Main.py

import random

import json

import pickle

import numpy as np

import nltk

nltk.download('punkt')

nltk.download('wordnet')

nltk.download('omw-1.4')

from nltk.stem import WordNetLemmatizer

from tensorflow.keras.models import load\_model

lemmatizer = WordNetLemmatizer()

intents = json.loads(open('intents.json').read())

words = pickle.load(open('words.pkl', 'rb'))

classes = pickle.load(open('classes.pkl', 'rb'))

model = load\_model('chatbotmodel.h5')

def clean\_up\_sentence(sentence):

    sentence\_words = nltk.word\_tokenize(sentence)

    sentence\_words = [lemmatizer.lemmatize(word)  for word in sentence\_words]

    return sentence\_words

def bag\_of\_words(sentence):

    sentence\_words= clean\_up\_sentence(sentence)

    bag = [0] \* len(words)

    for w in sentence\_words:

        for i, word in enumerate(words):

            if word == w:

                bag[i] = 1

    return np.array(bag)

def predict\_class(sentence):

    bow = bag\_of\_words(sentence)

    res = model.predict(np.array([bow]))[0]

    ERROR\_THRESHOLD = 0.25

    results = [[i,r] for i, r in enumerate(res) if r > ERROR\_THRESHOLD]

    results.sort(key=lambda  x:x[1], reverse=True)

    return\_list = []

    for r in results:

        return\_list.append({'intent': classes[r[0]], 'probability': str(r[1])})

    return return\_list

def get\_response(intents\_list,intents\_json):

    tag= intents\_list[0]['intent']

    list\_of\_intents =intents\_json['intents']

    for i in list\_of\_intents:

        if i['tag'] == tag:

            result = random.choice(i['responses'])

            break

    return result

print("|============= Welcome to College Equiry Chatbot System! =============|")

print("|============================== Feel Free ============================|")

print("|================================== To ===============================|")

print("|=============== Ask your any query about our college ================|")

while True:

    message = input("| You: ")

    if message == "bye" or message == "Goodbye":

        ints = predict\_class(message)

        res = get\_response(ints, intents)

        print("| Bot:", res)

        print("|===================== The Program End here! =====================|")

        exit()

    else:

        ints = predict\_class(message)

        res = get\_response(ints, intents)

        print("| Bot:", res)

train.py

import random

import json

import pickle

import numpy as np

import nltk

from nltk.stem import WordNetLemmatizer

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Activation, Dropout

from tensorflow.keras.optimizers import SGD

lemmatizer = WordNetLemmatizer()

intents = json.loads(open('intents.json').read())

words = []

classes = []

documents = []

ignore\_letters = ['?', '!',',','.']

for intent in intents['intents']:

    for pattern in intent['patterns']:

        word\_list = nltk.word\_tokenize(pattern)

        words.extend(word\_list)

        documents.append((word\_list,intent['tag']))

        if intent['tag'] not in classes:

            classes.append(intent['tag'])

words = [lemmatizer.lemmatize(word) for word in words if word not in ignore\_letters]

words = sorted(set(words))

classes = sorted(set(classes))

pickle.dump(words, open('words.pkl', 'wb'))

pickle.dump(classes, open('classes.pkl', 'wb'))

training = []

output\_empty = [0] \* len(classes)

for document in documents:

    bag =[]

    word\_patterns = document[0]

    word\_patterns = [lemmatizer.lemmatize(word.lower()) for word in word\_patterns]

    for word in words:

        bag.append(1) if word in word\_patterns else bag.append(0)

    output\_row = list(output\_empty)

    output\_row[classes.index(document[1])] = 1

    training.append([bag, output\_row])

random.shuffle(training)

training = np.array(training)

train\_x = list(training[:, 0])

train\_y = list(training[:, 1])

model = Sequential()

model.add(Dense(128, input\_shape=(len(train\_x[0]),), activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(64, activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(len(train\_y[0]), activation='softmax'))

sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)

model.compile(loss='categorical\_crossentropy', optimizer=sgd, metrics=['accuracy'])

hist = model.fit(np.array(train\_x), np.array(train\_y), epochs=200, batch\_size=5, verbose=1)

model.save('chatbotmodel.h5', hist)

print('Done')

intents.json

{"intents": [

  {"tag": "greetings",

  "patterns": ["hello","hey","hi","good day","Greetings","what's up?","how is it going?"],

  "responses": ["Hello!","Hey!","What can I do for you?"]

  },

  {"tag": "name",

  "patterns": ["what is your name","name","what's your name","who are you","what should I call you"],

  "responses": ["You can call me Ribot","I'm Ribot","I'm Ribot your virtual assistant"]

  },

  {"tag": "courses",

  "patterns": ["what courses are available", "how many courses are there in this college"],

  "responses": ["Courses offered by SRM KTR - B.Tech. - Civil Engineering, Biomedical Engineering, Electrical & Electronics Engineering, Electronics & Communication Engineering, Mechanical Engineering, Chemical Engineering, Biotechnology, Genetic Engineering, Food Process Engineering,\nBA - English (Hons.), Economics(Hons.), Political Science (Hons.), Psychology(Hons.),\nBachelor of Computer Application (BCA),\nB.Sc. - Computer Science (Hons.),Chemistry(Hons.),Physics(Hons.),Mathematics(Hons.)"]

