

Evaluating and Improving Software Quality using Text Analysis Techniques: A Mapping Study

Faiz Ali Shah and Dietmar Pfahl

Institute of Computer Science, University of Tartu, Estonia

Introduction

- 80 to 85% of data in software repositories is unstructured.
- Software artifacts , such as requirement document, test cases, commit messages contain unstructured data.
- Utilizing unstructured data is important to derive useful information.
- Analyzing unstructured data could be beneficial to improve and evaluate software Quality.

Motivation

- To establish consolidated body of knowledge about research that using text analysis techniques to improve and evaluate software quality.

A mapping study is conducted to review existing body of knowledge

Process steps

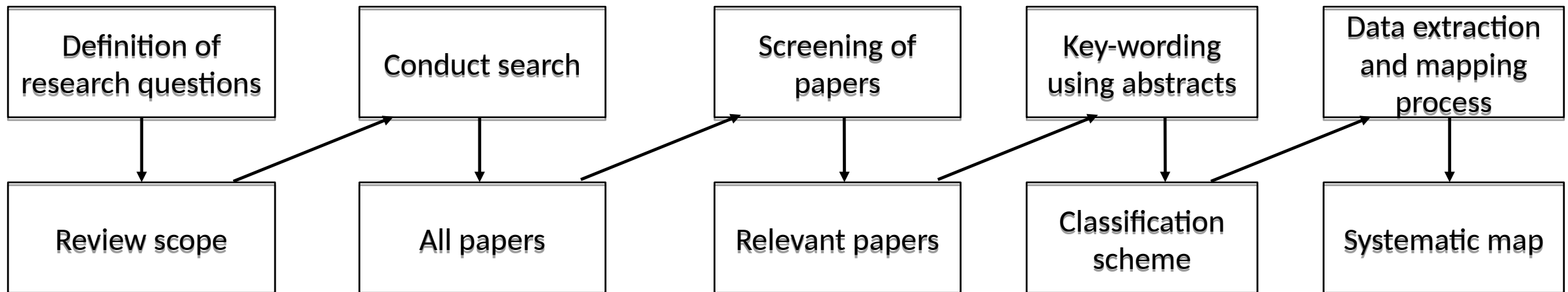
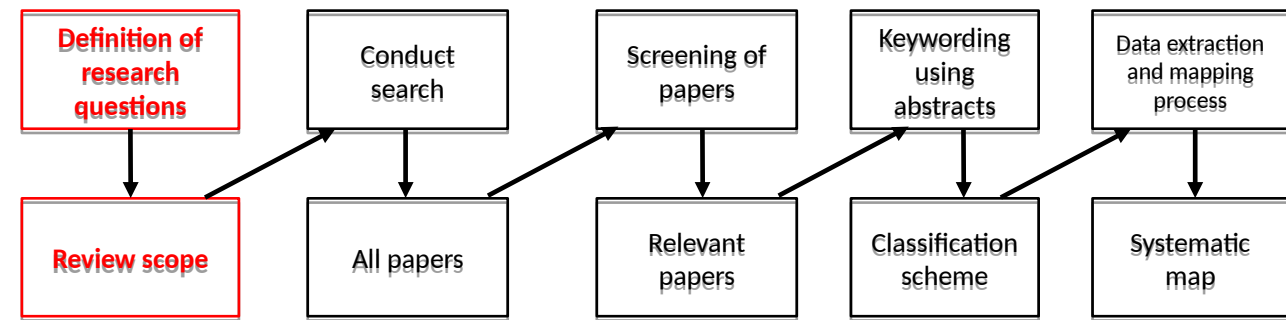


Figure : Mapping Study Process

Research questions?



