

Lab Assignment 2: Introduction to Mining Software Repositories for Bug Fixes in the Wild

INTRODUCTION, SETUP, AND TOOLS

Introduction

In this lab, we explore the process of mining bug-fix pairs from GitHub repositories using the **MineCPP** tool. The primary goal is to extract and analyze bug fixes in real-world projects, enabling us to better understand patterns in software development and contribute to improving maintainability in open-source projects.

Setup

- **minecpp Installation:** we need to install **minecpp** on the local machine before starting lab.
- **Pythin Installation:** We need to have Python 3.10.

Tools

- **minecpp:** Used to extract and analyze bug fixes in real-world projects.

METHODOLOGY AND EXECUTION WITH RESULT AND ANALYSIS

Software Tool Setup:

Software Tool Setup:

Download, install, and configure minecpp using command given on [MineCPP Github](#) [1]

```
pip install minecpp
```

However, then we get error indicating version conflicts between in some dependencies

```

Activities Terminal Jan 27 17:23
student@vlsi:~$ pip install minecpp
Collecting minecpp
  Using cached Minecpp-0.4-py3-none-any.whl (2.4 MB)
Collecting PyDriller==2.6
  Using cached PyDriller-2.6-py3-none-any.whl (33 kB)
Collecting pandas==2.8.3
  Using cached pandas-2.8.3-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (12.4 MB)
Requirement already satisfied: matplotlib in ./local/lib/python3.8/site-packages (from minecpp) (3.7.5)
Collecting tree-sitter==0.20.4
  Using cached tree-sitter-0.20.4-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (492 kB)
Collecting crystalbleu==0.1
  Using cached crystalbleu-0.1-py3-none-any.whl (12 kB)
Collecting code-bert-score==0.4.1
  Using cached code-bert-score-0.4.1.tar.gz (20 kB)
Collecting transformers==4.36.0
  Downloading transformers-4.36.0-py3-none-any.whl (8.2 MB)
Collecting flask
  Using cached flask-2.0.3-py3-none-any.whl (101 kB)
Collecting nltk==3.8.1
  Using cached nltk-3.8.1-py3-none-any.whl (1.5 MB)
Collecting chardet==4.0.0
  Downloading chardet-4.0.0-py2.py3-none-any.whl (178 kB)
Collecting torch==2.1.2
  Downloading torch-2.1.2-cp38-cp38-manylinux1_x86_64.whl (670.2 MB)
Requirement already satisfied: tqdm in /usr/lib/python3/dist-packages (from minecpp) (4.30.0)
Collecting lizard==1.17.10
  Downloading lizard-1.17.10-py2.py3-none-any.whl (66 kB)
Collecting types-pytz
  Downloading types-pytz-2024.2.0.20241221-py3-none-any.whl (10 kB)
Requirement already satisfied: pytz in /usr/lib/python3/dist-packages (from PyDriller==2.6->minecpp) (2019.3)
Collecting gipython
  Downloading GIPython-3.1.44-py3-none-any.whl (207 kB)
Collecting python-dateutil==2.8.2
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Requirement already satisfied: numpy==1.20.3; python_version < "3.10" in ./local/lib/python3.8/site-packages (from pandas==2.0.3->minecpp) (1.24.4)
Collecting tzdata==2022.1
  Downloading tzdata-2025.1-py2.py3-none-any.whl (346 kB)
Requirement already satisfied: pyparsing==2.3.1 in ./local/lib/python3.8/site-packages (from matplotlib->minecpp) (3.1.2)
Requirement already satisfied: packaging==20.0 in ./local/lib/python3.8/site-packages (from matplotlib->minecpp) (24.1)
Requirement already satisfied: MarkupSafe==2.1.1 in ./local/lib/python3.8/site-packages (from matplotlib->minecpp) (2.1.1)
Requirement already satisfied: importlib-resources==2.2.0 in ./local/lib/python3.8/site-packages (from matplotlib->minecpp) (4.53.0)
Requirement already satisfied: importlib-metadata==3.10.0; python_version < "3.10" in ./local/lib/python3.8/site-packages (from matplotlib->minecpp) (4.10.0)

Activities Terminal Jan 27 17:22
student@vlsi:~$ 
Collecting nvidia-cublas-cu12==12.1.3.1; platform_system == "Linux" and platform_machine == "x86_64"
  Downloading nvidia_cublas_cu12-12.1.3.1-py3-none-manylinux1_x86_64.whl (410.6 MB)
Collecting nvidia-cusparse-cu12==12.1.0.106; platform_system == "Linux" and platform_machine == "x86_64"
  Downloading nvidia_cusparse_cu12-12.1.0.106-py3-none-manylinux1_x86_64.whl (196.0 MB)
Collecting nvidia-curand-cu12==10.3.2.106; platform_system == "Linux" and platform_machine == "x86_64"
  Downloading nvidia_curand_cu12-10.3.2.106-py3-none-manylinux1_x86_64.whl (56.5 MB)
Collecting nvidia-cufft-cu12==11.0.2.54; platform_system == "Linux" and platform_machine == "x86_64"
  Downloading nvidia_cufft_cu12-11.0.2.54-py3-none-manylinux1_x86_64.whl (121.6 MB)
Collecting nvidia-cusolver-cu12==11.4.5.107; platform_system == "Linux" and platform_machine == "x86_64"
  Downloading nvidia_cusolver_cu12-11.4.5.107-py3-none-manylinux1_x86_64.whl (124.2 MB)
Collecting glibdb<5,>=4.0.1
  Downloading glibdb-4.0.12-py3-none-any.whl (62 kB)
Requirement already satisfied: six<1.15.0 in /usr/lib/python3/dist-packages (from python-dateutil==2.8.2->pandas==2.0.3->minecpp) (1.14.0)
Requirement already satisfied: zipp>=3.1.0; python_version < "3.10" in ./local/lib/python3.8/site-packages (from importlib-resources==2.2.0; python_version < "3.10"->matplotlib->minecpp) (3.19.2)
Collecting MarkupSafe==2.1.1
  Downloading MarkupSafe-2.1.5-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (26 kB)
Collecting mpmath<1.4,>=1.1.0
  Downloading mpmath-1.3.0-py3-none-any.whl (536 kB)
Collecting nvidia-nvjiptlnk-cu12
  Downloading nvidia_nvjiptlnk_cu12-12.8.61-py3-none-manylinux2018_x86_64.manylinux_2_12_x86_64.whl (39.2 MB)
Collecting smpmap<0,>=3.0.1
  Building wheel for smpmap (py3-none-any.whl (24 kB))
Building wheels for collected packages: code-bert-score
  Building wheel for code-bert-score: setup.py ... done
  Created wheel for code-bert-score: filename=code_bert_score-0.4.1-py3-none-any.whl size=22080 sha256=fb9b99c9ab0831f5aea7f89e1dedc86460782999d30aab8d343a58abf42e5
  Stored in directory: /home/student/.cache/pip/wheels/3d/6a/61/fdbb1c78198c2199cc183195a47fcfeffed45d443e6c8382
Successfully built code-bert-score
ERROR: importlib-metadata 8.5.0 has requirement zipp>=3.20, but you'll have zipp 3.19.2 which is incompatible.
ERROR: pandas 2.0.3 has requirement pytz>=2020.1, but you'll have pytz 2019.3 which is incompatible.
ERROR: code-bert-score 6.4.1 has requirement tensorflow<2.11.0, but you'll have tensorflow 2.10.0 which is incompatible.
ERROR: code-bert-score 6.4.1 has requirement tensorflow<2.11.0, but you'll have tensorflow 2.10.0 which is incompatible.
Installing collected packages: lizard, types-pytz, smpmap, glibdb, gipython, PyDriller, python-dateutil, tzdata, nvidia-cublas-cu12, nvidia-cusparse-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cusolver-cu12, nvidia-cuda-nvrtc-cu12, nvidia-cuda-cuda-cupti-cu12, nvidia-cuda-runtime-cu12, nvidia-cublas-cu12, nvidia-cudnn-cu12, nvidia-nvjiptlnk-cu12, nvidia-smpmap-cu12, nvidia-torch-cu12, nvidia-cuda-nvrtc-cu12, nvidia-cuda-cuda-cupti-cu12, nvidia-cuda-runtime-cu12, nvidia-cublas-cu12, nvidia-cudnn-cu12, nvidia-nvjiptlnk-cu12, nvidia-cusparse-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cuda-nvrtc-cu12-1.1.105, nvidia-cuda-runtime-cu12-1.2.1.105, nvidia-cudnn-cu12-8.5.0, nvidia-cuda-cuda-cupti-cu12-1.3.1.105, nvidia-cuda-nvrtc-cu12-1.1.105, nvidia-cuda-runtime-cu12-1.2.1.105, nvidia-cudnn-cu12-8.9.2.26, nvidia-cufft-cu12-11.0.2.54, nvidia-curand-cu12-10.3.2.106, nvidia-cusolver-cu12-11.4.5.107, nvidia-cusparse-cu12-12.10.106, nvidia-cublas-cu12-2.18.1, nvidia-nvjiptlnk-cu12-12.8.61, nvidia-nvtx-cu12-12.1.105, pandas-2.0.3 python-dateutil-2.9.0.post0 regex-2024.11.6 safetensors-0.5.2 smpmap-5.0.2 sympy-1.13.3 tokenizers-0.15.2 torch-2.1.2 transformers-4.36.0 tree-sitter-0.20.4 triton-2.1.0 types-pytz-2024.2.6.20241221 typing-extensions-4.12.2 tzdata-2025.1
student@vlsi:~$ 

