

COSC 310 - 001
Software Engineering
2021 Winter Term 2

Assignment 4 - Individual Project Documentation

Tanner Wright

Table of Contents

Table of Contents	1
1.0 Googletrans API	
1.1 getLang	2
1.2 responseTrans	2
1.3 transNode	3
2.0 wikipedia API	
2.1 wikipedia.page	4
2.2 wikipedia.summary	4
2.3 wikipedia.section	5
3.0 API Summary	6

1.0 Googletrans API

1.1 getLang

```
def getLang(self, response):  
    translator = googletrans.Translator()  
    tran = translator.translate(response, dest='english')  
    return tran.src
```

This method was designed to take in the user's response and using the googletrans API, determine the language the user is speaking in and return the language which will be used by later methods when translating back into the users language.

1.2 responseTrans

```
def responseTranslate(self, response):  
    translator = googletrans.Translator()  
    tran = translator.translate(response, dest='english')  
  
    return tran.text
```

This method takes in the users response using googletrans API and will translate and return the users message in English which will be used later by the nodes which are written in English, this new English response will be read and the correct response will be determined by the bot.

1.3 transNode

```
def transNode(self, response, lang):  
    translator = googletrans.Translator()  
    tran = translator.translate(response, src = 'en', dest = lang)  
  
    return tran.text
```

This method will take in the user's language which was determined from the getLang method that was determined earlier, as well as the bot's response to the user's already translated message. Using the googletrans API, the response will be translated to the user's language and then sent back to the app.py method to be displayed as a translated response to the user's message.

2.0 wikipedia API

2.1 wikipedia.page

```
try:
    wikipedia.set_lang(language)
    page_py = wikipedia.page(baseAnswer)
    newtext = self.transNode('here is a link containing more information on ', language)
    nodeValue['text'] = newtext + baseAnswer + ': ' + page_py.url
except:
    newtext = self.transNode('I cant find any information on wikipedia regarding that. I would consult your doctor for more information ', language)
    nodeValue['text'] = newtext
```

The built-in wikipedia.page function allows us to enter a string which we get from the user and returns the web page of the specified string. In this specific case, we are taking in the page's url, translating and displaying it to the user in the correct language.

2.2 wikipedia.summary

```
try:
    wikipedia.set_lang('en')
    abuses = wikipedia.summary("Abuse", sentences = 3)
    transtext = self.transNode(abuses, language)
    newtext = self.transNode('Here is a brief summary on substance abuse from wikipedia: ', language)
    nodeValue['text'] = newtext + ': ' + transtext
except:
    newtext = self.transNode('Drugs and alcohol can be highly addictive, you should try to refrain from taking them in high quantities.', language)
    nodeValue['text'] = newtext
```

The built-in wikipedia.summary function allows us to enter a string and the amount of sentences we want, which will be used to grab a wikipedia summary on the specified string. In this specific case, we are looking at the abuse wikipedia page and translating it to the user's language and giving a brief translated summary of abuse to the user.

2.3 wikipedia.section

```
try:
    wikipedia.set_lang('en')
    mc = wikipedia.page(answer)
    print(answer)
    stringBuild = 'Here are some of the wikipedia suggested readings associated with ' + answer + ': '
    sec = mc.section('Further reading')
    stringBuild = stringBuild + sec
    newtext = self.transNode(stringBuild, language)
    nodeValue['text'] = stringBuild
except:
    transtest = 'There is no recommended wikipedia readings associated with ' + answer
    newtext = self.transNode(transtest, language)
    nodeValue['text'] = newtext
```

Wikipedia API also contains a function: `wikipedia.section`, which allows us to take specific sections within the page. In this case we are taking the “Further reading” section which will be returned and displayed to the user. This will be used in the `bot.py` class to offer more information about the selected wikipedia page and some similar topics.

3.0 API Summary

The purpose of the new additions of the wikipedia API and googletrans API was to create a program which can be used by anyone speaking whichever language they choose. By integrating these API's the amount of information available to the user is determined by them. By offering more information on certain subjects or giving the opportunity to the user to discover more about something they discuss creates a whole new way of interacting with users in a unique way. Both API's are documented in the code and are integrated in a way that requires both of them to work together to create an ideal chatbot for anyone speaking any language.