

Reading assignment

MSR course

Faculty of Computer Science
Software Languages Team

Presenter: Prof. Dr. Ralf Lämmel

Paper: Manishankar Mondal, Chanchal K. Roy, Kevin A. Schneider: Prediction and ranking of co-change candidates for clones. In Proceedings of MSR 2014, 32-41, ACM

What is the context in software engineering?

- Clones in a code base
- Possibly independent evolution of clones
- Consistent co-change of clones sometimes needed
- Infer the clones to co-change

A clone: identical or similar code fragments scattered over a code base

What is the basic motivation for this research?

- Not all clones need co-change:
 - Looking at all clones impedes **productivity**.
- Some clones need to consistently co-change:
 - Missing these clones impedes **correctness**.
- To achieve both productivity and correctness, we need to automatically **predict** clones to co-change.

What sort of artifacts are considered?

Sys.	Lang.	Domains	LOC	Revs
Ctags	C	Code Def. Generator	33,270	774
QMail Admin	C	Mail Management	4,054	317
jEdit	Java	Text Editor	191,804	4000
Freecol	Java	Game	91,626	1950
Carol	Java	Game	25,091	1700
Jabref	Java	Reference Manager	45,515	1545

Revs = Revisions. Sys = Systems

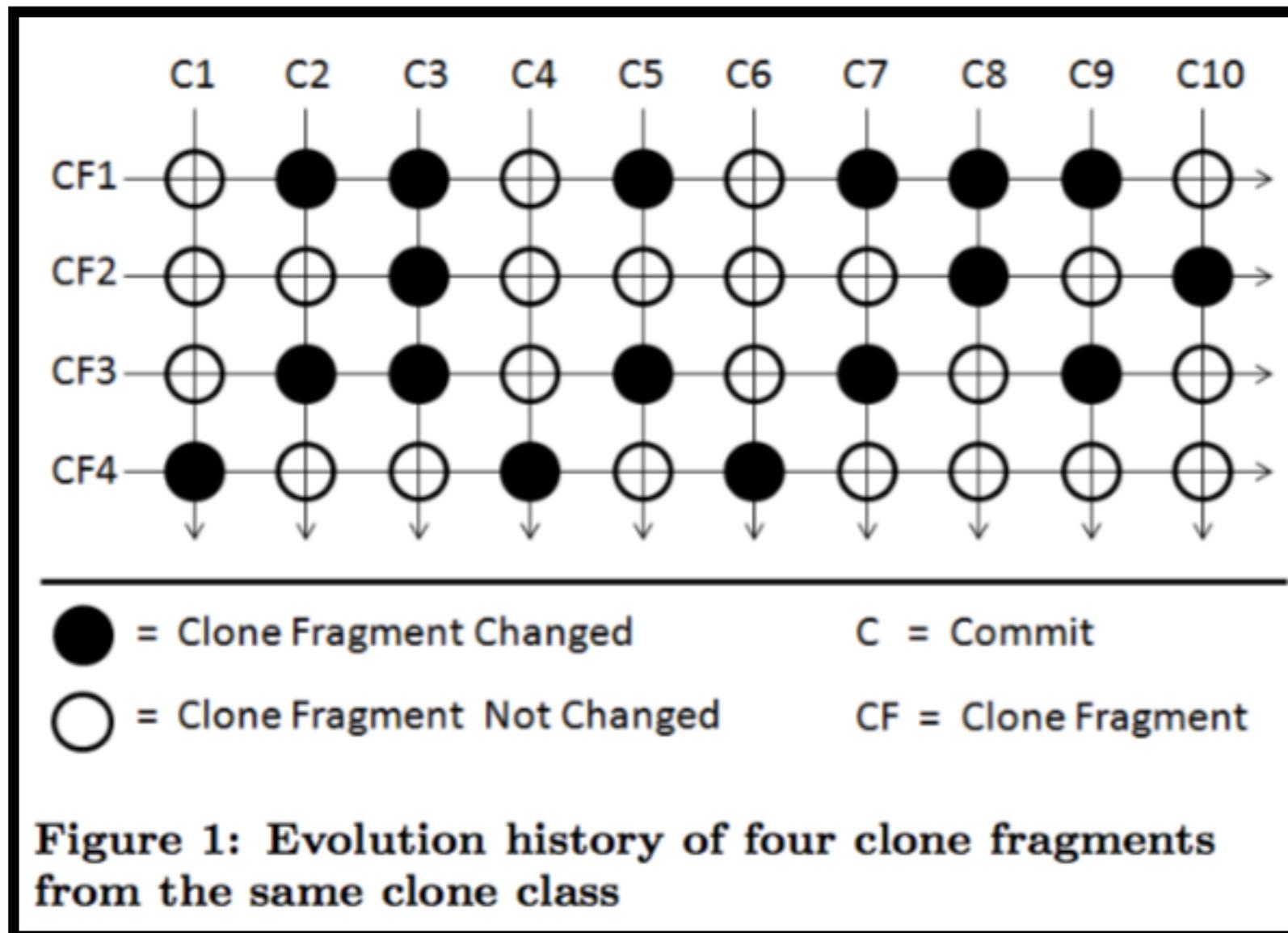
Source code
of systems

What research questions or hypotheses are posed?

