

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel) O

File Edit View Insert Cell Kernel Widgets Help

Code

```
In [1]: !pip install google_trans_new  
!pip install langdetect  
!pip install SpeechRecognition  
!pip install tkinter  
!pip install pytesseract  
!pip install Pillow
```

```
...  
  
In [ ]:  
import pytesseract  
from PIL import Image ,ImageTk  
import pytsx3  
import speech_recognition as sr  
from langdetect import DetectorFactory  
from langdetect import detect_langs  
from google_trans_new import google_translator  
from langdetect import detect  
  
DetectorFactory.seed = 0  
  
def detect_and_translate(text,target_lang):  
    result_lang = detect(text)  
    if result_lang == target_lang:  
        return text  
    else:  
        translator = google_translator()  
        translate_text = translator.translate(text,lang_src=result_lang,lang_tgt=target_lang)  
        return translate_text
```

```
In [2]: from tkinter import *  
from tkinter import filedialog  
import pytesseract  
from PIL import Image ,ImageTk  
import pytsx3  
import speech_recognition as sr  
from langdetect import DetectorFactory  
from langdetect import detect_langs  
from google_trans_new import google_translator  
from langdetect import detect  
  
#UpdATED  
#FINAL GUI  
  
root = Tk()  
root.title("Index MMT")  
  
# root.iconbitmap('Translator.ico')  
#getting screen width and height of display  
width= root.winfo_screenwidth()  
height= root.winfo_screenheight()  
  
#setting tkinter window size  
root.geometry("%dx%d" %(width, height))  
root.configure(bg="#856fff8")  
  
DetectorFactory.seed = 0  
  
def detect_and_translate(text,target_lang):  
    result_lang = detect(text)  
    if result_lang == target_lang:  
        return text  
    else:  
        translator = google_translator()  
        translate_text = translator.translate(text,lang_src=result_lang,lang_tgt=target_lang)  
        return translate_text  
  
  
#input type-1  
def word_sentence():  
  
    def printData(res):  
        #     print(res)  
        #     result = res  
        #     resultLabel.config(padx = 10, justify = CENTER, font =("Courier", 14), text=result)  
        #     text.delete("1.0", "end")  
        #     text.insert(INSERT, res)  
        #     text.pack()  
  
    def get_input():  
        sentence = entry1.get()  
        code = entry2.get()  
        r = detect_and_translate(sentence,target_lang=code)  
        printData(r)  
  
    def delete():
```

```

        entry1.delete(0,'end')
        entry2.delete(0,'end')

root1 = Tk()
root1.title("Multilingual Machine Translation")
# root.geometry("500x500")

#getting screen width and height of display
width= root1.winfo_screenwidth()
height= root1.winfo_screenheight()

#setting tkinter window size
root1.geometry("%dx%d" % (width, height))
root1.configure(bg="#856ff8")

text = Text(root1)

label1 = Label(root1,text = 'Enter Sentence')
label1.pack()
label1.config(padx = 10, justify = LEFT, font =("Courier", 14))

entry1 = Entry(root1, width = 70, bd=2)
entry1.pack()

label2 = Label(root1, text="Enter Lang_code")
label2.pack()
label2.config(padx = 10,justify = LEFT,font =("Courier", 14))

entry2 = Entry(root1, width = 70, bd=2)
entry2.pack()

button1 = Button(root1, text ="submit",font =("Courier", 14),activebackground="#00ff00")
button1.pack()
button1.config(command = get_input, padx=10, bd=2, width=10, fg='White', bg= 'dark green')

# resultLabel = Label(root1, text ="\\nTranslated text Here!!")
# resultLabel.pack()

button2 = Button(root1, text = "Delete", command = delete,font =("Courier", 14),activebackground="#00ff00")
button2.pack(pady = 5)
button2.config(padx=10, bd=2, width=10, fg='White', bg= 'dark red')

button3 = Button(root1, text = "Exit", command=root1.destroy,font =("Courier", 14),activebackground="#00ff00")
button3.pack(pady = 5)
button3.config( padx=10, bd=2, width=10, fg='White', bg= 'dark red')

root1.mainloop()

#input type-2
def txtfile():

root2 = Tk()
root2.title("Multilingual Machine Translation")

# root.geometry("500x500")

#getting screen width and height of display
width= root2.winfo_screenwidth()
height= root2.winfo_screenheight()

#setting tkinter window size
root2.geometry("%dx%d" % (width, height))
root2.configure(bg="#856ff8")

text = Text(root2)

def printData(res):
#     print(res)
#     result = res
#     res_text.config(padx = 10, justify = CENTER, font =("Courier", 14), text=result)
    text.delete("1.0", "end")
    text.insert(INSERT, res)
    text.pack()

# Function for opening the file
def openFile():
    filepath = filedialog.askopenfilename(initialdir=r"C:\Users\Porika Dhanrajnath\Desktop\NLP",
                                           title="Open file",
                                           filetypes= (("text files","*.txt"),
                                                       ))
#     file = open(filepath, 'r')
#     text_data = open(filepath, 'r').read()
    text_data = open(filepath, encoding="utf8")
    sentence = text_data.read()
    code = entry1.get()
    r = detect_and_translate(sentence,target_lang=code)
    printData(r)

label1 = Label(root2, text="Enter Lang_code")
label1.pack()
label1.config(padx = 10,justify = LEFT,font =("Courier", 14))

entry1 = Entry(root2, width = 70, bd=2)
entry1.pack()

