“La Sapienza” University of Rome
Faculty of information engineering, information technology and statistics
Department of informatics, automation and control engineering
"ANTONIO RUBERTI"
Degree program: Artificial Intelligence and Robotics

HOMEWORK
Task-oriented SDS

Student:
OREL, Egor
Matricola: 1836231

Student:
NAUTIYAL, Anupam
Matricola: 1850513

Student:
DEMIRAJ, Joris
Matricola: 1836521

Teacher: NARDI, Daniele

Rome, 2019

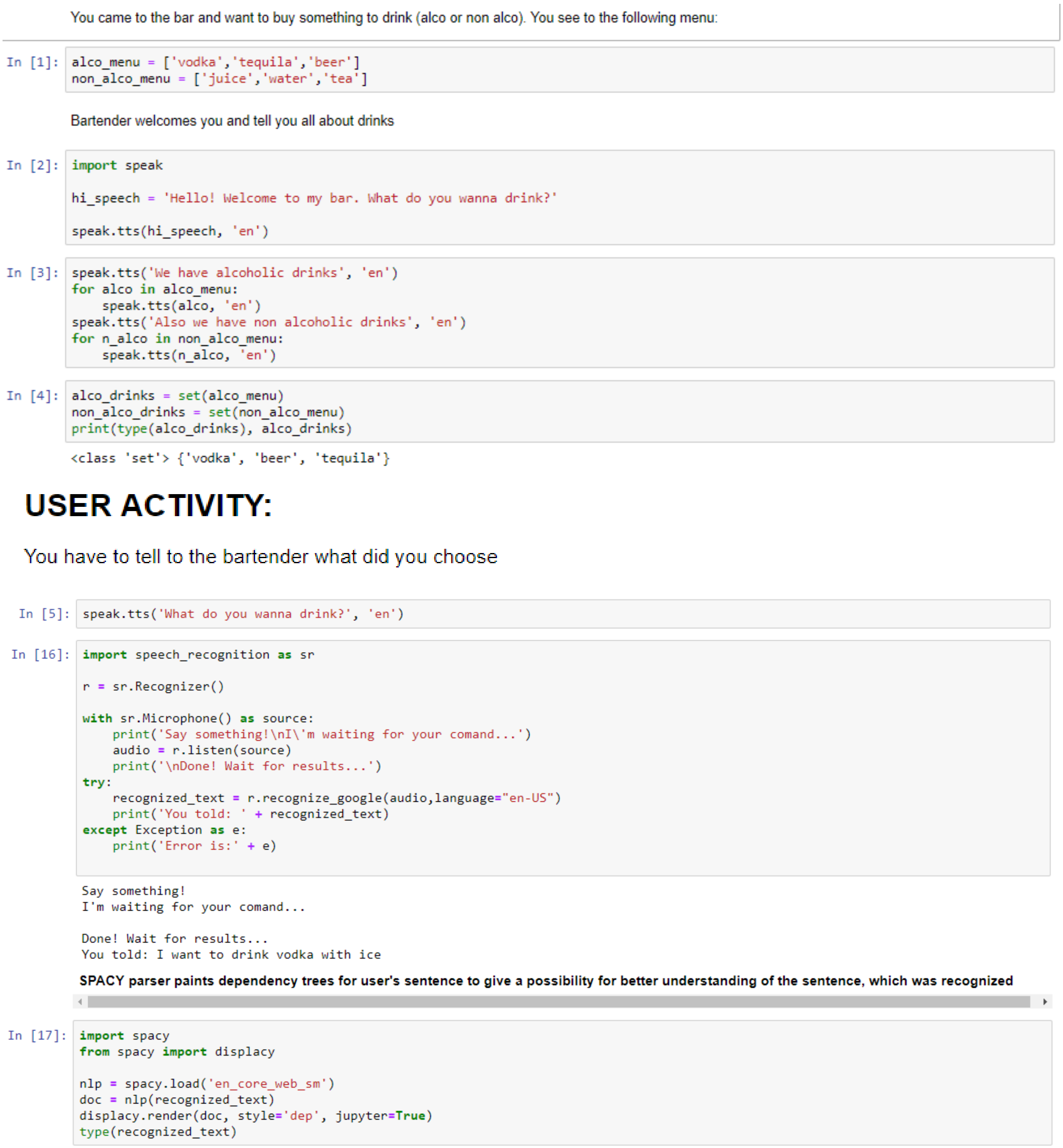
Exercise: to implement task-oriented SDS according to a preferable scenario.