```

ERROR: importlib-metadata 8.5.0 has requirement zipp>=3.20, but you'll have zipp 3.19.2 which is incompatible.

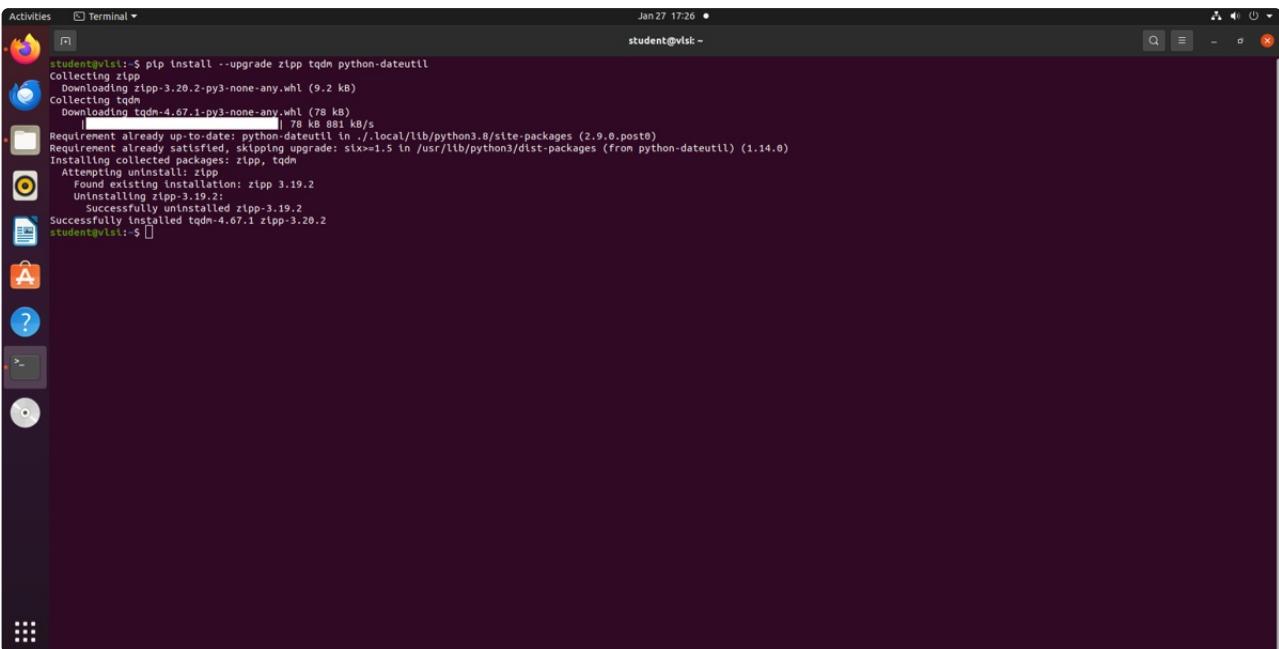
ERROR: pandas 2.0.3 has requirement pytz>=2020.1, but you'll have pytz 2019.3 which is incompatible.

