eflatun uav

Release 0.0.2

Muhammed Sezer

May 16, 2023

CONTENTS:

CHAPTER

ONE

EFLATUN_UAV

Modules

eflatun_uav.filters	Filter implementations for moving objects
eflatun_uav.helpers	

1.1 eflatun_uav.filters

Filter implementations for moving objects

Classes

BaseFilter(input_size, output_size)	Base Filter object for Filter module

class eflatun_uav.filters.**BaseFilter**(*input_size: List*, *output_size: List*)

Bases: `object`

Base Filter object for Filter module

__init__(*input_size: List*, *output_size: List*) → None

summary

Parameters

- **input_size** (*List*) – _description_
- **output_size** (*List*) – _description_

predict() → ndarray

summary

Raises

NotImplementedError – _description_

Returns

description

Return type

np.ndarray

update(*input_state: ndarray*)

summary

Parameters

input_state (*np.ndarray*) – _description_

Raises

NotImplementedError – _description_

1.2 eflatun_uav.helpers

Modules

eflatun_uav.helpers.number_generators	This module creates numbers for given variable type of inputs

1.2.1 eflatun_uav.helpers.number_generators

This module creates numbers for given variable type of inputs

Functions

convert_string_to_float(string)	Converts a string to a deterministic random float representation between 0 and 1.
convert_string_to_int(string, *[, base])	Converts a string to an deterministicly random integer representation using the specified base.

eflatun_uav.helpers.number_generators.**convert_string_to_float**(*string: str*) → float

Converts a string to a deterministic random float representation between 0 and 1.

Works better for texts longer than 5 letters.

Parameters

string (*str*) – The input string to be converted to a float.

Returns

The float representation of the input string between 0 and 1.

Return type

float

Example

```
>>> convert_string_to_float("Hello, World")
0.3350260018341942
>>> convert_string_to_float("Hi, World?")
0.8893743173684925
>>> convert_string_to_float("Hi, World")
0.03764671504177386
```

eflatun_uav.helpers.number_generators.**convert_string_to_int**(*string: str*, *, *base: int | None = 256*) → int

Converts a string to an deterministicly random integer representation using the specified base.

Works better for texts longer than 5 letters.

Parameters

- **string** (*str*) – The input string to be converted to an integer.
- **base** (*Optional[int], optional*) – The base to be used for the conversion. Defaults to 256.

Raises

ValueError – If the base is not an integer or if it is 0, -1, or 1.

Returns

The integer representation of the input string.

Return type

int

Example

```
>>> convert_string_to_int("Hello, World!")
157
>>> convert_string_to_int("Hello, World")
84
>>> convert_string_to_int("Hello, World!", base = 36)
13
```

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX