lab soft
software engineering laboratory

A Literature Review and Comparison of Three Feature Location Techniques using ArgoUML-SPL

Daniel Cruz[1], Eduardo Figueiredo[1], Jabier Martinez[2]

February 08, 2019

[1] Universidade Federal de Minas Gerais, Brazil
[2] Tecnalia, Spain

Schedule

□ Introduction

□ Literature Review

□ Comparative Study

□ Threats to Validity

□ Conclusion

□ Future Work

Introduction

Introduction

- Software Product Lines
  - Allows a high level of reuse
  - Usually created through an extractive process from existing systems

- Variability Mining
  - In the extractive context, is the process of locating features in an existing system
  - The goal is to produce variations of an SPL

Introduction

- Feature Location Techniques
  - Identification of code artifacts that implement a feature
  - Possibility to automate the refactoring of systems, as long as the features are located
- Related Work
  - Focus on most recent works, taking into account the evolution of algorithms in areas such as information retrieval and machine learning

Literature Review

Goals

- Revisit feature location strategies
- Complement previous literature reviews
- Provide a strong background for the comparative study

Research Questions

- *RQ1*. What are the strategies used by the most recent feature location techniques?

- *RQ2*. What are the characteristics of feature location techniques?

- *RQ3*. How have feature location techniques been evaluated?

Protocol

- Collection Process
  - 142 papers collected
  - Digital Libraries: ACM, IEEE, Science Direct

- Inclusion Criteria
  - Published from 2005 to 2017
  - Studies that propose feature location techniques or improvements

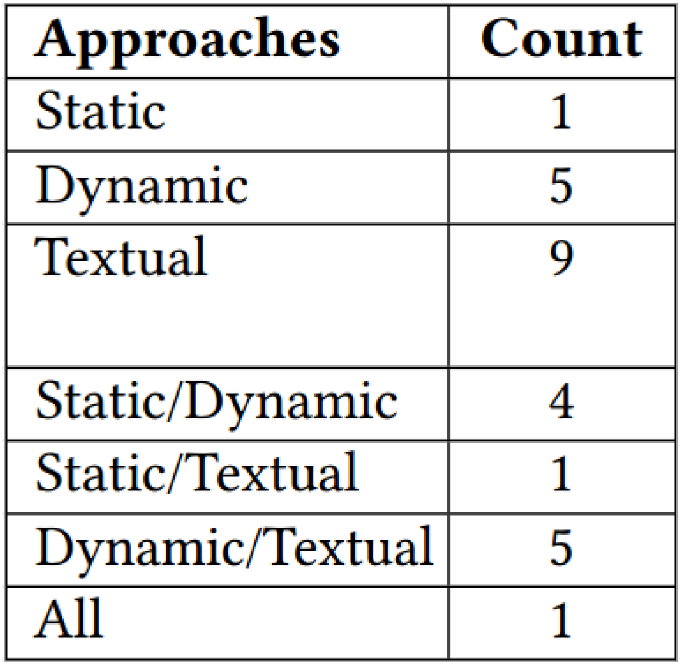
Protocol

- Exclusion Criteria
  - Case studies only using existing techniques
  - Empirical studies comparing techniques
  - Surveys with comparative analysis among techniques