ERROR: hugoingface-hub 6.27.1 has requirement tqdm>=4.42.1, but you'll have tqdm 4.38.8 which is incompatible.

ERROR: code-bert-score 6.4.1 has requirement tqdm>=4.31.1, but you'll have tqdm 4.38.8 which is incompatible.

to surpass this dependencies issue: we used following command to upgrade the required packages mention in the error (**zipp, tqdm, python-dateutil**)

```
pip install --upgrade zipp tqdm python-dateutil
```



```
student@vlst:~$ pip install --upgrade zipp tqdm python-dateutil
Collecting zipp
  Downloading zipp-3.20.2-py3-none-any.whl (9.2 kB)
    collecting tqdm
      Downloading tqdm-4.67.1-py3-none-any.whl (78 kB)
        Requirement already up-to-date: python-dateutil in ./local/lib/python3.8/site-packages (2.9.0.post0)
Requirement already satisfied, skipping upgrade: six>=1.5 in /usr/lib/python3/dist-packages (from python-dateutil) (1.14.0)
Installing collected packages: zipp, tqdm
  Attempting uninstall: zipp
    Found existing installation: zipp 3.19.2
    Uninstalling zipp-3.19.2:
      Successfully uninstalled zipp-3.19.2
Successfully installed tqdm-4.67.1 zipp-3.20.2
student@vlst:~$
```

After upgrading the dependencies, we retried the installation of **MineCPP** with the command mentioned in **MineCPP Github [1]**. However, this time we dont get error but during lab I remember we have got new error. stating we need to upgrade **pytz**.

```
ERROR: pandas 2.0.3 requires pytz>=2020.1, but you have pytz 2019.3, which is incompatible.
```

so we use

```
pip install --upgrade pytz
```

Repository Selection:

For the analysis I selected the repository [python-patcher\[2\]](#). This is a Python-based project that provides a tool for patching executable files. It fits the criteria of a medium-to-large scale open-source repository.

Define Selection Criteria:

to establish our own criteria for selection (inclusion/exclusion) of repositories I use the SEART GitHub Search Engine [3] and established the following **selection criteria**:

Search:

In search I added python so that I get repository which are working either with python or on python.

Number of Commits:

The repository must have commits between **1000 and 5000**. This will indicates that the repository is actively developed and also not too large to make analysis difficult.

Number of Stars:

100 stars. This is to ensure that the project is recognized by the GitHub community as a valuable or well-regarded open-source project.

And selected [python-patcher](#)[2] repository as first preference. the following image shows the process and output.