  },

   {"tag": "courseDuration",

  "patterns": ["how long will be BTech course", "how long will it take to complete the course"],

  "responses": ["Our college offers 4 year long BTECH course and 2 year long MTECH course."]

  },

   {"tag": "Location",

  "patterns": ["location","where is it located","what is the location of the college"],

  "responses": ["Near Potheri,\nkattankulathur,\npincode-403202,\nKanchipuram."]

  },

  {"tag": "semesters",

  "patterns": ["how many semesters are there in a year","how many semesters one should study in a year"],

  "responses": ["There are two semesters in a year."]

  },

  {"tag": "semDuration",

  "patterns": ["how many months are there in a semester","how long will be a single semester"],

  "responses": ["The single semester will be around 4 months."]

  },

  {"tag": "studentRequirements",

  "patterns": ["what are the student requirements for admission","entry requirements","admission requirements"],

  "responses": ["Should have passed 10+2 or equivalent examination with minimum 50 percent.\nAlso, Physics, Chemistry, and Mathematics should have been studied as main subjects.\nHowever, the minimum percentage marks in the qualifying examination for SRM KTR is 60%"]

  },

  {"tag": "classes",

  "patterns": ["how many classes will be there in a day","how long are the classes?"],

  "responses": ["There may be 6-7 classes per day, depends on the Day order. Each class will be of 50 minutes."]

  },

  {"tag": "teachingStyle",

  "patterns": ["what is the teaching style of this college?","Is the teaching pattern different from other college?","what is the teaching format?"],

  "responses": ["Our college has different teaching patterns than other colleges.The faculty creates a positive and caring environment for student development and learning.\nA variety of teaching methods and materials are employed, utilizing media and technology to the maximum.\nThe faculty also plan informative sessions that involve individual development and teamwork."]

  },

  {"tag": "exams",

  "patterns": ["what are the exams like?","What is the exam pattern"],

  "responses": ["There are 3 Assessments CT 1,2 and 3 which carries 50 marks of total 100 marks.\nFinal University exams carries remaining 50 marks."]

  },

  {"tag": "hours",

  "patterns": ["what are your hours","when are you guys open","what your hours of operation"],

  "responses": ["You can message us here at any hours. But our college premises will be open from 8:00 am to 5:00 pm only."]

  },

  {"tag": "funActivities",

  "patterns": ["will there be any extra curriculum activities?","does the college conducts any fun program"],

  "responses": ["Yes, Of course. Our college not only provides excellent education but also encourage students to take part in different curriculum activities. The college conducts yearly programs like Sports meet, Carnival, Holi festival, and Christmas. \n Also our college has basketball court, badminton court, table tennis, chess, carrom board and many more refreshment zones."]

  },

  {"tag": "facilities",

  "patterns": ["what facilities are provided by the college?","what are the facilities of college for students", "what are the college infrastructures "],

  "responses": ["With excellent education facilities, Our College provides various other facilities like 24 hours internet, library, classes with AC, discusson room, canteen, parking space, and student service for any students queries."]

  },

  {"tag": "fee",

  "patterns": ["how much is the college fee","what is the fee structure"],

  "responses": ["B.Tech. - Computer Science and Engineering (CSE), Tution fee-2,00,000/-.\nComputer Science & Engineering with specialization in Data Science & Artificial Intelligence (in association with IBM), Tution fee-2,50,000/,\nCivil Engineering (CE), Biomedical Engineering (BME), Electrical & Electronics Engineering (EEE),Electronics & Communication Engineering (ECE), Mechanical Engineering (ME), Tution fee-1,75,000/-,\nM.Tech. - Civil (Structural Engineering), M.Tech. - Electronics & Communication (Microelectronics and VLSI Design ),Production Engineering,Computer Science and Engineering (Big Data Analytics), Tution fee-1,00,000/- "]

  },

  {"tag": "goodbye",

  "patterns": ["cya","See you later","Goodbye","I am leaving","Have a Good Day","bye","see ya"],

  "responses": ["Sad to see you go :(","Talk you later","Goodbye"]

  },

  {"tag": "invalid",

    "patterns": ["","gvsd","asbhk"],

    "responses": ["Sorry, can't understand you", "Please give me more info", "Not sure I understand"]

  },

  {"tag": "thanks",

    "patterns": ["Thanks", "Thank you", "That's helpful", "Awesome, thanks", "Thanks for helping me"],

    "responses": ["Happy to help!", "Any time!", "My pleasure"]

  }