- 26 papers were selected for the review

RQ1. Strategies

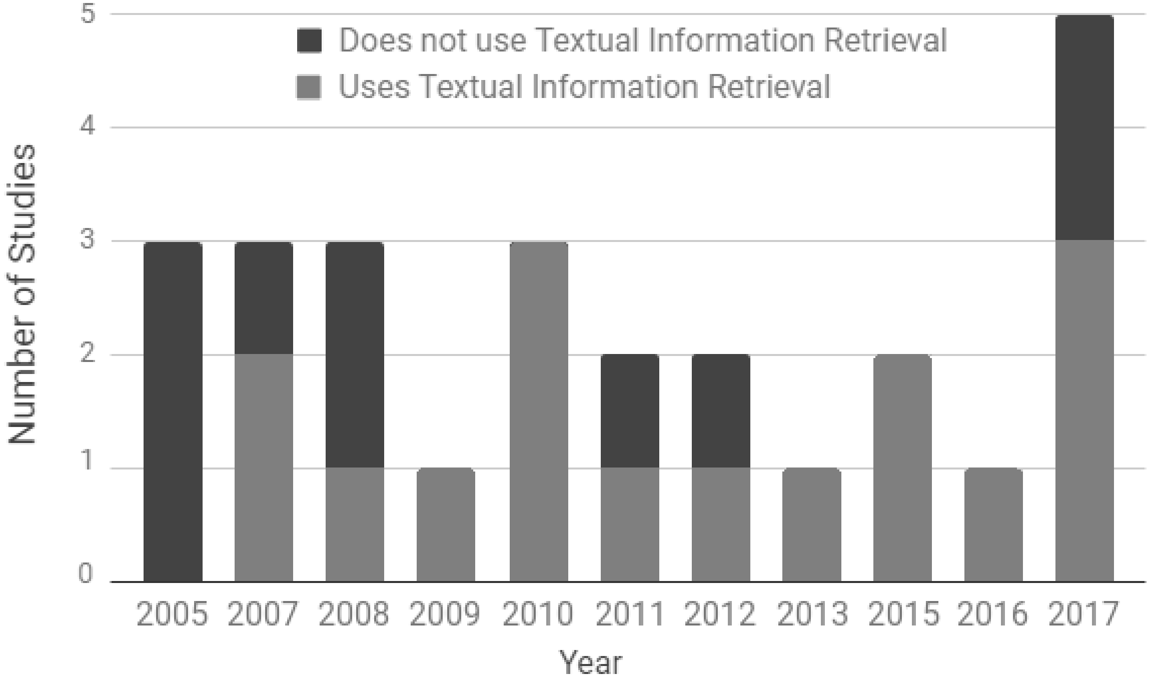
- We characterize the strategies used by the techniques based on the approaches of location:
  - Static
  - Dynamic
  - Textual
  - Hybrid



Approaches	Count
Static	1
Dynamic	5
Textual	9
Static/Dynamic	4
Static/Textual	1
Dynamic/Textual	5
All	1

RQ1. Strategies

- Many techniques include at least one step where textual information retrieval was used



RQ2. Characteristics

- Type of process: automatic (65.4%) vs semi-automatic (34.6%)
- Input artifacts
  - **Source Code**
  - **Execution Traces**
  - Ontology models
  - Source control history

RQ2. Characteristics

- Output
  - Rank of Artifacts with many granularities:
    - Classes
    - Methods
    - Blocks
  - Exploratory User Interface

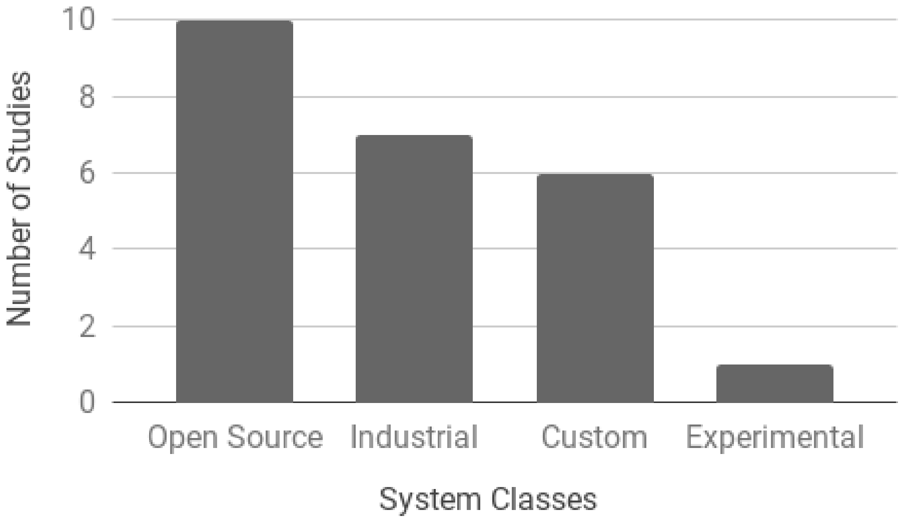
***RQ3*. Evaluation**

- Lack of standardization in the techniques results evaluation, including quantitative and qualitative analyses

