

Improving Concept Location using Contextual Relations

by Tezcan Dilshener

Task of locating relevant program elements implementing concepts at hand during software maintenance.

What is Concept Location?



Querying lexical similarities between search terms and source code identifiers.

How to perform Concept Location?

query = {calculate covariance}

```
public class ReaderCorrelation extends BaseRiskCorrelation{
    public BigDecimal getCorrelation(Integer pIndexID){
        return buildCorrelation(pIndexID);
    }

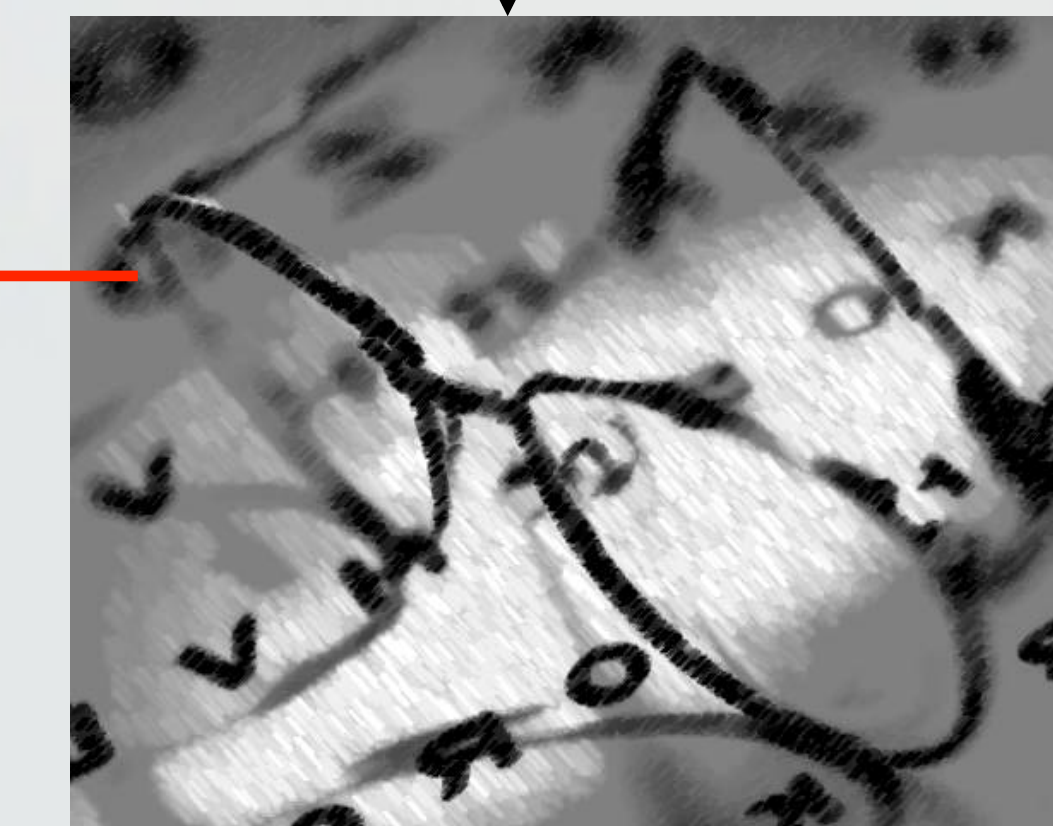
    public BigDecimal getCovariance(Integer pIndexID){
        return buildCovariance(pIndexID);
    }

    private BigDecimal buildCovariance(Integer pIndexID){
        Integer actIndexID = ReaderIndex.getIndex(pIndexID);
        BigDecimal vola = ReaderVolatility.getVolatility(actIndexID);
        BigDecimal assetMV = getCorrelation(actIndexID) * stdRisk;
        return calcCovar(vola, assetMV);
    }
    ...
}
```

Also to locate concepts that can not be detected by directly looking at source code identifiers:

- Multi word (n -grams) concepts.
e.g. “standalone risk”.
- Compound concepts.
e.g. “covariance” derived from “correlation”, “volatility” and “index”

Why improve Concept Location?



Research Aim

Use of conceptual information captured by domain specific ontologies to:

- improve concept mapping during searchable contextual model generation.
- provide meaningful clues to identify n -gram concepts during concept location.
- enhance lexical similarity techniques.

Research Method

Creation of a contextual model providing strong relational clues by combining:

- domain specific ontologies.
- natural language (NL) approaches.
- programming semantics e.g. call-graph.

Validation by empirical analysis:

- comparing precision and recall against existing methods.

Expected Contributions

- Novel concept mapping approach using domain specific ontology relations.
- Detection of all relevant program elements implementing n -gram concepts.
- Ontology supported abbreviation expansion.
- Semi-automatic extraction of concepts from source code.