Instruments: Google Speech Recognition API (1), SPACY Parser (2), Jupyter Notebook (3).

Mechanism of the program: a program written in (3) provides a dialog with a user. It recognizes user's voice and translates it into a text using (1), after that works out this data and "understands its meaning" using (2) and gives an answer based on output data of (2).

Scenario: a bartender controls age of a user when a user tries to buy alcohol.

The report is given as a collection of screenshots from (3).



You came to the bar and want to buy something to drink (alco or non alco). You see to the following menu:

```
In [1]: alco_menu = ['vodka','tequila','beer']
        non_alco_menu = ['juice','water','tea']
```

Bartender welcomes you and tell you all about drinks

```
In [2]: import speak

        hi_speech = 'Hello! Welcome to my bar. What do you wanna drink?'

        speak.tts(hi_speech, 'en')
```

```
In [3]: speak.tts('We have alcoholic drinks', 'en')
        for alco in alco_menu:
            speak.tts(alco, 'en')
        speak.tts('Also we have non alcoholic drinks', 'en')
        for n_alco in non_alco_menu:
            speak.tts(n_alco, 'en')
```

```
In [4]: alco_drinks = set(alco_menu)
        non_alco_drinks = set(non_alco_menu)
        print(type(alco_drinks), alco_drinks)

        <class 'set'> {'vodka', 'beer', 'tequila'}
```

USER ACTIVITY:

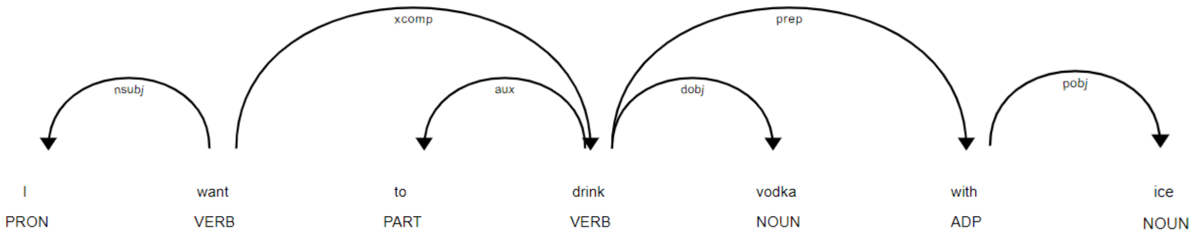
You have to tell to the bartender what did you choose

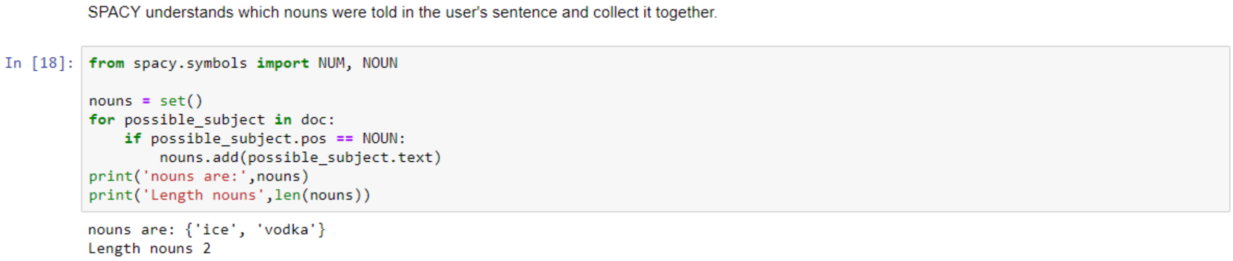
```
In [5]: speak.tts('What do you wanna drink?', 'en')
```

```
In [16]: import speech_recognition as sr

         r = sr.Recognizer()

         with sr.Microphone() as source:
             print('Say something!\nI\'m waiting for your comand...')
             audio = r.listen(source)
             print('\nDone! Wait for results...')
         try:
             recognized_text = r.recognize_google(audio,language="en-US")
             print('You told: ' + recognized_text)
         except Exception as e:
             print('Error is:' + e)

         Say something!
         I'm waiting for your comand...

         Done! Wait for results...
         You told: I want to drink vodka with ice
```

SPACY parser paints dependency trees for user's sentence to give a possibility for better understanding of the sentence, which was recognized

```
In [17]: import spacy
         from spacy import displacy

         nlp = spacy.load('en_core_web_sm')
         doc = nlp(recognized_text)
         displacy.render(doc, style='dep', jupyter=True)
         type(recognized_text)
```





SPACY understands which nouns were told in the user's sentence and collect it together.