- Nine different quantitative metrics:
  - **Precision (7)**, **Recall (7)**, Mean Reciprocal Rank (4), F-Measure (3), Effectiveness (3), Lattice Distillation Factor (1), Lattice Browsing Complexity (1), Uniqueness (1), Coverage (1)

RQ3. Evaluation

- Different types of systems used for techniques evaluation:
  - Open Source, Industrial System, Custom, Experimental



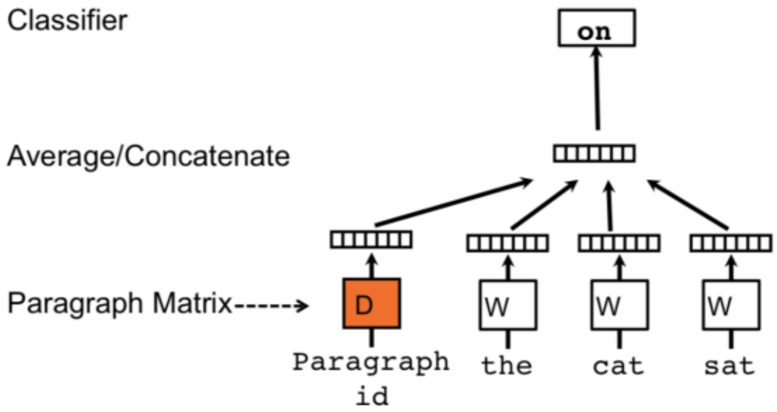
Comparative Study

Study Goal

- Compare techniques to:
  - Provide guidelines for future industrial cases
  - Improve the feature location state of the art
- Focus on comparing textual information retrieval techniques
- Use a benchmark to provides quantitative evaluation

Feature Location Techniques

- Paragraph Vectors (DV)
  - Learn vectors representations for documents and words using neural networks
  - The vector has K dimensions, where K is a defined hyperparameter



Feature Location Techniques

- Latent Dirichlet Allocation (LDA)
  - Probabilistic model for collections of discrete data such as text corpora
  - Represents a document as a probabilistic mixture of topics, where a topic is a distribution of words
  - Each document has a probability of belonging to each latent topic, built on the corpus model
  - The number of topics is defined by a parameter K

Feature Location Techniques

- Latent Semantic Indexing (LSI)
  - Obtains an underlying latent semantic structure from data composed by words
  - Applies Singular Value Decomposition (SVD) to factorize the terms in the text into K orthogonal factors, where K need to be defined
  - The goal is to obtain a new representation that benefits the information retrieval

ArgoUML-SPL Benchmark

- Created from an SPL of a UML editor with 8 optional features
- Unify the largely used ArgoUML-SPL
- Ground-truth for feature location

https://variability-challenges.github.io/2018/ArgoUMLSPL/

ArgoUML-SPL Benchmark

- Generate different set of variants

- Each variant is a product of the SPL, e.g, a combination of the eight optional features