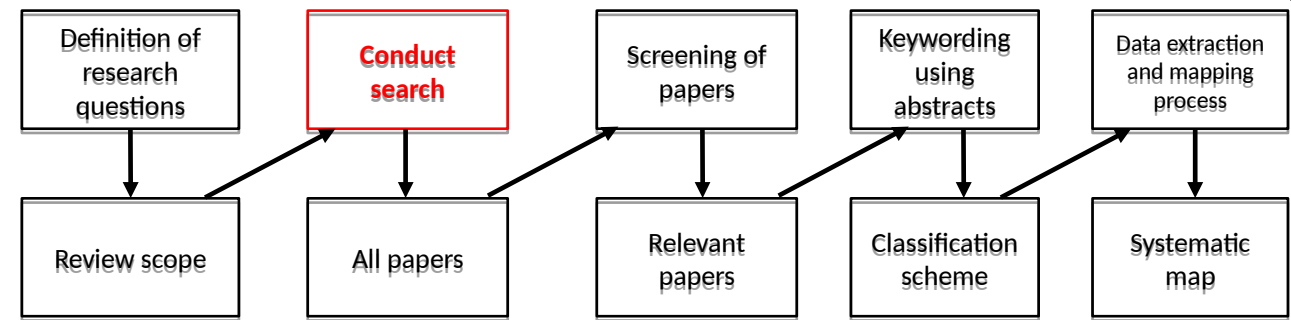
RQ 1: To what data sources have text analysis techniques been applied?

RQ 2 :What types of research papers have been published?

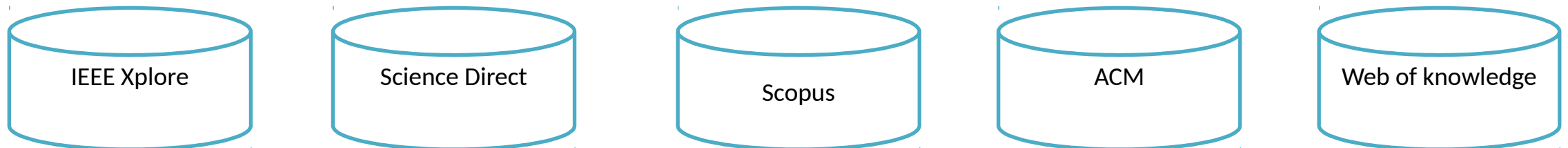
RQ 3: What techniques of text analysis have been used?

RQ 4: How does text analytics help in improving and evaluating software quality?