Let's focus on RQ1.

- RQ1: Can we **predict co-change candidates** for a particular clone fragment **by using evolutionary coupling**?
- RQ2: Can we achieve **better ranking** of co-change candidates that exhibited evolutionary coupling **by considering co-change recency instead of co-change frequency**?
- RQ3: What are the characteristics of the clone fragments that exhibit evolutionary coupling? **Which characteristic** can **help** us **in better ranking of co-change candidates** that have not yet exhibited evolutionary coupling?
- RQ4: How can we rank both types of co-change candidates - (1) the candidates that exhibited evolutionary coupling, and (2) the candidates that did not exhibit evolutionary coupling for a particular clone fragment?

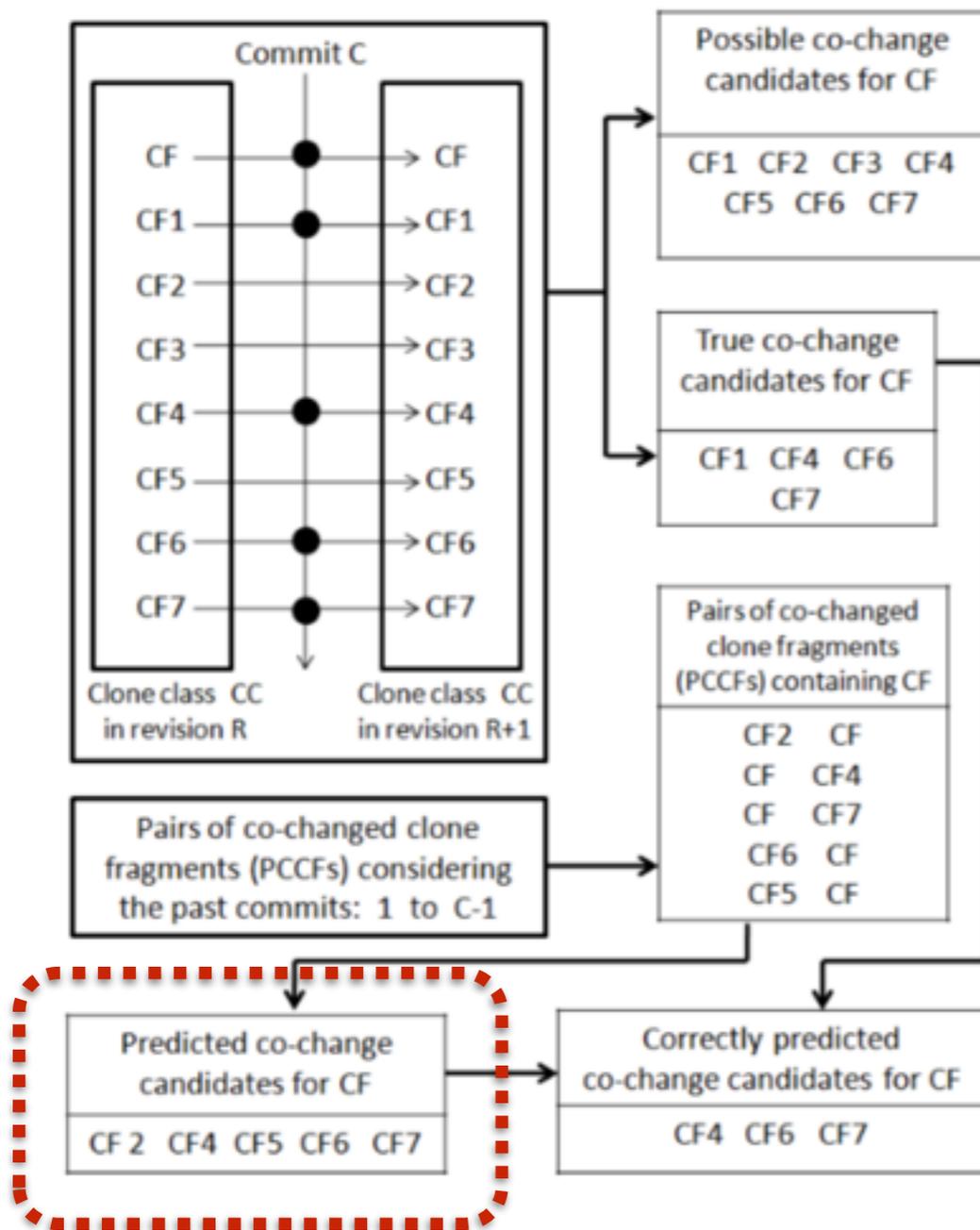
What data is extracted and how?



