此文章已于 15:36:15 2016/12/6 发布到 Elmo66的博客

ChatterBot聊天机器人教程01

* 创建第一个BOT聊天

创建一个名为chatbot.py新文件。然后打开chatbot.py，导入包和实例化一个ChatBot。

|  |
| --- |
| from chatterbot import ChatBot  bot = ChatBot(*'Norman'*) |

* 存储适配器

聊天机器人有自带的适配器类，允许它连接到不同类型的数据库。在本教程中，我们将使用JsonFileStorageAdapter，这是可以存储在硬盘上的一个JSON格式文件存储数据的简单存储适配器。此功能使得这款存储适配器进行测试和调试非常好。JsonFileStorageAdapter不能用来存储大量数据。我们将选择JsonFileStorageAdapter在我们的聊天机器人的构造函数中指定它。

|  |
| --- |
| bot = ChatBot(  *"Norman"*,  storage\_adapter=*"chatterbot.storage.JsonFileStorageAdapter"*,  database=*"./database.json"*) |

数据库参数用于指定的路径。在这个例子中，我们将调用数据库 database.json。database.json文件如果它不存在会被自动创建。JsonFileStorageAdapter是聊天机器人的默认适配器。如果你没有在你的构造函数指定适配器，该适配器JsonDatabase会自动使用。

* 输入和输出适配器

输入终端适配器只读取来自终端的用户的输入。输出端适配打印聊天机器人的反应。

|  |
| --- |
| bot = ChatBot(  *"Norman"*,  storage\_adapter=*"chatterbot.storage.JsonFileStorageAdapter"*,  input\_adapter=*"chatterbot.input.TerminalAdapter"*,  output\_adapter=*"chatterbot.output.TerminalAdapter"*,  database=*"./database.json"*) |

* 逻辑适配器

|  |
| --- |
| bot = ChatBot(  *"Norman"*,  storage\_adapter=*"chatterbot.storage.JsonFileStorageAdapter"*,  input\_adapter=*"chatterbot.input.TerminalAdapter"*,  output\_adapter=*"chatterbot.output.TerminalAdapter"*,  logic\_adapters=[  *"chatterbot.logic.MathematicalEvaluation"*,  *"chatterbot.logic.TimeLogicAdapter"*],  database=*"./database.json"*) |

该logic\_adapters参数是逻辑适配器的列表。在这个例子中，我们将使用两个逻辑适配器。TimeLogicAdapter当输入语句请求返回当前时间。该MathematicalEvaluation适配器解决使用基本操作的数学问题。

* 响应

运行一个while循环。当用户输入停止程序CTRL + C，特定的异常被触发，我们可以退出循环。

|  |
| --- |
| while True:  try:  bot\_input = bot.get\_response(None)  except(KeyboardInterrupt, EOFError, SystemExit):  break |

* 例子

更多例子见：<https://github.com/gunthercox/ChatterBot/tree/master/examples>

简单例子：

|  |
| --- |
| # -\*- coding: utf-8 -\*-  from chatterbot import ChatBot  # Create a new chat bot named Charlie  chatbot = ChatBot(*"Charlie"*)  # Get a response to the input "How are you?"  response = chatbot.get\_response(*"How are you?"*)  print(response) |

终端例子：

|  |
| --- |
| # -\*- coding: utf-8 -\*-  from chatterbot import ChatBot  import logging  # Uncomment the following line to enable verbose logging  # logging.basicConfig(level=logging.INFO)  # Create a new instance of a ChatBot  bot = ChatBot(*"Terminal"*,  storage\_adapter=*"chatterbot.storage.JsonFileStorageAdapter"*,  logic\_adapters=[  *"chatterbot.logic.MathematicalEvaluation"*,  *"chatterbot.logic.TimeLogicAdapter"*,  *"chatterbot.logic.BestMatch"*  ],  input\_adapter=*"chatterbot.input.TerminalAdapter"*,  output\_adapter=*"chatterbot.output.TerminalAdapter"*,  database=*"../database.db"*)  print(*"Type something to begin..."*)  # The following loop will execute each time the user enters input  while True:  try:  # We pass None to this method because the parameter  # is not used by the TerminalAdapter  bot\_input = bot.get\_response(None)  # Press ctrl-c or ctrl-d on the keyboard to exit  except (KeyboardInterrupt, EOFError, SystemExit):  break |

用户在控制台输入数学问题和时间问题（数学和时间的逻辑适配器）后回车，系统会精确匹配答案返回，见下图。