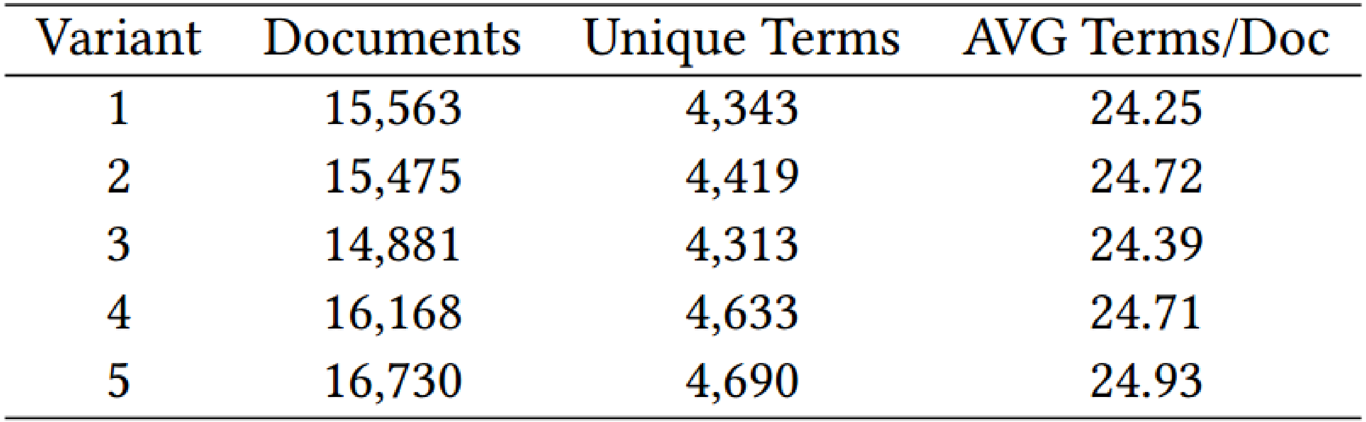
Textual Characterization

- The benchmark description includes metrics about size in terms of lines of code (LOC)

- For the purpose of this work, it is important to provide a characterization of the benchmark from the perspective of documents and terms

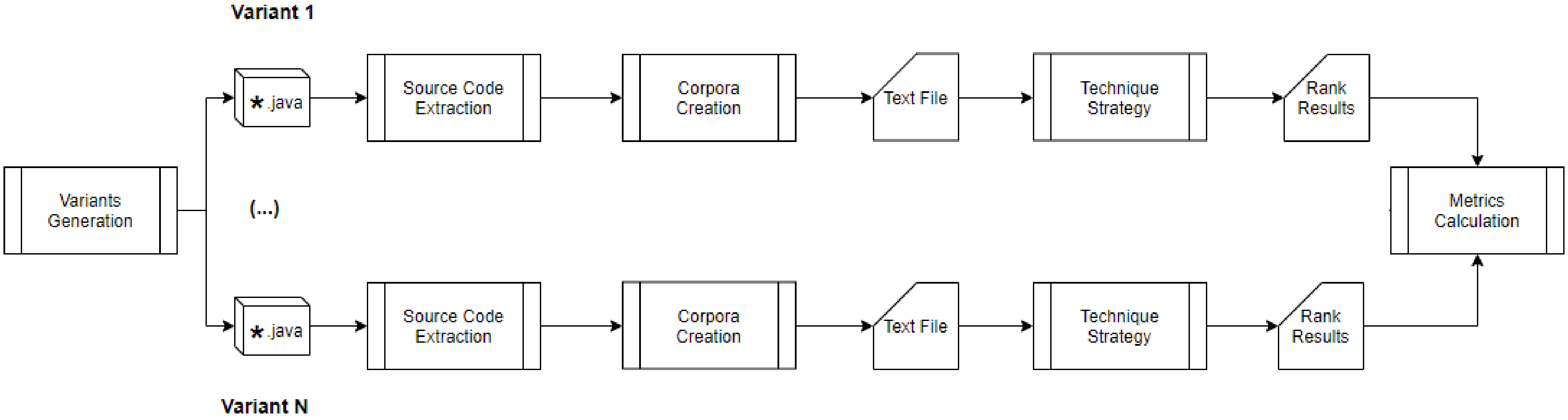
Textual Characterization

- We described the variants according to two textual metrics:
  - Unique Terms
  - Average Terms per Document

