Enhancing Software Traceability By Automatically Expanding Corpora With Relevant Documentation

Tathagata Dasgupta, Mark Grechanik: U. of Illinois, Chicago
Evan Moritz, Bogdan Dit, Denys Poshyvanyk: College of William and Mary
Requirements Traceability (RT)

- Mapping requirements to software artifacts
  - Requirements are expressed using text, diagrams, or schemas
  - Software artifacts include code, configuration data, and databases
- Requirements traceability leads to better code understanding
Tracing Requirements

locate
secure
Abstractions

Realizations
Traceability Links

locate

service

send

XML

secure
Traceability Links (TLs)

- TLs are mappings between concepts in high-level requirements and design documents and program entities.

- TLs can be viewed as matches between words in requirements and code.
Traceability Is a Similarity Measure

Upper lip

Upper lip with shaved mustache
Traceability Is a Similarity Measure

word matching
The Problem

Ouch!

Low precision of TLs

Incorrect TLs

Missed TLs

Lower accuracy

It is a norm

Vocabulary problem

Polysemy

Synonymy

Different stakeholders

Software quality is affected
Our Hypotheses

It is possible to increase the precision of information retrieval approaches that are based on syntagmatic associations by expanding the vocabulary of artifacts using related words.

by using JDK API calls as semantic anchors.

by using the hybrid of syntagmatic and paradigmatic vocabulary expansions.
Key Question

How to expand the vocabulary automatically in a semantically meaningful and systematic way in order to increase the accuracy of automatic traceability approaches?
Expansion With Related Words

Expansion Word: disease
Original Word: HIV
Expansion Word: AIDS

\[ C(\text{disease} \Rightarrow \text{HIV}) = 0.16 \]
\[ S_d: 535,000,000 \]
\[ S_{d \cap H}: 85,700,000 \]

\[ C(\text{AIDS} \Rightarrow \text{HIV}) = 0.96 \]
\[ S_A: 129,300,000 \]
\[ S_{A \cap H}: 124,000,000 \]
Expanding JDK API Calls

Method Detail

### getStringTable

definition:
protected String[] getStringTable()

- Returns: the string table for class PrintQuality.

- Overrides:
  - getStringTable in class EnumSyntax

- Returns:
  - the string table

### getEnumValueTable

definition:
protected EnumSyntax[] getEnumValueTable()

- Returns: the enumeration value table for class PrintQuality.

- Overrides:
  - getEnumValueTable in class EnumSyntax

- Returns:
  - the value table

### getOffset

definition:
protected int getOffset()

- Returns: the offset of the lowest enumeration value.
Experiments

We used TraceLab

- http://www.coest.org

We experimented with multiple IR approaches

- VSM, LSI, JS, LDA, RTM
# Corpus Treatment Methods

**Strawman**  
Our baseline method  
- Complete source code corpus is treated as bag of words.

**JDK expansion**  
- Replaced JDK API calls with their corresponding description in the JDK documentation.  
- Identifiers are split.  
- Comments are discarded.

**J+W**  
Extension of JDK expansion where  
- Wordnet synsets of dictionary words found in the corpus are injected.

**J+S**  
Expansion of S and JDK  
- Comments are included.

**J+S+W**  
is ENTRANCER
Summary of Results
Results

RQ1
Expanding the corpus with the documentation JDK API calls only is often not enough to get higher precision of traceability links when applying word match similarity methods.

RQ2
expansion of the corpus with a combined documentation from the JDK API calls and Wordnet does not always result in a higher precision of traceability links.
Results

RQ3
including words from comments results in a higher precision of traceability links when expanding the corpus with a combined documentation from the JDK API calls and Wordnet.

RQ4
There is a correlation between the size of the corpus and higher precision of recovered TLs.

RQ5
using VSM results in a higher precision of traceability links when expanding the corpus with a combined documentation from the JDK API calls and Wordnet
It Is Just the Tip of The Iceberg

We need systematic methods to expand the corpora to experiment with traceability and to increase the precision of different traceability approaches.
Conclusions

An automatic approach for ENhancing TRAceability usiNg API Calls and rElevant woRds (ENTRANCER).

This is the first comprehensive study of an automated approach that expands the base multilingual corpora, i.e., Italian and English.

We showed that ENTRANCER can increase the precision of the recovered TLs by up to 31% in the best case.
Email: drmark@uic.edu
Collaborators: Denys Poshivanyk, College of William and Mary
Students: Tathagata Dasgupta, Evan Moritz, Bogdan Dit
http://www.cs.uic.edu/~drmark/entrancer.htm