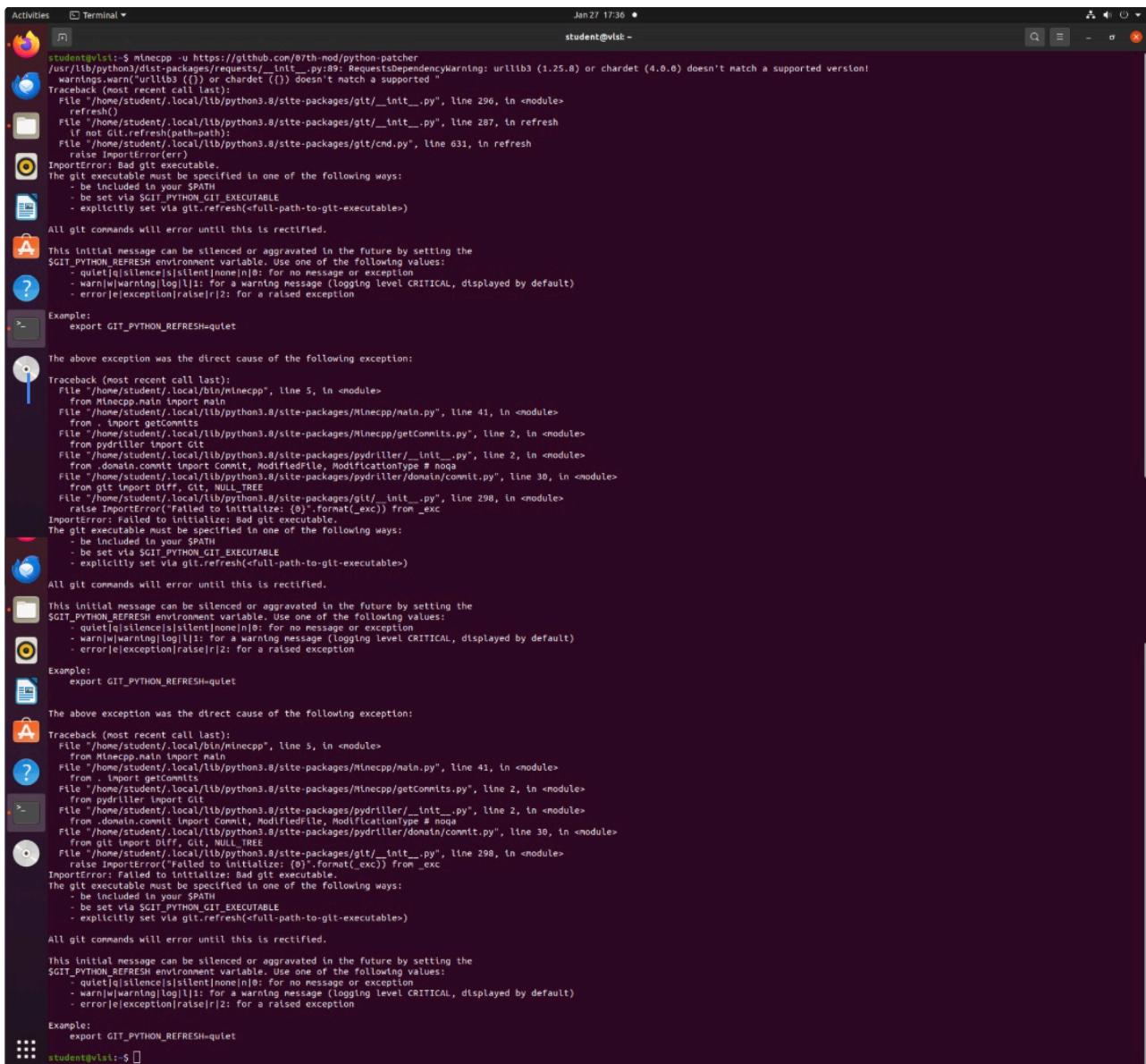
The screenshot shows a search interface for GitHub repositories. At the top, there's a search bar with 'python' and filters for 'License' (Has topic). Below that are sections for 'History and Activity' (Number of Commits, Issues, Branches, Pull Requests, Releases) and 'Popularity Filters' (Number of Stars, Watchers, Forks). On the right, there are 'Date-based Filters' (Created Between, Last Commit Between), 'Size of codebase' (Non Blank Lines, Code Lines, Comment Lines), and 'Additional Filters' (Sorting, Repository Characteristics like Exclude Forks, Has License, Has Open Issues, Has Wiki, Has Pull Requests). A 'Search' button is at the bottom of the filter section. The main area displays search results for 'Results: 338'. It lists two repositories: '07th-mod/python-patcher' (JS, 1505 commits, 11 forks) and '20tab/unrealenginepython' (C, 2248 commits, 570 forks). Each repository card shows metrics like Watchers, Stars, Forks, Branches, Releases, Last Push, and Size.

Run the Software Tool on Selected Repositories:

to Execute minecpp on the selected repositories I used the command given in MineCPP Github [1].

```
minecpp -u https://github.com/07th-mod/python-patcher
```

However, this got to in the following error message:



```
student@vls1:~$ minecpp -u https://github.com/07th-nod/python-patcher
/usr/lib/python3/dist-packages/requests/_int_.py:89: RequestsDependencyWarning: urllib3 (1.25.8) or chardet (4.0.0) doesn't match a supported version!
  warnings.warn("urllib3 ({}) or chardet ({}) doesn't match a supported "
Traceback (most recent call last):
  File "/home/student/.local/lib/python3.8/site-packages/git/_init__.py", line 290, in <module>
    from . import _git
  File "/home/student/.local/lib/python3.8/site-packages/git/_init__.py", line 287, in refresh
    if not Git.refresh(path=path):
  File "/home/student/.local/lib/python3.8/site-packages/git/cmd.py", line 631, in refresh
    raise ImportError(err)
ImportError: Bad git executable.
The git executable must be specified in one of the following ways:
- be included in your $PATH
- be set via $GIT_PYTHON_GIT_EXECUTABLE
- explicitly set via git.refresh(<full-path-to-git-executable>)

All git commands will error until this is rectified.

[A] This initial message can be silenced or aggravated in the future by setting the
$GIT_PYTHON_REFRESH environment variable. Use one of the following values:
- quiet|silence|silent|none|0: for no message or exception
- warn|warning|log|1: for a warning message (logging level CRITICAL, displayed by default)
- error|exception|raise|r2: for a raised exception

Example:
export GIT_PYTHON_REFRESH=quiet

The above exception was the direct cause of the following exception:

Traceback (most recent call last):
  File "/home/student/.local/bin/minecpp", line 5, in <module>
    from Minecpp.main import main
  File "/home/student/.local/lib/python3.8/site-packages/Minecpp/main.py", line 41, in <module>
    from . import getcommits
  File "/home/student/.local/lib/python3.8/site-packages/Minecpp/getCommits.py", line 2, in <module>
    from pydriller import Git
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/_init__.py", line 2, in <module>
    from .domain.commit import Commit, ModifiedFile, ModificationType # noqa
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/domain/commit.py", line 30, in <module>
    from git import Diff, Git, NULL_TREE
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/_init__.py", line 290, in <module>
    raise ImportError("Failed to initialize: {}").format(_exc) from _exc
ImportError: Failed to initialize: Bad git executable.
The git executable must be specified in one of the following ways:
- be included in your $PATH
- be set via $GIT_PYTHON_GIT_EXECUTABLE
- explicitly set via git.refresh(<full-path-to-git-executable>)

All git commands will error until this is rectified.

[B] This initial message can be silenced or aggravated in the future by setting the
$GIT_PYTHON_REFRESH environment variable. Use one of the following values:
- quiet|silence|silent|none|0: for no message or exception
- warn|warning|log|1: for a warning message (logging level CRITICAL, displayed by default)
- error|exception|raise|r2: for a raised exception

Example:
export GIT_PYTHON_REFRESH=quiet

The above exception was the direct cause of the following exception:

Traceback (most recent call last):
  File "/home/student/.local/bin/minecpp", line 5, in <module>
    from Minecpp.main import main
  File "/home/student/.local/lib/python3.8/site-packages/Minecpp/main.py", line 41, in <module>
    from . import getcommits
  File "/home/student/.local/lib/python3.8/site-packages/Minecpp/getCommits.py", line 2, in <module>
    from pydriller import Git
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/_init__.py", line 2, in <module>
    from .domain.commit import Commit, ModifiedFile, ModificationType # noqa
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/domain/commit.py", line 30, in <module>
    from git import Diff, Git, NULL_TREE
  File "/home/student/.local/lib/python3.8/site-packages/pydriller/_init__.py", line 290, in <module>
    raise ImportError("Failed to initialize: {}").format(_exc) from _exc
ImportError: Failed to initialize: Bad git executable.
The git executable must be specified in one of the following ways:
- be included in your $PATH
- be set via $GIT_PYTHON_GIT_EXECUTABLE
- explicitly set via git.refresh(<full-path-to-git-executable>)

All git commands will error until this is rectified.

[C] This initial message can be silenced or aggravated in the future by setting the
$GIT_PYTHON_REFRESH environment variable. Use one of the following values:
- quiet|silence|silent|none|0: for no message or exception
- warn|warning|log|1: for a warning message (logging level CRITICAL, displayed by default)
- error|exception|raise|r2: for a raised exception

Example:
export GIT_PYTHON_REFRESH=quiet
student@vls1:~$
```