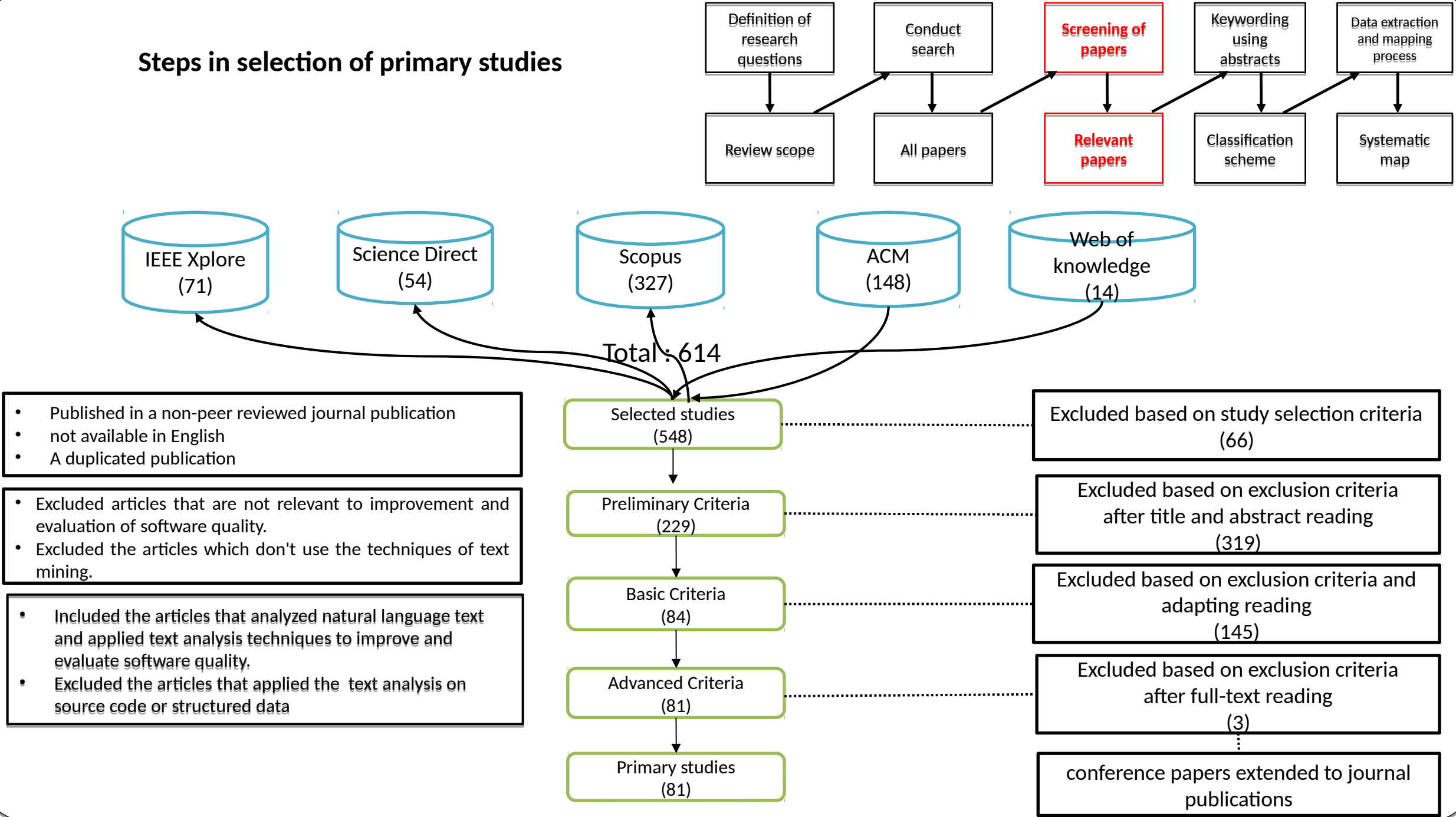
Search string



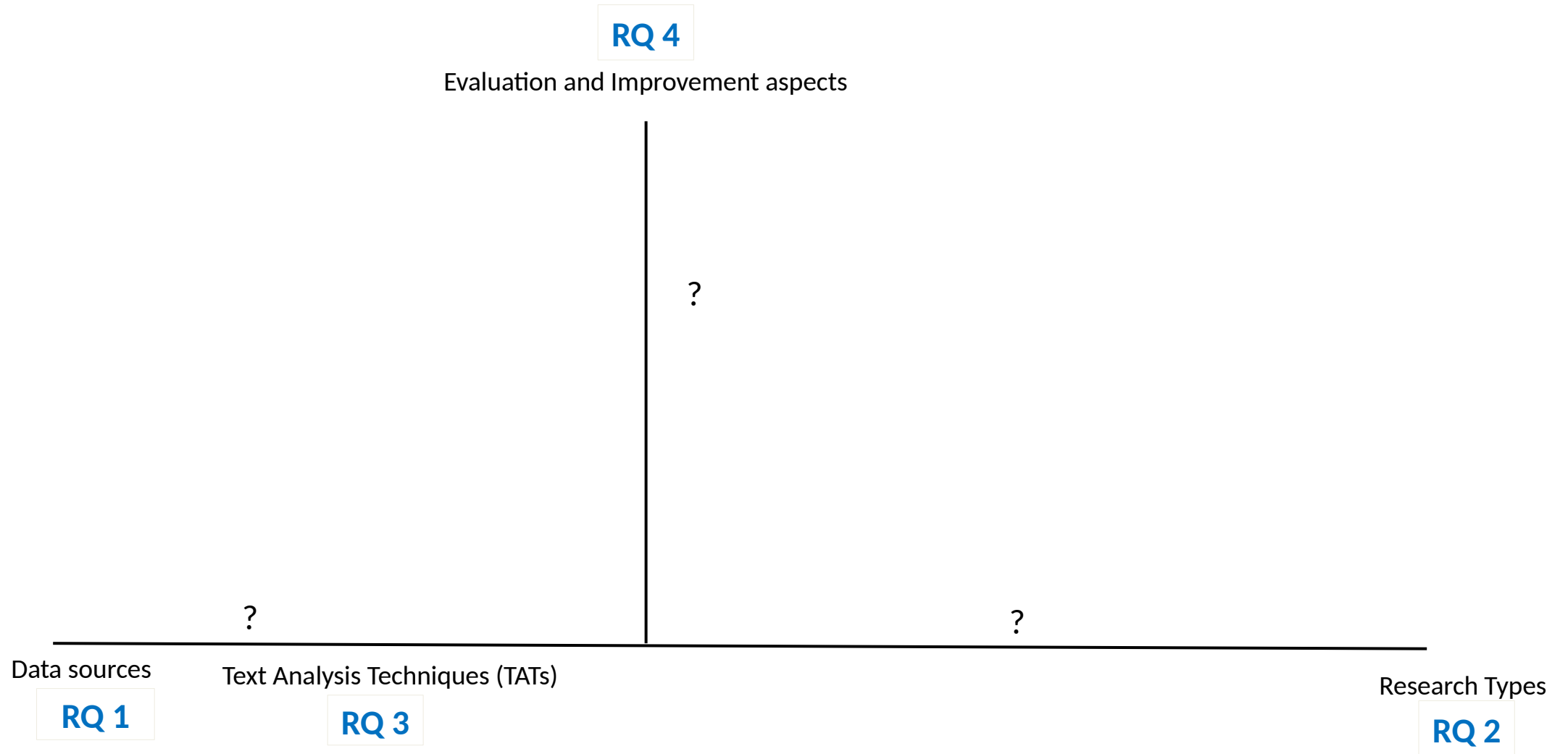
("software quality" OR "software testing" OR "software product quality" OR "software quality analysis" OR "software bugs" OR "software defects" OR "bug reports") AND ("opinion mining" OR "text