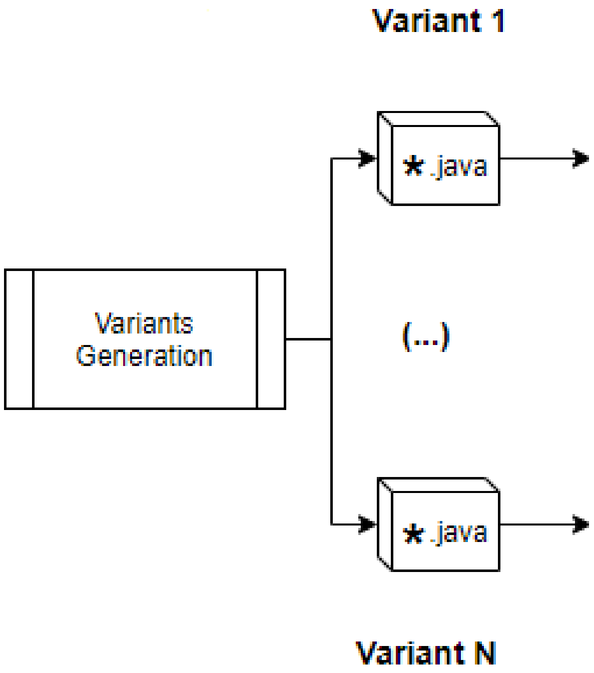


Variant	Documents	Unique Terms	AVG Terms/Doc
1	15,563	4,343	24.25
2	15,475	4,419	24.72
3	14,881	4,313	24.39
4	16,168	4,633	24.71
5	16,730	4,690	24.93

Study Design

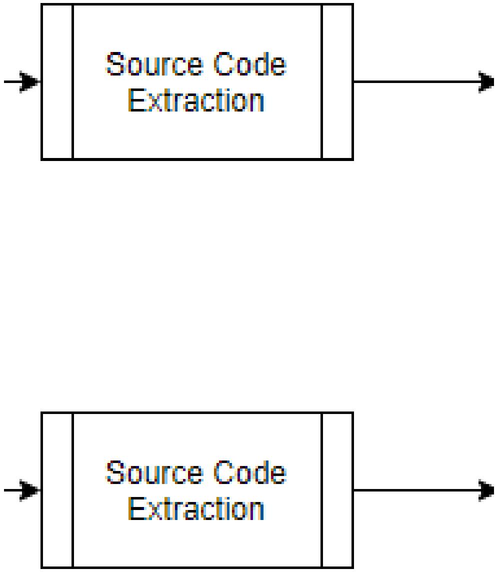


Study Design



We generate five random variants (products) from the ArgoUML-SPL Benchmark

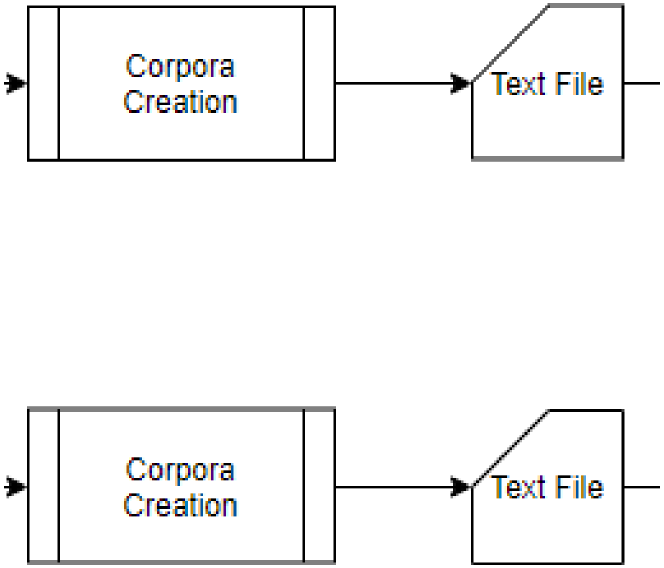
Study Design



The source code of each variant is extracted using an ANTLR v3 Java-based tool: Teaser[1]

[1] https://github.com/nkraft/teaser

Study Design



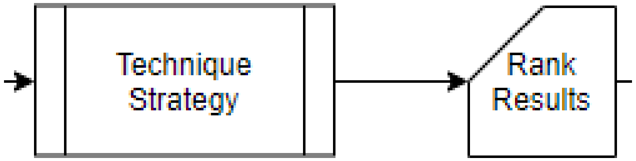
The XML extracted is processed by a custom parser[1] and preprocessed as follows:

- CamelCase and Naming conventions split
- Lower Case normalization
- Non-Letter tokens filtering
- English stopwords and length filtering

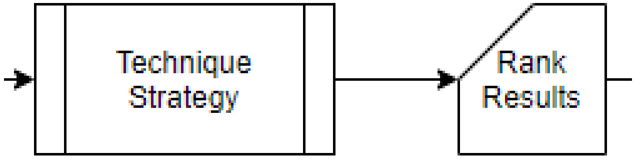
The results are text files for each class and method