It appears to be error related to Git. So I thought it could be same as git is not install properly or dependency issue like we were getting previously.

So, I ran the following command to check if Git was installed properly.

```
git --version
```

And found that git was not installed in the system so used the command given by the terminal to install git

```
sudo apt install git
```

```
Activities Terminal • Jan 27 17:37 •
student@vist: ~
student@vist:~$ Command 'git' not found, but can be installed with:
student@vist:~$ sudo apt install git
student@vist:~$ Command 'git' not found, but can be installed with:
student@vist:~$ sudo apt install git
student@vist:~$ Command 'git' not found, but can be installed with:
student@vist:~$ sudo apt install git
student@vist:~$ sudo apt install git
[sudo] password for student:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libfwupdplugin libxml2
Use 'sudo apt autoremove' to remove them.
You might also want to run 'sudo apt --fix-broken install' to correct these.
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
  git git-man liberror-perl
0 upgraded, 0 newly installed, 0 to remove and 88 not upgraded.
Need to get 5,525 kB of archives.
After this operation, 38.8 MB of additional disk space will be used.
Do you want to continue [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu focal/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 git-man all 1:2.25.1-1ubuntu3.13 [887 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 git amd64 1:2.25.1-1ubuntu3.13 [4,612 kB]
Fetched 4,304 kB in 35s (112 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 278569 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1:2.25.1-1ubuntu3.13_all.deb ...
Unpacking git-man (1:2.25.1-1ubuntu3.13) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1:2.25.1-1ubuntu3.13_amd64.deb ...
Unpacking git (1:2.25.1-1ubuntu3.13) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git-man (1:2.25.1-1ubuntu3.13) ...
Setting up git (1:2.25.1-1ubuntu3.13) ...
Processing triggers for man-db (2.9.1-1) ...
student@vist:~$
```

