Montag, 12. Oktober 2020

Developer documentation

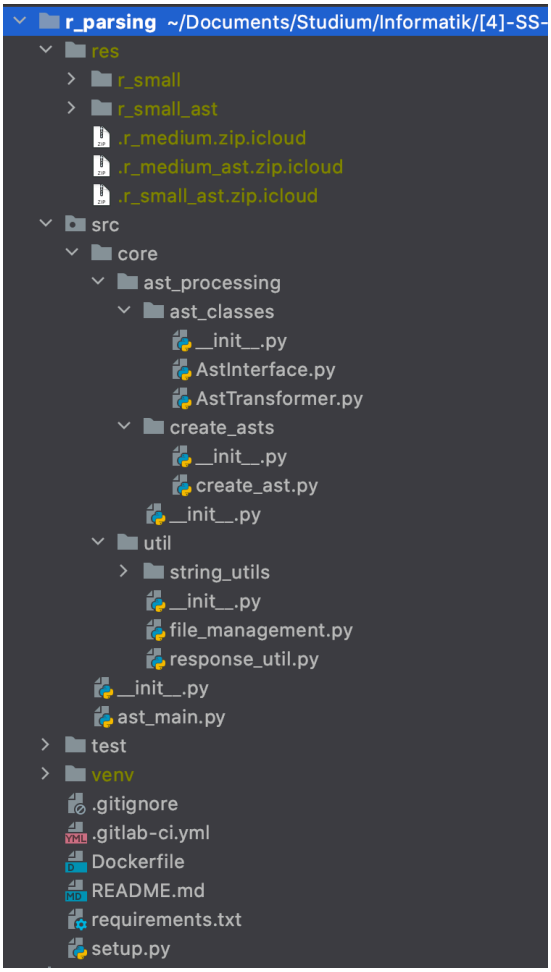
Author: Joshua Kraft

Summary:

- Name: r_parsing
- Description: parse and extract ast's out of python code
- Loc: 792 (src/test, 316/273, 55/45%)
- Testcoverage: 78%
- Style: 0 found PIP8 warnings

Repository structure:

- Source/Test/Resources split
- Source and Test share the same structure
- Source includes all runnable code on the highest level including all main() files
- Source/core/ast_ processing includes the main classes in ast_classes which are used in create_ast
- Res contains the used and created dataset
- It adds a _ast to the original dataset
- The data is organized in directories for each GitHub-repo of R-code
- Some of the paths in ast_main are hardcoded and there need to be changed if the location an name is updated on the computer
- Preview ——————————————>



Code description:

This Parser takes a repository of Github-Repositories of R-files.

In the first step creat_ast creates an ast for each source file with a rpy2 implementation.

In the next step the ast is used to create an AstTransformer object which can operate on this ast.

The AstTransformer class is then able to find all „Function“-nodes in this tree and crop the subtree for further processing. One ast can have multiple functions and therefor subtrees.

In the process the code makes sure that every function subtree is on the highest tree lvl and a complete subtree.

Then all of the Function subtrees are saved in an .json file in the same structure as the original data

Main classes/structures:

Ast: the ast is a pandas data frame and represents a tree. Each row is a node with all necessary information and the parent node. This structure is defined by the main r interpreter and can’t be changed.

AstTransformer: takes ast as the constructor input. This class is programmed to operate on the ast data structure. It can access all information like terminal, value etc. and also find children and parent nodes as well as Function nodes.

To be continued:

This project is complete and tested on correctness.