
Flipgenic

Daniel Thwaites

Jul 04, 2021

CONTENTS

1 Documentation Contents	1
1.1 Quickstart	1
1.2 flipgenic	2
Python Module Index	5
Index	7

DOCUMENTATION CONTENTS

1.1 Quickstart

1.1.1 Installation

1. Install Flipgenic from PyPI:

```
python -m pip install flipgenic
```

2. Download the spaCy model:

```
python -m spacy download en_core_web_md
```

1.1.2 Database Setup

Create an instance of `Responder`:

```
from flipgenic import Responder
responder = Responder('/path/to/storage/directory/')
```

Several files will be created in the given directory, which is also created if it does not exist.

1.1.3 Learning Responses

Initially, a `Responder` has no knowledge of how to communicate. You need to provide example replies which are reused in future conversations.

Responses are taught in pairs:

```
responder.learn_response('Hello', 'Hi')
```

Our `Responder` now knows that “Hi” is a suitable reply to “Hello”.

Batches

Multiple pairs can be added as a batch before saving:

```
responder.add_response('Hello', 'Hi')
responder.add_response('Hello', 'How are you?')
responder.add_response('How are you?', 'Fine, thanks!')
responder.commit_responses()
```

This is much more efficient when you are importing a large dataset. Data is held in memory until `commit_responses` is called.

1.1.4 Getting a Response

A response can be recalled as follows:

```
response, distance = responder.get_response('Hello')
print(response.text)
```

`distance` increases for uncertain replies. The lower the distance, the closer the match, and therefore responses with a lower distance are more likely to make sense.

Exact matches have a distance of 0. If no responses were available, the distance will be infinite. There is no upper limit on distances which can be returned, however values over 10 are unlikely for a populated database.

1.2 flipgenic

1.2.1 flipgenic package

```
class flipgenic.Message(text: str, metadata: Optional[str] = None)
```

Bases: object

A piece of text.

Parameters

- **text** – The text of the message.
- **metadata** – An arbitrary string which can be retrieved when the message is chosen as a response.

```
metadata: str = None
```

```
text: str
```

```
class flipgenic.Responder(data_directory, spacy_model='en_core_web_md')
```

Bases: object

Holds a database connection and handles learning and producing responses.

Parameters

- **data_directory** – Path to a folder which holds files for this Responder. Will be created if it doesn't exist.
- **model** – SpaCy model, or the name of one to be loaded.

add_response (*prompt, message*)

Process a response pair without saving it immediately.

The response won't be available as an output from `get_response` until `commit_responses` is called.

Parameters

- **prompt** – The text this is in response to.
- **message** – The response to be learned. This can be a simple string, or an instance of `Message` if you would like to include metadata.

commit_responses ()

Save any responses which have not yet been written to the database.

get_all_responses (*text*)

Return all relevant responses to a prompt along with their distance.

The distance returned gets closer to 0 the more confident the response is. There is no hard limit to how large the distance can be, however distances tend to range between 0 and 10.

Note that this only returns messages which are linked to the same vector, so their distances are all the same.

Returns (list of `Message` instances, distance)

get_response (*text*)

Return the most confident response to a prompt.

The distance returned gets closer to 0 the more confident the response is. There is no hard limit to how large the distance can be, however distances tend to range between 0 and 10.

Parameters **text** – The prompt to respond to.

Returns Tuple of (message, distance).

learn_response (**args, **kwargs*)

Add a response pair and save it immediately.

Takes the same parameters as `add_response`.

Other unsaved responses will also be committed when you call this method.

Submodules**flipgenic.db_models module**

```
class flipgenic.db_models.Response (**kwargs)
    Bases: sqlalchemy.orm.decl_api.Base
    id
    meta
    ngt_id
    response
```


PYTHON MODULE INDEX

f

`flipgenic`, 2

`flipgenic.db_models`, 3

A

`add_response()` (*flipgenic.Responder* method), 2

C

`commit_responses()` (*flipgenic.Responder* method), 3

F

`flipgenic`
module, 2

`flipgenic.db_models`
module, 3

G

`get_all_responses()` (*flipgenic.Responder* method), 3

`get_response()` (*flipgenic.Responder* method), 3

I

`id` (*flipgenic.db_models.Response* attribute), 3

L

`learn_response()` (*flipgenic.Responder* method), 3

M

`Message` (class in *flipgenic*), 2

`meta` (*flipgenic.db_models.Response* attribute), 3

`metadata` (*flipgenic.Message* attribute), 2

module

`flipgenic`, 2

`flipgenic.db_models`, 3

N

`ngt_id` (*flipgenic.db_models.Response* attribute), 3

R

`Responder` (class in *flipgenic*), 2

`Response` (class in *flipgenic.db_models*), 3

`response` (*flipgenic.db_models.Response* attribute), 3

T

`text` (*flipgenic.Message* attribute), 2