```

```

# Button Label
button1 = Button(root2, text ='Choose file', command=openFile, font =("Courier", 14),activebackground="#00ff00")
button1.pack(pady = 5)
button1.config(padx=10, bd=2, width=15, fg='White', bg= 'dark red')

# button2 = Button(root2, text ='submit',font =("Courier", 14),activebackground="#00ff00")
# button2.pack()
# button2.config(command = get_input, padx=10, bd=2, width=10, fg='White', bg= 'dark green')

# resultLabel = Label(root2, text ="\\nTranslated text Here!!",height=9)
# resultLabel.pack(pady=20)
# resultLabel.config(padx = 10,justify = LEFT,font =("Courier", 14))

button3 = Button(root2, text = "Exit", command=root2.destroy,font =("Courier", 14),activebackground="#00ff00")
button3.pack(pady = 5)
button3.config( padx=10, bd=2, width=10, fg='white', bg= 'dark red')

root2.mainloop()

#input type-3
def img_file():

    root3 = Tk()
    root3.title("Multilingual Machine Translation")

    # root.geometry("500x500")

    #getting screen width and height of display
    width= root3.winfo_screenwidth()
    height= root3.winfo_screenheight()

    #setting tkinter window size
    root3.geometry("%dx%d" % (width, height))
    root3.configure(bg='#85ffff')

    text = Text(root3)

    def printData(res):
        # print(res)
        # result = res
        # res.text.config(padx = 10, justify = CENTER, font =("Courier", 14), text=result)
        text.delete("1.0", "end")
        text.insert(INSERT, res)
        text.pack()

    def select_image():
        path = filedialog.askopenfilename()
        code = entry1.get()
        img = Image.open(path)
        printData(img)
        pytesseract.pytesseract.tesseract_cmd ='C:/Program Files (x86)/Tesseract-OCR/tesseract.exe'
        result_text = pytesseract.image_to_string(img)
        # result_text = pytesseract.image_to_string(img, Lang='te')
        # print("Text detected!")
        # print("\n"+result_text)
        text4=detect_and_translate(result_text,target_lang=code)
        printData(text4)

    label1 = Label(root3, text="Enter Lang_code")
    label1.pack()
    label1.config(padx = 10,justify = LEFT,font =("Courier", 14))

    entry1 = Entry(root3, width = 70, bd=2)
    entry1.pack()

    # Button Label
    button1 = Button(root3, text="Choose image", command=select_image, font =("Courier", 14),activebackground="#00ff00")
    button1.pack(pady=5)
    button1.config( padx=10, bd=2, width=10, fg='White', bg= 'dark red')

    button3 = Button(root3, text = "Exit", command=root3.destroy,font =("Courier", 14),activebackground="#00ff00")
    button3.pack(pady = 5)
    button3.config( padx=10, bd=2, width=10, fg='white', bg= 'dark red')

    root3.mainloop()

#input type-4
def voice():

    root4 = Tk()
    root4.title("Multilingual Machine Translation")

    # root.geometry("500x500")

    #getting screen width and height of display
    width= root4.winfo_screenwidth()
    height= root4.winfo_screenheight()

    #setting tkinter window size
    root4.geometry("%dx%d" % (width, height))
    root4.configure(bg='#85ffff')

