Mining Software Repositories

(MSR)

Introduction

Extracting, analyzing, and interpreting data from software project repositories (Git, GitHub, Jira, etc.)

Importance:

- Software quality improvement
- Bug prediction
- Effort estimation
- Understanding team activity

QBET

Data Types

- Commit logs
- Bug reports
- Pull requests
- Test results
- CI/CD logs

Process of MSR

- **Define research question:** e.g., “Which files are most bug-prone?”
- **Select repositories:** Open-source or organization-specific
- **Extract data:** Commits, issues, PRs
- **Clean and preprocess data:** remove duplicates, normalize
- **Analyze data:** metrics calculation, visualization
- **Interpret results:** derive insights, recommendations
- **Report & validate findings:** documentation, reproducibility

MSR Metrics & Analyses

- **Code & Repository Metrics**
  - **Lines of Code (LOC)**: growth, churn
  - **Commit count**: per developer, per module
  - **Code churn**: added vs deleted lines
  - **Bug density**: bugs per LOC
- **Team & Process Metrics**
  - Developer activity trends
  - Collaboration networks
  - Issue resolution time
- **Example Visualization**
  - Commit trends over time
  - Bug distribution per module
  - Developer collaboration network graph

Reference

Codabux, Z., Fard, F., Verdecchia, R., Palomba, F., Di Nucci, D. and Recupito, G., 2024. Teaching Mining Software Repositories. In Handbook on Teaching Empirical Software Engineering (pp. 325-362). Cham: Springer Nature Switzerland.