CCEvovis manual

1. Overview

The purpose of this system is to analyze and visualize the evolution of code clones between multiple versions and provide them to developers. CCEvovis can analyze the following two projects.

(1) Local project

In order to analyze, it is necessary to collect multiple versions of a project in the same folder and change the folder name to the version date.

(2) Git remote repository

CCEvovis starts analysis automatically when you provide the URL, branch, analysis period, and analysis interval of the remote repository published.

2. Requirements

Tested to work on the following platforms

Windows10

The following environment is required to execute the system.

- Java 1.8
- Git (if targeting a remote repository)
- Python 2.6 32bit (if using CCFinderX)

3. Configuration

This system consists of the following two tools.

- setting.jar (Main class: “CCEvovsiSetter¥src ¥Main.java”)
  Configuration tool for CCEvovis. You can create a configuration file. By referring to git_example and local_example, you can also create and edit a configuration file with your own editor. Japanese Only.
- analyze.jar (Main class: “src¥cn¥Main.java”)
  Jar file of the CCEvovis body. The ccm.bat is a script that executes analyze.jar. Change the arguments in the configuration file and execute.

4. How to run

4.1. The main flow of execution

(1) Start setting.jar and input various setting items related to the project to be analyzed. Alternatively, you can also create and edit a configuration file using your own editor.

(2) Edit the ccm.bat and give the analyze.jar the configuration file generated in step (1) as an argument.

(3) Execute ccm.bat

(4) The analysis result is located under “¥users¥user name (default: guest)¥” of the folder specified as the output destination.

4.2.Configuration File Items

4.2.1. Common Settings

- PROJECT_NAME:
  Project name.

- ANALYSIS_NAME:
  Detailed name of the analysis.

- TOOL:
  Set one of the following three tools.
  (1) SourcererCC
  A token-based tool that can quickly detect code clones with similar semantics in large-scale software.
  (2) CCFinderX
  A lexical analysis-based tool that can detect syntactically matched code clones.
  (3) CCVolti
  A vector-based tool that can quickly detect block-level code clones with similar semantics in large-scale software.

- LANGUAGE:
  Programming language of the target project.

- TOKEN:
  Set the minimum number of tokens of the code fragment to be detected.
  Default is 50.

- HTML_DIR:
  Set output destination folder for analysis results.

- CSV:
  Set to true if you want to see the results in CSV output.

- CSV_DIR:
  Set output destination of CSV result.

- WORK_DIR
  Set work directory. Default is "~ ¥ CCEvovis ¥ file".

- USER_ID:
  Set user name. Default is "guest".

4.2.2. When analyzing a Git remote repository

Create a configuration file referring to git_example.

- GIT_DIRECT:
  Set true if you want to detect Git remote repositories. (LOCAL_TARGET set to false)
  Otherwise, set false. (LOCAL_TARGET set to true)

- GIT_BRANCH:
  Set the branch name. ex) master

- TARGET_DIR:
  Set the download destination of the repository.

- START_DATE:
  Set an analysis start date. For example, for May 21, 2018, enter 20180521.

- END_DATE:
  Set an analysis end date.

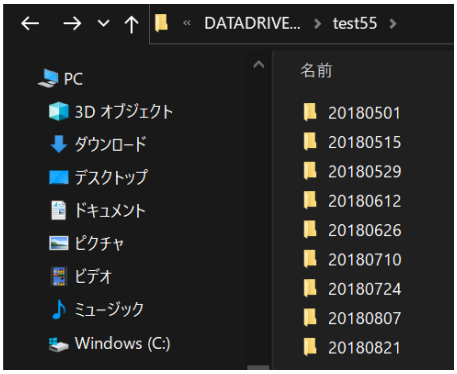
- INTERVAL:
  Set the analysis interval. Enter 7 if you want to analyze at weekly intervals. The analysis starts weekly with the input analysis start date and end date.

4.2.3. When analyzing local projects

Create a configuration file referring to local_example.

- LOCAL_TARGET:
  Set true if you want to detect local project (GIT_DIRECT set to false)
  Otherwise, set false. (GIT_DIRECT set to true)

- TARGET_DIR: E:
  Set the folder containing the project to be analyzed. All folders here are analyzed. The folder name cannot be analyzed unless it is a date. For example, for the version on September 10, 2019, it is necessary to use the folder name "20190910". See the image below.



5. 注意点

- The screen drawing of setting.jar may fail depending on the execution environment. In that case, try changing the screen size from the display settings.
- Project Some projects cannot be analyzed depending on the detection tool. I recommend SourcererCC and CCVolti because their detection is relatively stable.
- Log Logs related to analysis are output to ccm.log.
- The number of versions that can be analyzed at one time is set to 30.

6. Contact

   If you have any bug reports or questions, feel free to contact us below.

   Mail：hhman321@gmail.com

   Skype：live:f7aa671626ff244f