After installing the git, I retried running **MineCPP** on selected Repositories and it successfully completed.
(Please refer to the first and last screenshots for the command prompt)

```
Activities Terminal Jan 27 19:57 ● student@vlsi: ~
Setting up git-man (1:2.25.1-1ubuntu3.13) ...
Setting up git (1:2.25.1-1ubuntu3.13) ...
Processing triggers for man-db (2.9.1-1) ...
student@vlsi:~$ minecpp -u https://github.com/07th-mod/python-patcher
/usr/lib/python3/dist-packages/requests/_int.py:89: RequestsDependencyWarning: urllib3 (1.25.8) or chardet (4.0.0) doesn't match a supported version!
  warnings.warn("urllib3 ({}) or chardet ({}) doesn't match a supported "
  "/home/student/.local/lib/python3.8/site-packages/huggingface_hub/file_download.py:795: FutureWarning: 'resume_download' is deprecated and will be removed in version 1.0.0. Downloads always resume when possible. If you want to force a new download, use 'force_download=True'.
  warnings.warn(
  tokenzier_config.json: 100%
  vocab.json: 100%
  merges.txt: 100%
  added_tokens_map.json: 100%
  special_tokens_map.json: 100%
  config.json: 100%
  pytorch_model.bin: 100%
[nltk_data] Unziping tokenizers/punkt.zip.
[!] Processing for project: python-patcher
Getting fixed commits...
Repo: /home/student/.local/lib/python3.8/site-packages/Minecpp/python-patcher/python-patcher
Total commits: 1505
Processing SZZ
Processing Commits: 0%| [ 0:00<00:00, 1.51k/1.51k [ 0:00<00:00, 229kB/s]
| 639k/639k [ 0:00<00:00, 972kB/s ]
| 294k/294k [ 0:00<00:00, 16.7MB/s ]
| 57.0/57.0 [ 0:00<00:00, 53.6kB/s ]
| 15.5k/15.5k [ 0:00<00:00, 14.8MB/s ]
| 1.10k/1.10k [ 0:00<00:00, 473kB/s ]
| 892M/892M [ 0:00<00:00, 1.82MB/s ]
? Processing SZZ
Processing for commit: 0f3a401bbada18c123139947c7bd32bbiac79e3
Corresponding commit msg: Merge umtneko and higurashi Install data lnto one file. Flx higurashi.json.
Processing Commits: 0%| [ 0:00<00:00, 1.47k/1.47k [ 0:00<00:00, 7.17kB/s ]
Processing for commit: 0f4f6e79b06f3e1948ccfcfa8a3d8f088fd74872e
Corresponding commit msg: Fix inconsistent names - all python names should be camelcase
/home/student/.local/lib/python3.8/site-packages/huggingface_hub/file_download.py:795: FutureWarning: 'resume_download' is deprecated and will be removed in version 1.0.0. Downloads always resume when possible. If you want to force a new download, use 'force_download=True'.
  warnings.warn(
  tokenizer_config.json: 100%
  vocab.json: 100%
  merges.txt: 100%
  tokenizer_map.json: 100%
  special_tokens_map.json: 100%
  config.json: 100%
  pytorch_model.bin: 100%
[nltk_data] Unziping tokenizers/punkt.zip.
[!] Processing for project: python-patcher
Getting fixed commits...
Repo: /home/student/.local/lib/python3.8/site-packages/Minecpp/python-patcher/python-patcher
Total commits: 1505
Processing SZZ
Processing Commits: 0%| [ 0:00<00:00, 1.54k/1.54k [ 0:00<00:00, 1.22MB/s ]
| 798k/798k [ 0:00<00:00, 1.80MB/s ]
| 456k/456k [ 0:00<00:00, 1.98MB/s ]
| 2.11M/2.11M [ 0:00<00:00, 5.40MB/s ]
| 286k/286k [ 0:00<00:00, 313kB/s ]
| 703k/703 [ 0:00<00:00, 798kB/s ]
| 499k/499k [ 0:00<00:00, 1.80MB/s ]
? Processing SZZ
Processing for commit: f90d06a7c4bd30f6e096d705fd19d45e9e2ba23
Corresponding commit msg: Tidy code, make fixes for 2.6, fix circular import, fix CFBundleIdentifier
/home/student/.local/lib/python3.8/site-packages/huggingface_hub/file_download.py:795: FutureWarning: 'resume_download' is deprecated and will be removed in version 1.0.0. Downloads always resume when possible. If you want to force a new download, use 'force_download=True'.
  warnings.warn(
  Processing Commits: 1%| [ 0:00<00:00, 1.47k/1.47k [ 0:00<00:00, 1.8750s/lt ]

```

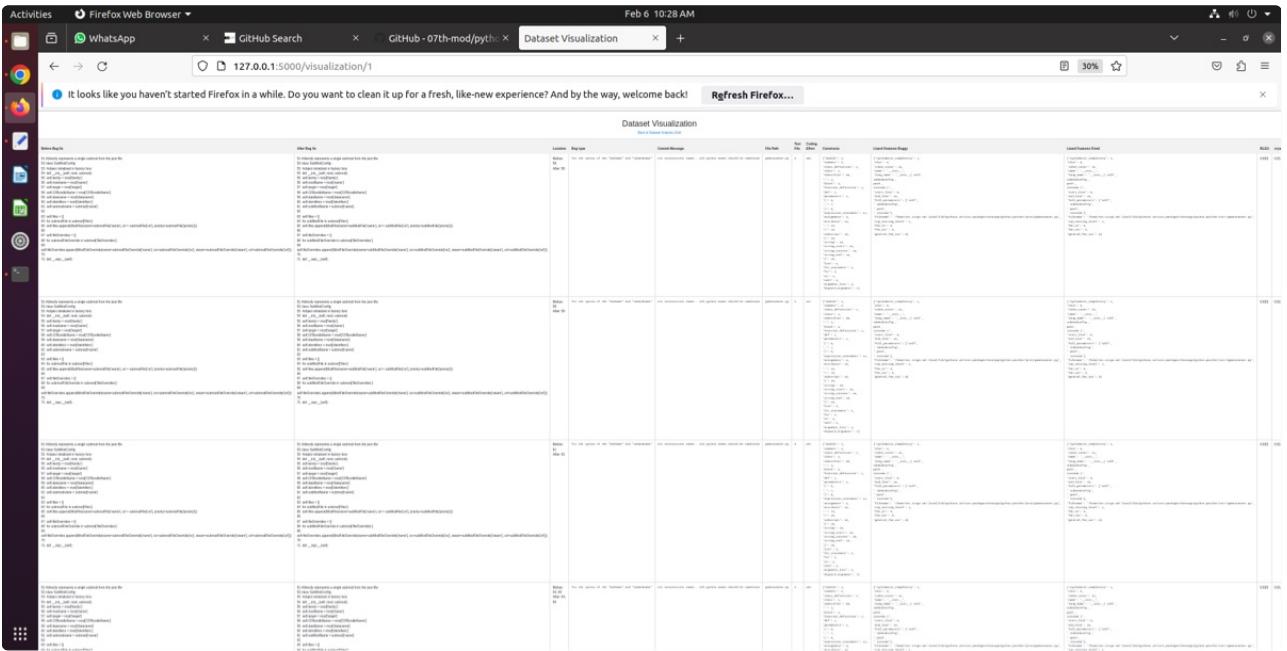
```

Activities Terminal • Jan 27 19:56 • student@vish: ~
warnings.warn()
Processing Commits: 97%
Processing for commit: 2b12943db4966f485a29c70b44f7fdcc1a8b0eb5
Corresponding commit msg: Migrate download hosting to GitHub
Processing Commits: 98%
Processing for commit: 1fadefb112e8437794fa793e9242a1a999c1562
Corresponding commit msg: Fix wrong download link for Unineko Answer Ryukishi Sprite Option
Corresponding commit msg: e9e19ddc5c32ef4a4493252dbccfa35533ccf98
Corresponding commit msg: Fix wrong download link for Higurashi Ch.8 Voice Only HD Sprite Option
Processing Commits: 98%
Processing for commit: 5fedabde1a8f42d0de83643b79643c2d82c1df
Corresponding commit msg: 09aa0b5e11cd82939a92fe8bae40ddaa391
Corresponding commit msg: [CI] Upgrade actions where possible to fix node depreciation warnings
Processing Commits: 99%
Processing for commit: 21fb98cf1963b9a32857f643c050f053e6de0b6
Corresponding commit msg: [CI] Try to fix download artifact error
Processing for commit: 8a523699239a0254a11ba0d1dee876dc8b1a44
Corresponding commit msg: Use setup-rust-toolchain to resolve depreciation warnings and toolchain toml
Processing Commits: 99%
Processing for commit: 1953d0405a418b7aae242fc4cfa077c9201223
Corresponding commit msg: Fix setup-rust-toolchain not finding rust-toolchain.toml
Processing for commit: 1ef2f593e3587ea70edcc04e1a67cdab9967e1d6
Corresponding commit msg: [CI] Fix cache detection by specifying project folder
Processing Commits: 99%
Processing for commit: ed924af53f45d3867ce3609995df15d896ec2c
Corresponding commit msg: [CI] Fix syntax error
Processing for commit: 798cb3dcf70081935fc2cdf2132fb8747702e2
Corresponding commit msg: Also clear Update Folder (.txt scripts) when clearing scripts
/home/student/.local/lib/python3.8/site-packages/huggingface_hub/file_download.py:795: FutureWarning: 'resume_download' is deprecated and will be removed in version 1.0.0. Downloads always resume when possible. If you want a new download, use 'force_download=True'.
warnings.warn()
Processing Commits: 100%
Processing for commit: bac0ac65e3524d811f78c64d94e0c50bddf32fe
Corresponding commit msg: Fix messed up TsumiHoroboshi-System.tz which contained Ep08 files instead of Ep06 files
Processing Commits: 100%
Removing project folder: python-patcher
* Serving Flask app 'Mnecpp.app'
* Debug mode: off
Matplotlib is running on development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [27/Jan/2025 10:32:48] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [27/Jan/2025 10:32:48] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [27/Jan/2025 10:32:52] "GET /quantitative HTTP/1.1" 200 -
/home/student/.local/lib/python3.8/site-packages/Mnecpp/app.py:114: UserWarning: Starting a Matplotlib GUI outside of the main thread will likely fail.
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))
127.0.0.1 - - [27/Jan/2025 10:32:58] "POST /quantitative HTTP/1.1" 200 -