In [18]:

```
from spacy.symbols import NUM, NOUN

nouns = set()
for possible_subject in doc:
    if possible_subject.pos == NOUN:
        nouns.add(possible_subject.text)
print('nouns are:',nouns)
print('Length nouns',len(nouns))
```

```
nouns are: {'ice', 'vodka'}
Length nouns 2
```

Text continues at the next page...

USER ACTIVITY:

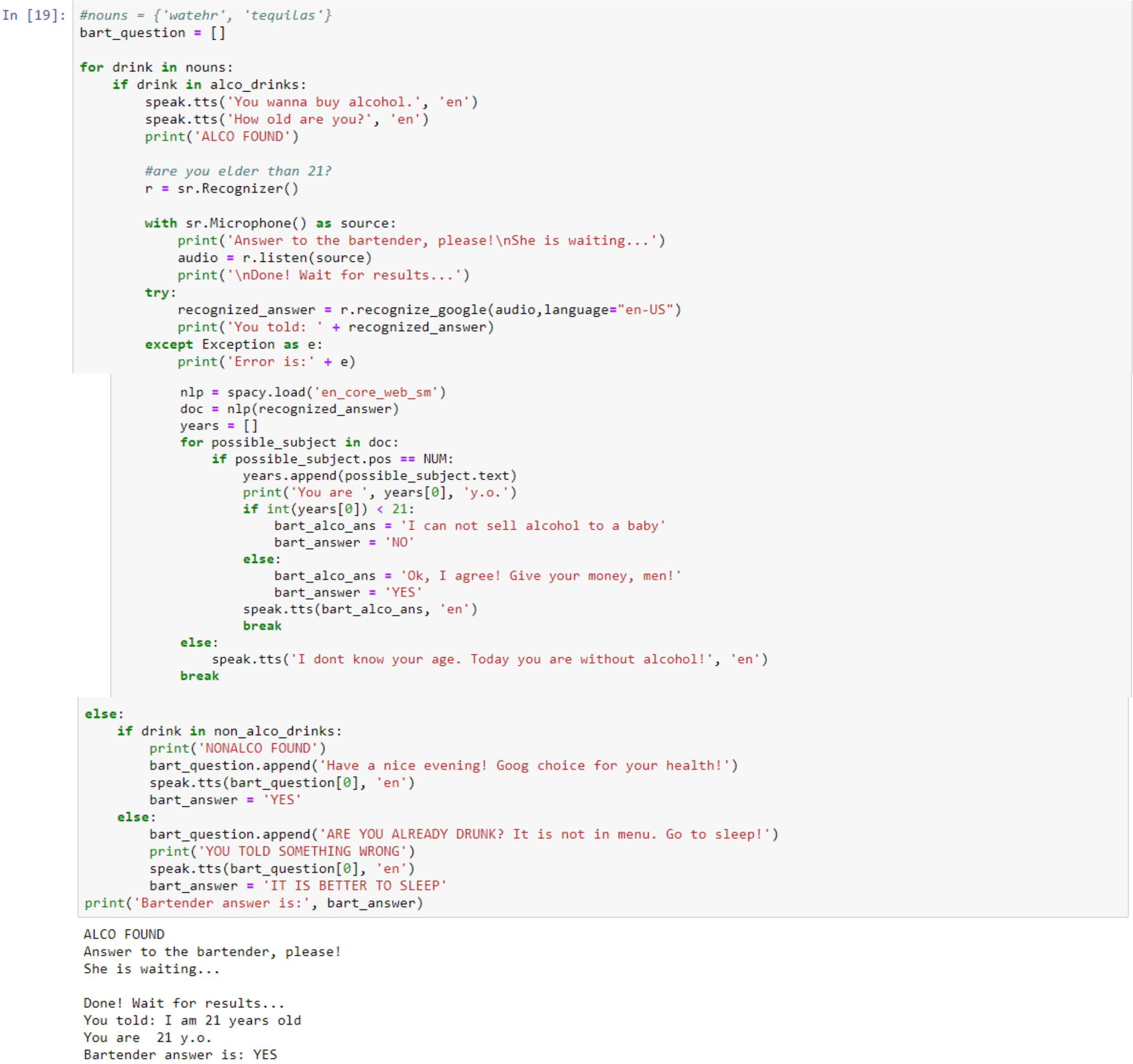
program understands, which drinks do you want to buy and provides different reaction depending on this:

1) alco drink: asks, how old is user and gives him a possibility to answer with numbers:

```
a) if a user tells < than 21 y.o., then the bartender provides a negative answer;
b) if a user tells >= than 21 y.o., then the bartender provides a positive answer;
```

2) non alco drink: the bartender provides a positive answer and asks money;

3) words of a user does not contain contents of the menu: negative answer and a user has to go to the start of the program.