[1] https://github.com/DVSCross/TextualIRFeaturesImpl

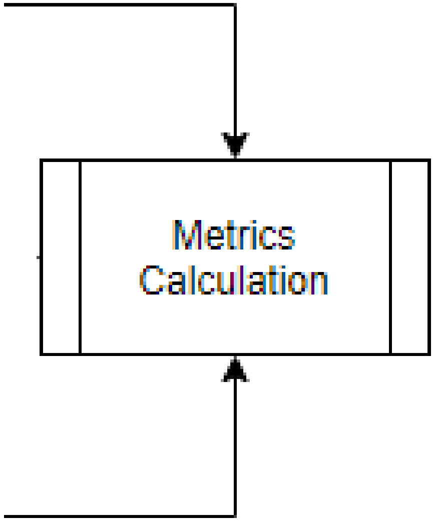
Study Design



The techniques are applied and the outputs are ranks containing all the artifacts (classes and methods)



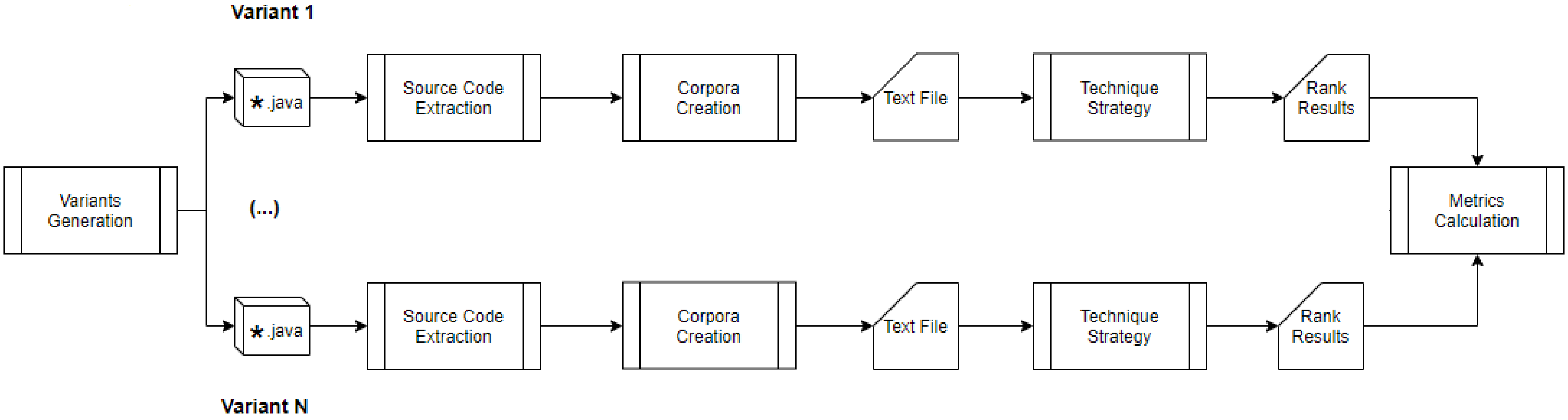
Study Design



Finally, the metrics are calculated by the ArgoUML-SPL Benchmark using a ground truth. They are:

- Precision
- Recall
- F-Measure

Study Design



Results

Results

- The metrics used and available at the benchmark:
  - Precision, Recall, and F-Measure

- All the techniques produce a rank as output, containing all the artifacts from source code (classes and methods)

- So, the difference is about the results order

Results

- Relevance filtering on the techniques results
  - The main resources that implement the feature must be on the top