```

which started local server at <http://127.0.0.1:5000>

Navigate and Analyze Bug Type Descriptions Inferred by the Pre-trained LLM:



link to csv we get <https://drive.google.com/file/d/1UirjVvVFD9I3qpdlrQIDgyCEOMyyjGu/view?usp=sharing>.

On Labeling 15 bug fix pair with one of these labels {TOOL, DEV, BOTH, NONE} I calcified (between Bug Type and Commit Message) as:

row number: 1, 12, 14, 18, 20, 25, 38, 81, 91, 99, 105, 110, 115, 125, 130

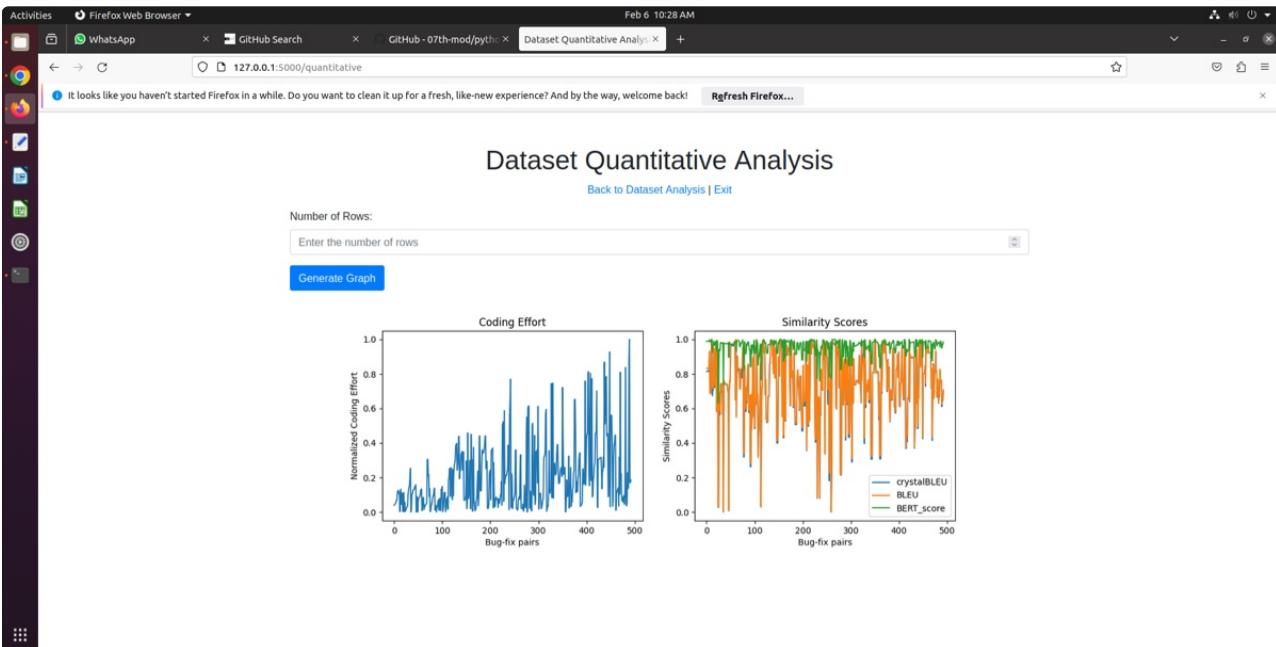
7 TOOL: LLM's description is more accurate. 14, 18, 20, 25, 38, 99, 110

5 DEV: Developer's commit message is more accurate. 1, 12, 81, 91, 115

3 BOTH: Both are equally helpful. 105, 125, 130

NONE: Neither is helpful.

