1.Change

1.1The Change Of GUI

Add a console to the right to better display information to the user.

Code implementation

Add a textEdit and set properties。

```
    self.textEdit = QtWidgets.QTextEdit(self.centralwidget)
    self.textEdit.setGeometry(QtCore.QRect(314, 10,314,748))
    self.textEdit.setObjectName("textEdit")
    self.textEdit.setStyleSheet("background-color: Black ")
    self.textEdit.setReadOnly(True)

 self.textEdit.verticalScrollBar().setValue(self.textEdit.verticalScrollBar().ma
ximum())
```

Add output functions to UI class

```
    def print(self,string):
        self.textEdit.append("<font color=\"#F8F8FF\">"+string+"</font> ");
```

Because another listening thread is used to output to the UI thread, you need to use the signal to implement it。

```
_signal=QtCore.pyqtSignal(str)          #添加信号
_signal.connect(self.print)             #绑定信号到print函数
```

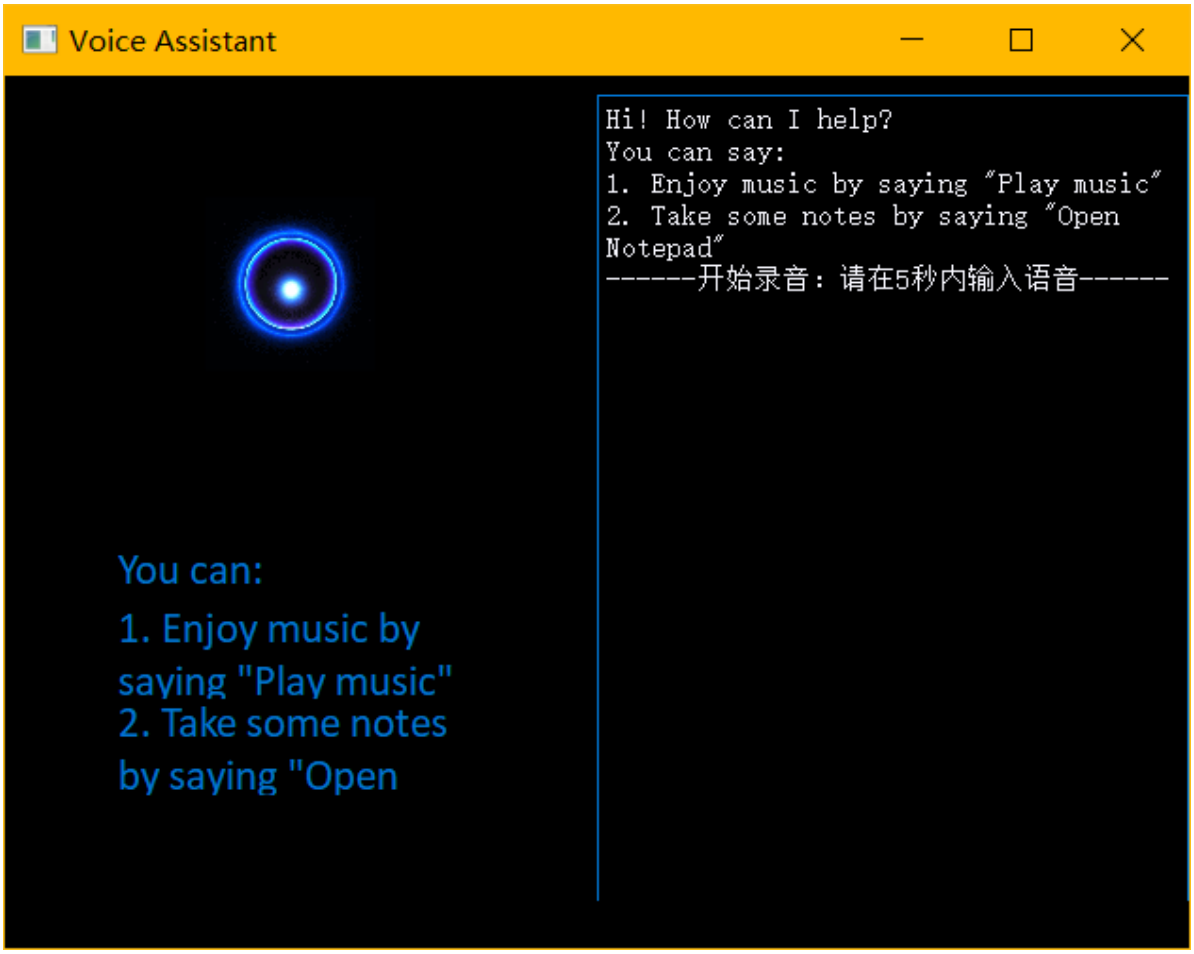
Complete output by emit() function

```
self._signal.emit("Hi! How can I help?")    #调用信号的emit函数进行输出
```

I wanted to add a background picture, but I felt uglier and didn't set it up

```
MainWindow.setStyleSheet("#MainWindow{border-image:url(./icon/bg.jpg);}")
```

The final results are as follows:



1.2Code changes

Three additional categories

1. LAEThread

   Defined in listenAndExecuteThread.py, inheritance and Thread classes, through which to create a new thread. Listen to microphone input and call Baidu speech recognition interface in the Recongnizer. If the result is returned successfully, it is left to the Orders to process, and the re is used to determine whether to execute a command or not。

2. Orders

   The matching of recognition results and commands is judged by regular expressions. Execute each order by win32api。

3. Recognizer

   External recorder_and_rec (loan, language) methods

   ```
       def recorder_and_rec(self,language):
           frames_data=self.record(5,"./temp.wav")
           return self.get_result(frames_data,'wav',self.rate,language)
   ```

   External needs to provide parameters language,language language code: Baidu provides interface support language as follows：

dev_pid	语言	模型	是否有标点	备注
1537	Mandarin (Pure Chinese Recognition)	Input Model	Yes	Support for custom word libraries
1737	English	English Model	Yes	Don't support for custom word libraries
1637	Cantonese	Cantonese model	Yes	Don't support for custom word libraries
1837	Sichuan dialect	Sichuan dialect model	Yes	Don't support for custom word libraries
1936	Mandarin Far Away	Far-field model	Yes	Don't support for custom word libraries

The method calls the record to record n seconds of voice stored in the temp.wav, and then calls the get_result to obtain Baidu interface results. (see code for implementation).

Speech Recognition Accuracy

The recognition accuracy of speech using the speech_recognition library was low, only accidentally opening the music once. The main reason is that the pronunciation requirements are relatively high, can not meet the conditions of its recognition, its own recognition effect is not very good.

Therefore, Tencent 's voice recognition interface was used, but there was a problem with his official SDK library, but Tencent' s official library has a problem. Due to the python version, it could not be successfully imported into SSL,, so it did not successfully realize the Tencent's voice recognition interface. So the turn to Baidu's voice recognition interface, for a high success rate of Chinese recognition, almost every time can be successful.