

Package ‘IMWatson’

March 29, 2019

Type Package

Title Chat with Watson's Assistant API

Version 0.5.0

Author Ignacio Martinez

Maintainer Ignacio Martinez <ignacio@protonmail.com>

Description Chat with a chatbot created with the 'IBM Watson Assistant' <<https://www.ibm.com/cloud/watson-assistant/>>.

License GPL-3

Encoding UTF-8

LazyData true

Imports htrr, jsonlite, R6, assertive, magrittr, shiny, V8, shinyjs, glue, shinydashboard

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-29 16:40:03 UTC

R topics documented:

chatbot	2
chatbotUI	3
continue_chat	3
init	4
ShinyChatbot	5
watson_chat_factory	6

Index	7
--------------	----------

chatbot

chatbot Shiny server Module

Description

chatbot Shiny server Module

Usage

```
chatbot(input, output, session, api_key, workspace, rv.chat = FALSE)
```

Arguments

input	input variable to read the value from
output	output variable to read the value from
session	session variable to read the value from
api_key	api key to connect to your watson assistant
workspace	your watson assistant workspace
rv.chat	optional variable to pass your chatbot as a reactive variable

Details

A shiny server module for the chatbot

Examples

```
if(interactive()){  
  ## app.R ##  
  library(shiny)  
  library(shinydashboard)  
  library(IMWatson)  
  ui <- dashboardPage(  
    dashboardHeader(),  
    dashboardSidebar(),  
    dashboardBody(IMWatson::chatbotUI("watson"))  
  )  
  server <- function(input, output, session) {  
    callModule(IMWatson::chatbot, "watson", api_key = your_api_key,  
    workspace = your_workspace)  
  }  
  shinyApp(ui, server)  
}
```

chatbotUI

chatbotUI Shiny UI Module

Description

An user interface for your shiny watson assistant apps

Usage

```
chatbotUI(id)
```

Arguments

id namespace id

Examples

```
if(interactive()){  
  ### APP ###  
  library(shiny)  
  library(shinydashboard)  
  ui <- dashboardPage(  
    dashboardHeader(),  
    dashboardSidebar(),  
    dashboardBody(uiOutput("context"),  
                  IMWatson::chatbotUI("watson"))  
  )  
  server <- function(input, output, session) {  
    rv <- reactiveValues(rv.chat = FALSE)  
    bot <- reactive({  
      callModule(IMWatson::chatbox, "watson", texlog)  
    })  
    output$context <- renderUI(bot()$magic)  
  }  
  shinyApp(ui, server)  
}
```

continue_chat

Continue Chat with Watson

Description

Chat with Waston after initializing a connection

Usage

```
continue_chat(text, context, url, version, api_key, workspace)
```

Arguments

text	Whatever you want to say to Watson
context	context returned by your assistant
url	optional argument that defaults to "https://gateway.watsonplatform.net/assistant/api/v1/workspaces"
version	optional argument that defaults to "2018-09-20"
api_key	api key to connect to your watson assistant
workspace	Workspace api

Details

continue your conversation with your assistant

Value

a list with the response and context produced by our assistant

Examples

```
if(interactive()){
  connection <- init(api_key, workspace)
  connection[["text"]]
  chat1 <- chat("Ignacio", connection, api_key, workspace)
  chat1[["text"]]
  chat2 <- chat("I would like flower suggestions", chat1, api_key, workspace)
  chat2[["text"]]
}
```

init

Initialize connection with Watson's Conversation API

Description

Connect to Watson's Conversation API

Usage

```
init(api_key, workspace, url, version)
```

Arguments

api_key	API Key
workspace	Workspace api
url	optional argument that defaults to "https://gateway.watsonplatform.net/assistant/api/v1/workspaces"
version	optional argument that defaults to "2018-09-20"

Details

This function allows you to start a connection to your Watson Assistant

Value

The function returns a list with the text and the context that is returned by your assistant.

Examples

```
if(interactive()){  
# See https://cloud.ibm.com/apidocs/assistant  
connection <- init(api_key, workspace)  
connection[["text"]]  
chat1 <- chat("Ignacio", connection, username, password, workspace)  
chat1[["text"]]  
chat2 <- chat("I would like flower suggestions", chat1, username, password, workspace)  
chat2[["text"]]  
}
```

ShinyChatbot

Shiny Chatbot

Description

FUNCTION_DESCRIPTION

Usage

```
ShinyChatbot(api_key, workspace, title = "Chatbot powered by Watson")
```

Arguments

api_key	The API key
workspace	The workspace api for your chatbot
title	The title for the chatbox

See Also

[runApp](#)

Examples

```
if(interactive()){  
library(IMWatson)  
ShinyChatbot(api_key = your_api_key,  
workspace = your_workspace)  
}
```

watson_chat_factory *watson chat factory*

Description

watson chat factory

Usage

watson_chat_factory

Format

An object of class R6ClassGenerator of length 24.

Index

*Topic **datasets**

watson_chat_factory, [6](#)

chatbot, [2](#)

chatbotUI, [3](#)

continue_chat, [3](#)

init, [4](#)

runApp, [5](#)

ShinyChatbot, [5](#)

watson_chat_factory, [6](#)