Visualize Metrics Through the Tool Provided GUI:



1. Interpretation of the Metric "Coding Effort"

The **left graph** shows "Normalized Coding Effort" versus "Bug-fix pairs." It suggests that initial bug-fix pairs have **lower coding effort**, but as the number of bug-fix pairs increases, the coding effort also rises significantly indicating that some bug fixes required **more substantial code changes** than others.

Analysis:

Areas with **higher coding effort** indicate significant modifications to the codebase, possibly affecting maintainability.

Large spikes may suggest **complex bug fixes**, possibly requiring **code refactoring** or deeper debugging efforts.

Developers should **review sections with high coding effort** to check if:

There are opportunities to **improve maintainability** and reduce complexity.

Future development can be optimized by reducing the need for extensive code changes.

2. Identifying Patterns and Trends

A noticeable **increasing trend** in coding effort over time suggests that later bug fixes were more demanding. Some bug-fix pairs show **spikes**, indicating that certain fixes required disproportionately high modifications.

Key Identifications:

Identify **specific files or modules** that frequently require large efforts.

Track **recurring patterns** of high-effort bug fixes—this may indicate poor code structure or areas needing better documentation.

Focus on **high-effort** regions and conduct **code reviews** to improve maintainability.

Identify **two bug-fix pairs** with high impact on coding effort and analyze why they required **more effort** than others.

The **right graph** visualizes **similarity scores** for bug-fix pairs using multiple metrics:

CrystalBLEU (Blue Line), BLEU Score (Orange Line), BERT Score (Green Line)

The **BERT score is higher** than BLEU and CrystalBLEU, indicating that it **captures semantic similarities more betterly**.

4. Suggest Improvements:

A **low similarity score** may indicate a significant change in how a bug was fixed.

A **high similarity score** suggests that the fix closely follows the original buggy code, which may mean only **minor corrections** were made.

BLEU scores vary significantly, possibly indicating that some fixes required **more rewriting** compared to others.

Investigate **bug-fix pairs with low similarity scores**—these might represent **complex fixes** requiring significant logic changes.

Use **BERT Score as a reliable metric** for evaluating bug-fix quality, as it captures contextual changes better than traditional BLEU-based metrics.

DISCUSSION AND CONCLUSION

This lab provides us with hands-on experience with **MineCPP**, repository mining, and bug-fix analysis. We also explored bug-fix patterns, coding effort trends, and comparative repository analysis. also we get to learn about SEART to search Repo online.

Analyze Bug Type Descriptions

for Label 15 bug-fix pairs with **(7 TOOL, 5 DEV, 3 BOTH, 0 NONE)**. I calcified 15 row as 1, 12, 14, 18, 20, 25, 38, 81, 91, 99, 105, 110, 115, 125, 130

Analysis of Coding Effort:

Higher coding effort was observed in later bug-fix pairs, indicating more substantial code changes.

Spikes in effort suggested complex bug fixes that may have required code refactoring or deeper debugging.

These areas with high coding effort may point to sections in the codebase that need further optimization. Developers should focus on these parts to reduce complexity and improve maintainability.

APPENDIX

I would like to express my sincere gratitude to my course instructor, Prof. [**Shouwick Mondal**](#), for his invaluable guidance and support throughout this lab. I also appreciate the assistance from All TAs, whose help with me with troubleshooting.

Additionally, I am grateful for the resources provided, including Which I have sited down which helped me resolve issues efficiently. Finally, I'd like to thank my peers for contributing to a collaborative and supportive learning environment.

CITATION

- [1] Set-litgn. (n.d.). *GitHub - SET-IITGN/MineCPP: MineCPP - A tool to mine GitHub repositories and obtain a dataset containing a list of bug-fix pairs and related information*. GitHub. <https://github.com/SET-IITGN/MineCPP>
- [2] Th-Mod. (n.d.). *GitHub - 07th-mod/python-patcher: Mod Installer for the Higurashi and Umineko Games*. GitHub. <https://github.com/07th-mod/python-patcher>
- [3] Dabić, O. (n.d.). *GitHub Search*. [Usi.Ch](https://seart-ghs.si.usi.ch/). Retrieved January 28, 2025, from <https://seart-ghs.si.usi.ch/>
- [4] Class Slides - <https://drive.google.com/file/d/16tTzLdotE0VJyIMcwdUcuMOAn8-Yo0at/view?usp=sharing>
- [5] Avula, S. K. [@SaiKrishnaAvula-u5b]. (n.d.). *MineCPP: A Tool for Mining Bug-Fix pairs and their Structures*. Youtube. Retrieved January 28, 2025, from <https://www.youtube.com/watch?v=ln99irvbADE&feature=youtu.be>
- [6] Lab Manual - <https://drive.google.com/file/d/1UAK4tYvQiZtrP50Fzc4idhxfL06dZPWA/view>
- [7] Video Demonstration- <https://youtu.be/ln99irvbADE>
- [8] SEART (Github Repo Search)- <https://seart-ghs.si.usi.ch/>