- Take the first N results
  - N = 10
  - N = 100
  - N = 1000

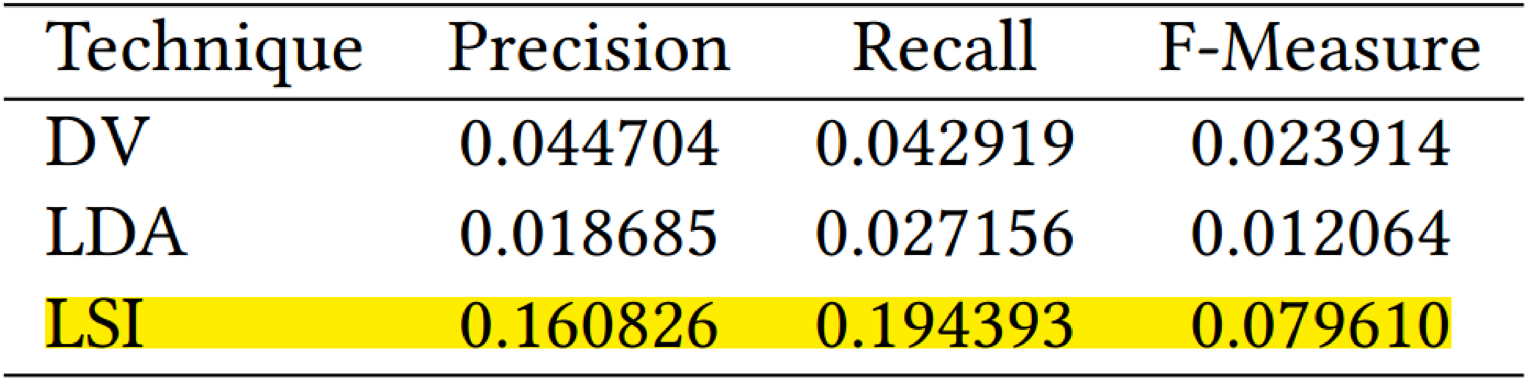
Results

- As mentioned, each technique model has a hyperparameter K, that assume values as follow:
  - 100, 200, 300, 400, and 500

