

Package ‘gitGPT’

April 4, 2023

Type Package

Title Automated Git Commit Messages using the 'OpenAI' 'GPT' Model

Version 0.1.3

Imports httr, jsonlite

Maintainer Steve Condylis <steve.condylis@gmail.com>

BugReports <https://github.com/stevecondylis/gitGPT/issues>

License MIT + file LICENSE

URL <https://github.com/stevecondylis/gitGPT>

Description Automates the process of adding, committing, and pushing changes to a 'git' repository using commit messages generated by passing the git diff output to the 'OpenAI' 'GPT-3.5 Turbo' model (<<https://platform.openai.com/docs/models/gpt-3>>).

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation no

Author Steve Condylis [aut, cre]

Repository CRAN

Date/Publication 2023-04-04 19:40:02 UTC

R topics documented:

add_commit_push	2
commit	2
generate_commit_message	3
generate_encoded_git_diff_output	4
generate_git_diff_output	5
gitGPT	5
suggest_commit_message	6

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

add_commit_push	<i>Add, commit, and push changes to a Git repository</i>
-----------------	--

Description

Automates the process of adding changes to a Git repository, committing those changes with a commit message, and pushing the changes to a remote repository.

Usage

```
add_commit_push(commit_message, prepend)
```

Arguments

`commit_message` A character string containing the commit message for the changes.

`prepend` A character string to prepend to the commit message. Defaults to an empty string.

Value

No return value, called for side effects.

Examples

```
## Not run:  
  add_commit_push()  
  
## End(Not run)
```

commit	<i>Commit changes using an auto-generated commit message</i>
--------	--

Description

An R wrapper to automatically add, commit, and push changes to a git repository using an auto-generated commit message. The commit message is generated by an API call to the 'OpenAI' 'GPT-3.5 Turbo' model using the git diff output as input.

Usage

```
commit(commit_message, prepend)
```

Arguments

- commit_message A custom commit message. If not provided, a message will be generated automatically.
- prepend A string to prepend to the commit message. Defaults to an empty string. A convention may be to prepend 'GPT: ' to the beginning of the commit message so you can differentiate between those you wrote vs those GPT composed.

Value

No return value, called for side effects.

Examples

```
## Not run:  
# Simple command to git add, git commit, and git push with a commit  
# message based on git diff of the working directory  
commit()  
  
## End(Not run)
```

```
generate_commit_message  
    Generate commit message
```

Description

Generates a commit message by making an API call to the 'OpenAI' 'GPT-3.5 Turbo' model using the encoded git diff output as input.

Usage

```
generate_commit_message(encoded_git_diff_output)
```

Arguments

- encoded_git_diff_output
A character vector containing the URL-encoded git diff output.

Value

A character vector of length 1 containing the generated commit message.

Examples

```
## Not run:  
# Sends the encoded git diff to GPT and returns a  
# character vector containing the commit message:  
git_diff_output <- generate_git_diff_output()  
edo <- generate_encoded_git_diff_output(git_diff_output)  
enerate_commit_message(edo)  
  
## End(Not run)
```

```
generate_encoded_git_diff_output  
    Encode git diff output
```

Description

Takes the git diff output and encodes it for use in the API call to the 'OpenAI' 'GPT-3.5 Turbo' model.

Usage

```
generate_encoded_git_diff_output(git_diff_output)
```

Arguments

```
git_diff_output  
    A character vector containing the git diff output.
```

Value

A character vector containing the URL-encoded git diff output.

Examples

```
## Not run:  
git_diff_output <- generate_git_diff_output()  
generate_encoded_git_diff_output(git_diff_output)  
  
## End(Not run)
```

```
generate_git_diff_output
  Generate git diff output
```

Description

Returns the git diff output for the current working directory, and lists any new files found. It checks the system's OS and executes the appropriate command to generate the git diff.

Usage

```
generate_git_diff_output()
```

Value

A character vector containing the git diff output.

Examples

```
## Not run:
# View with cat() for easier reading
cat(generate_git_diff_output())

## End(Not run)
```

```
gitGPT          gitGPT: Automated Git Commit Messages Using the 'OpenAI' 'GPT-3.5 Turbo' Large Language Model
```

Description

An R wrapper that automates the process of adding, committing, and pushing changes to a git repository using auto-generated commit messages. Leverages the 'OpenAI' 'GPT-3.5 Turbo' model to generate meaningful commit messages based on the git diff output.

Functions

- `commit`: Automatically adds, commits, and pushes changes using an auto-generated or custom commit message.
- `generate_git_diff_output`: Returns the git diff output for the current working directory.
- `generate_encoded_git_diff_output`: Encodes the git diff output for use in the API call to the 'OpenAI' 'GPT-3.5 Turbo' model.
- `generate_commit_message`: Generates a commit message using an API call to the 'OpenAI' 'GPT-3.5 Turbo' model with the encoded git diff output.
- `add_commit_push`: Adds, commits, and pushes changes to a git repository using the provided commit message.

`suggest_commit_message`*Suggest a commit message based on the provided git diff*

Description

Suggests a commit message by utilizing the 'OpenAI' 'GPT-3.5 Turbo' model. It takes a git diff as input and returns a meaningful commit message.

Usage

```
suggest_commit_message(diff)
```

Arguments

<code>diff</code>	An optional character vector containing the git diff. If not provided, the function will automatically generate the git diff output for the current working directory.
-------------------	--

Value

A character vector of length 1 with the suggested commit message based on the provided git diff.

Examples

```
## Not run:  
  suggest_commit_message()  
  
## End(Not run)
```

Index

`add_commit_push`, [2](#), [5](#)

`commit`, [2](#), [5](#)

`generate_commit_message`, [3](#), [5](#)

`generate_encoded_git_diff_output`, [4](#), [5](#)

`generate_git_diff_output`, [5](#), [5](#)

`gitGPT`, [5](#)

`suggest_commit_message`, [6](#)