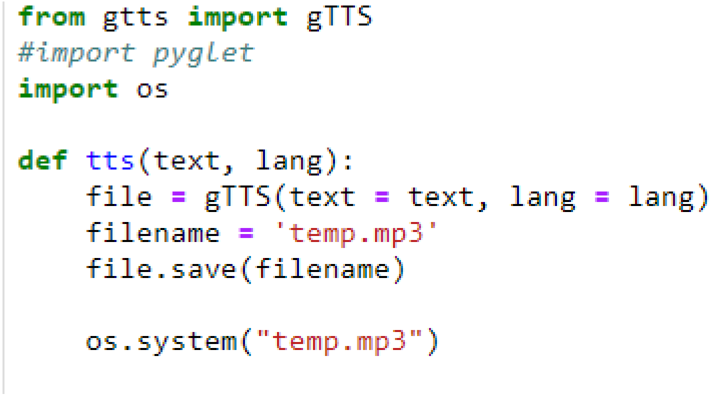
In [19]:

```python
#nouns = {'watehr', 'tequilas'}
bart_question = []

for drink in nouns:
    if drink in alco_drinks:
        speak.tts('You wanna buy alcohol.', 'en')
        speak.tts('How old are you?', 'en')
        print('ALCO FOUND')

        #are you elder than 21?
        r = sr.Recognizer()

        with sr.Microphone() as source:
            print('Answer to the bartender, please!\nShe is waiting...')
            audio = r.listen(source)
            print('\nDone! Wait for results...')
        try:
            recognized_answer = r.recognize_google(audio,language="en-US")
            print('You told: ' + recognized_answer)
        except Exception as e:
            print('Error is:' + e)

        nlp = spacy.load('en_core_web_sm')
        doc = nlp(recognized_answer)
        years = []
        for possible_subject in doc:
            if possible_subject.pos == NUM:
                years.append(possible_subject.text)
                print('You are ', years[0], 'y.o.')
                if int(years[0]) < 21:
                    bart_alco_ans = 'I can not sell alcohol to a baby'
                    bart_answer = 'NO'
                else:
                    bart_alco_ans = 'Ok, I agree! Give your money, men!'
                    bart_answer = 'YES'
                speak.tts(bart_alco_ans, 'en')
                break
        else:
            speak.tts('I dont know your age. Today you are without alcohol!', 'en')
        break

else:
    if drink in non_alco_drinks:
        print('NONALCO FOUND')
        bart_question.append('Have a nice evening! Goog choice for your health!')
        speak.tts(bart_question[0], 'en')
        bart_answer = 'YES'
    else:
        bart_question.append('ARE YOU ALREADY DRUNK? It is not in menu. Go to sleep!')
        print('YOU TOLD SOMETHING WRONG')
        speak.tts(bart_question[0], 'en')
        bart_answer = 'IT IS BETTER TO SLEEP'
print('Bartender answer is:', bart_answer)
```

```
ALCO FOUND
Answer to the bartender, please!
She is waiting...

Done! Wait for results...
You told: I am 21 years old
You are  21 y.o.
Bartender answer is: YES
```

Conclusion

Resulting program has a good ability for recognizing and understanding human English spoken language (fast Internet connection and silence in environment around are needed). It uses following libraries: speech_recognition, spacy, speak. Model “speak” was written by myself and contains the following text:



```
from gtts import gTTS
#import pyglet
import os

def tts(text, lang):
    file = gTTS(text = text, lang = lang)
    filename = 'temp.mp3'
    file.save(filename)

    os.system("temp.mp3")
```

Possible improvements

Algorythm of the dialogue can be made longer and deeper. Module “speak” can use internal Python *.mp3 player instead of using external program. Also some moments of the program can be improved to increase efficiency of calculations, but for this little demonstration task it is enough.