```

```

text = text(root4)
# dtext= Text(root4)

def printData(res):
#     print(res)
#     result = res
#     res_text.config(padx = 10, justify = CENTER, font =("Courier", 14), text=result)
text.delete("1.0", "end")
text.insert(INSERT, res)
text.pack()

def detect_text(s):
    label2 = Label(root4,text=s)
    label2.pack()
    label2.config(padx = 10, justify = CENTER, font =("Courier", 14))

def record():

    recorder=sr.Recognizer()
    with sr.Microphone() as source:
#         print("Speak Now! ")
        printData("Speak Now!")
        audio=recorder.listen(source)

# txt=recorder.recognize_google(audio)
    try:
# Auto detect the language
        txt=recorder.recognize_google(audio)
    except sr.UnknownValueError:
        printData("Google Speech Recognition could not understand audio")
    except sr.RequestError as e:
        printData("Could not request results from Google Speech Recognition service")
    detect_text("You said: "+txt)
    lang=entry1.get()
    text3=detect_and_translate(txt,target_lang=lang)
    printData("\n"+text3)

label1 = Label(root4, text="Enter Lang_code")
label1.pack()
label1.config(padx = 10,justify = LEFT,font =("Courier", 14))

entry1 = Entry(root4, width = 70, bd=2)
entry1.pack()

#Button Label

button1 = Button(root4, text=" record voice", command=record, font =("Courier", 14),activebackground="#00ff00")
button1.pack(pady=5,side=TOP)
button1.config( padx=10, bd=2, width=13, fg='White', bg= 'dark red')

button3 = Button(root4, text = "Exit", command=root4.destroy,font =("Courier", 14),activebackground="#00ff00")
button3.pack(pady = 5)
button3.config( padx=10, bd=2, width=10, fg='White', bg= 'dark red')

root4.mainloop()

#Main root- GUI

# def Lang_code_print():
#     f=open("Lang_codes.txt", "r")
#     res=f.read()
#     text = Text(root)
#     text.insert(INSERT, res)
#     text.pack()

T = Text(root, height=4, width=50, bd=4, font =("Courier", 14),highlightbackground="#00ff00")
T.pack(side=LEFT, fill=Y)
lang_code_data="""
*Languages with codes*

code      Language

ar->Arabic
as->Assamese
bn->Bengali
en->English
fr->French
de->German
gu->Gujarati
hi->Hindi
it->Italian
ja->Japanese
kn->Kannada
ks->Kashmiri
la->Latin
ml->Malayalam
mr->Marathi
mn->Mongolian
ne->Nepali
pa->Panjabi; Punjabi
sa->Sanskrit
ta->Tamil
te->Telugu

```

```

th-->Thai
bo-->Tibetan
tr-->Turkish
vi-->Vietnamese
ur-->Urdu
"""

T.insert(END, lang_code_data)

label1 = Label(root,text = 'Welcome to Multilingual Machine Translation')
label1.pack()
label1.config(padx = 10, justify = CENTER, font =("Courier", 14))

# Lang_button = Button(root, text = "Languages with codes",font =("Courier", 14),activebackground="#00ff00")
# Lang_button.pack()
# Lang_button.config(command = Lang_code_print, padx=10, bd=2, width=20, fg='White', bg='dark green')

button1 = Button(root, text ='word or sentence',font =("Courier", 14),activebackground="#00ff00")
button1.pack()
button1.config(command = word_sentence, padx=10, bd=2, width=20, fg='White', bg='dark green')

button2 = Button(root, text ='text file',font =("Courier", 14),activebackground="#00ff00")
button2.pack()
button2.config(command = txtfile, padx=10, bd=2, width=20, fg='White', bg='dark green')

button3 = Button(root, text ='Image',font =("Courier", 14),activebackground="#00ff00")
button3.pack()
button3.config(command = img_file, padx=10, bd=2, width=20, fg='White', bg='dark green')

button4 = Button(root, text ='Voice',font =("Courier", 14),activebackground="#00ff00")
button4.pack()
button4.config(command = voice, padx=10, bd=2, width=20, fg='White', bg='dark green')

button5 = Button(root, text = "Exit", command=root.destroy,font =("Courier", 14),activebackground="#00ff00")
button5.pack(pady = 5)
button5.config(padx=10, bd=2, width=10, fg='White', bg= 'dark red')

root.mainloop()

Exception in Tkinter callback
Traceback (most recent call last):
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\tkinter\__init__.py", line 1892, in __call__
    return self.func(*args)
  File "C:\Users\PORIKA~1\AppData\Local\Temp\ipykernel_11708/1850852732.py", line 60, in get_input
    r = detect_and_translate(sentence,target_lang=code)
  File "C:\Users\PORIKA~1\AppData\Local\Temp\ipykernel_11708/1850852732.py", line 39, in detect_and_translate
    translate_text = translator.translate(text,lang_src=result_lang,lang_tgt=target_lang)
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\site-packages\google_trans_new\google_trans_new.py", line 188, in translate
    raise e
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\site-packages\google_trans_new\google_trans_new.py", line 152, in translate
    response = json.loads(response)
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\json\__init__.py", line 357, in loads
    return _default_decoder.decode(s)
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\json\decoder.py", line 340, in decode
    raise JSONDecodeError("Extra data", s, end)
json.decoder.JSONDecodeError: Extra data: line 1 column 391 (char 390)
Exception in Tkinter callback
Traceback (most recent call last):
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\tkinter\__init__.py", line 1892, in __call__
    return self.func(*args)
  File "C:\Users\PORIKA~1\AppData\Local\Temp\ipykernel_11708/1850852732.py", line 121, in word_sentence
    root1.mainloop()
  File "C:\Users\Porika Dhanrajnath\anaconda3\lib\tkinter\__init__.py", line 1429, in mainloop
    self.tk.mainloop(n)
KeyboardInterrupt

```

In []: