen=200)

self.model = self.\_build\_model()

self.action\_size=2

def \_build\_model(self):

input\_1=Input(shape=(12, 12, 1))

x1 = conv2D(16, (4,4),strides=(2,2),activation='relu')(input\_1)

x1=conv2D(32, (2,2),strides=(1,1), activation='relu')(x1)

x1=Flatten()(x1)

input\_2=Input(shape=(12, 12, 1))

x2=conv2D(16, (4,4),strides=(2,2), activation='relu')(input\_2)

x2=conv2D(32, (2,2),strides=(1,1), activation='relu')(x2)

x2=Flatten()(x2)

input\_3=Input(shape=(2, 1))

x3=Flatten()(input\_3)

x=keras.layers. Concatenate([x1, x2, x3])

x=Dense (128, activation='relu’) (x)

x=Dense (64, activation='relu’) (x)

x=Dense (2, activation='relu’) (x)

model=Model (inputs= [input\_1, input\_2, input\_3], output=[x])

model. compile (optimizer=keras. optimizers. RMSprop (lr=self. learning\_rate), loss='mse')

return model

def remember (self, state, action, reward, next\_state,done):

self.memory. Append((self,state,action,reward,next\_state,done))

def act(self,state):

if np. random. rand () <=self. epsilon:

return random. randrang (self. action-size)

act values=self. model. pretict(state)

return np. argmax (act values [0])

def replay (self, batch\_size):

minibatch=random. Sample (self. memory, batch\_size)

for state, action, reward, next\_state, done in minibatch:

target=reward

if not done:

target= (reward+self. gamma\*np. amax (self. model. pretict(next\_state) [0]))

target\_f=self. model. pretict(state)

target\_f[0][action]=target

self.model.fit(state,target\_f,epochs=1,verbose=0)

def load (self, name):

self. model. load\_weights(name)

def save (self, name):

self. model. save\_weights(name)

class SumoIntersection:

def\_init\_(self)

try:

sys. path. Append (os. path. Join (os. path. dirname(\_file\_), ‘..’, ‘..’, ‘..’, ‘..’,"tools"))

sys. path. append (os. path. join (os.environ.get ("SUMO\_HOME”, os. path. join (os. path. dirname(\_file\_), ‘..’, ‘..’, ‘..’, ‘..’)),"tools"))

from sumolib import checkBinary

except ImportError:

sys. exit ("please declare environment variable 'SUMO\_HOME' as the root dictionary of your sumo installation (it should contain folders 'bin', 'tools'and 'docs')")

def generate\_routefile(self):

random. seed (42)

N=3600

pH=1. / 7

pV=1. / 11

pAR=1. / 30

pAL=1. / 25

with open("input\_routes.rou.xml","w") as routes:

print ('''<routes xmlns: xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi: nonamespacesSchemaLocation="http//sumo.dlr.de/xsd/rotes\_file.xsd">

<vType id="SUMO\_DEFAULT\_TYPE" accel="0.8" decl="4.5" sigma="0" length="5" minGap="2" maxSpeed="70"/>

<vType id="always\_right" edges="1fi 1si 4o 4fi 4si 2o 2fi 2si 3o 3fi 3si 1o 1fi"/>

<vType id="horizontal" edges="2fi 2si 1o 1fi 1si 2o 2fi"/>''')