- The average for all K and N variations was taken

Results

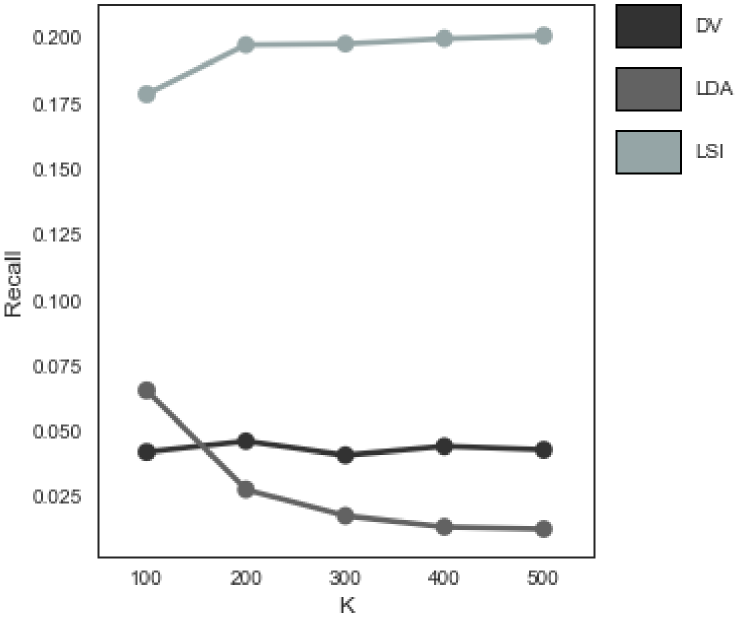
- □ LSI got slightly better results

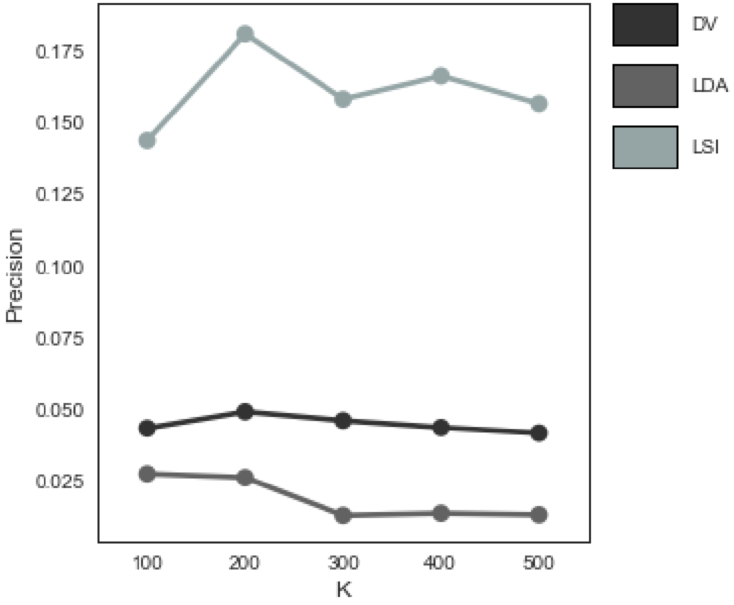


Technique	Precision	Recall	F-Measure
DV	0.044704	0.042919	0.023914
LDA	0.018685	0.027156	0.012064
LSI	0.160826	0.194393	0.079610

Results

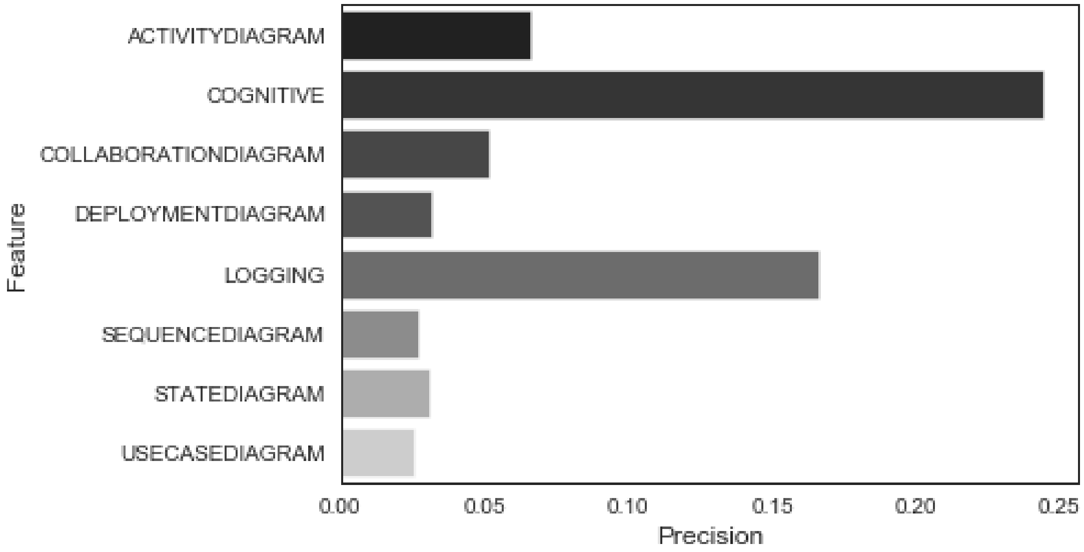
- DV and LSI presents better results at K value equals to 200 and LDA decreases the recall as K is increased





Results

- Some features have better metric results
  - More distinct terms
  - Better code styling



lab▪soft
software engineering laboratory

Threats to Validity

Threats to Validity

- Possible bugs in the implementation
  - To avoid them the implementations were done using a widely used library (Gensim[1])
  - The code is available on open source format[2]

- Possible bugs in the benchmark, e.g., on the ground-truth
  - This is the first published work using it

[1] https://radimrehurek.com/gensim/

[2] https://github.com/DVSCross/TextualIRFeaturesImpl

Conclusion and Future Work

Conclusion

- The feature location activity in the context of extractive SPL adoption is still challenging

- We have presented a literature review that revisits the feature location approaches

- We provided a characterization of ArgoUML-SPL Benchmark with regard to important aspects of textual techniques

Conclusion

- We have shown that the use of textual information retrieval techniques, in isolation or combined with other techniques, is sustained along the years

- The result suggests that LSI outperforms slightly, DV and LDA

Future Work

- Confirm these results with other benchmarks
- Propose feature location techniques by extending our current implementations
- Evaluate the application of LSI in hybrid approaches

A Literature Review and Comparison of Three Feature Location Techniques using ArgoUML-SPL

Daniel Cruz[1], Eduardo Figueiredo[1], Jabier Martinez[2]

February 08, 2019

[1] Universidade Federal de Minas Gerais, Brazil

[2] Tecnalia, Spain