Pairs of co-changed fragments are extracted, subject to clone detection and tracking clone fragments over commits.

For instance, CF1 and CF3 co-change at commit point C2.

What data is synthesized and how?



For each commit and each clone, **we predict whether a change to a specific CF implies co-change for some other CF** by looking at pairs of co-changed fragments in the **previous commits**. Basically, the more often a co-change happened in the past, the more likely it will happen again.

Are the results analyzed?

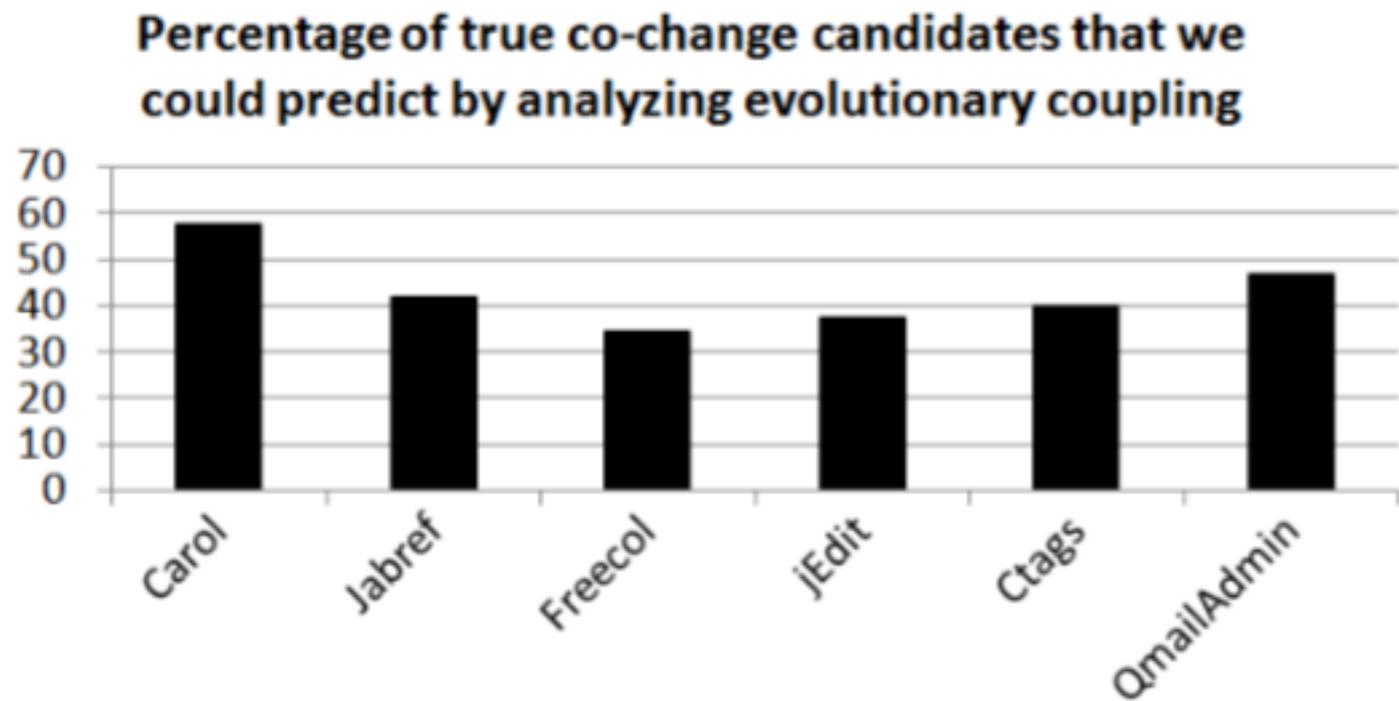


Figure 6: The proportion of true co-change candidates that we could predict by analyzing evolutionary coupling (i.e., the recall)

Yes. For instance, the **recall of predicting true co-change candidates** is computed for different subject systems.

What are the overall results?

Answer to RQ 1. From our analysis and discussion we can say that evolutionary coupling can help us predict true co-change candidates for a particular clone fragment with considerable accuracy in terms of precision (= 85.18%) and recall (= 43.17%).

See other RQs alike.

Is the research reproducible?

Not completely:

- clone/detection coupling numbers made persistent online
- all measurements included only in text (not online)
- source-code of implementation / tools not available

We automatically retrieve evolutionary coupling from each of the candidate subject systems. The XML files containing the pairs of co-changed clone fragments are available online¹. Our primary goal in this research work is to investigate whether we can predict and rank co-change candidates for clones using evolutionary coupling. In the following subsections, we answer four research questions regarding this.

¹XML Files: <https://homepage.usask.ca/~mam815/ongoingresearch.php>