mining" OR "sentiment analysis" OR "text analysis" OR "text analytics")



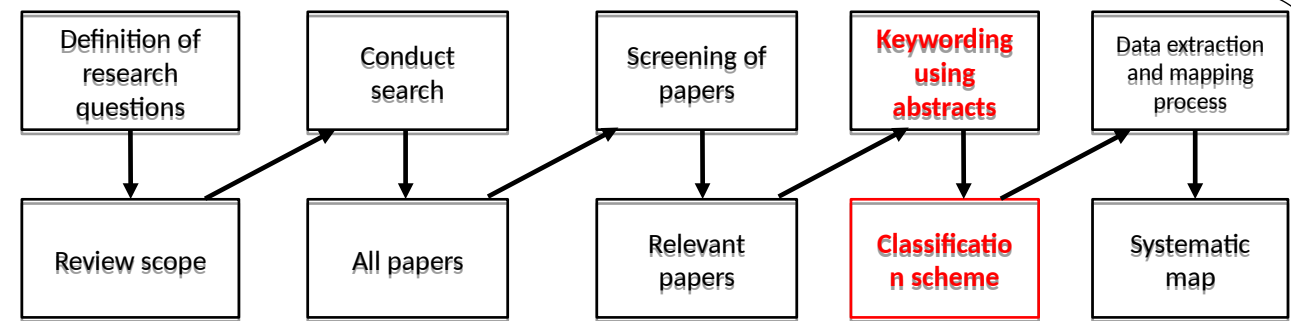
Steps in selection of primary studies



Systematic map visualization



Evaluation and improvement aspects



Use key-wording of abstracts to identity high level concepts

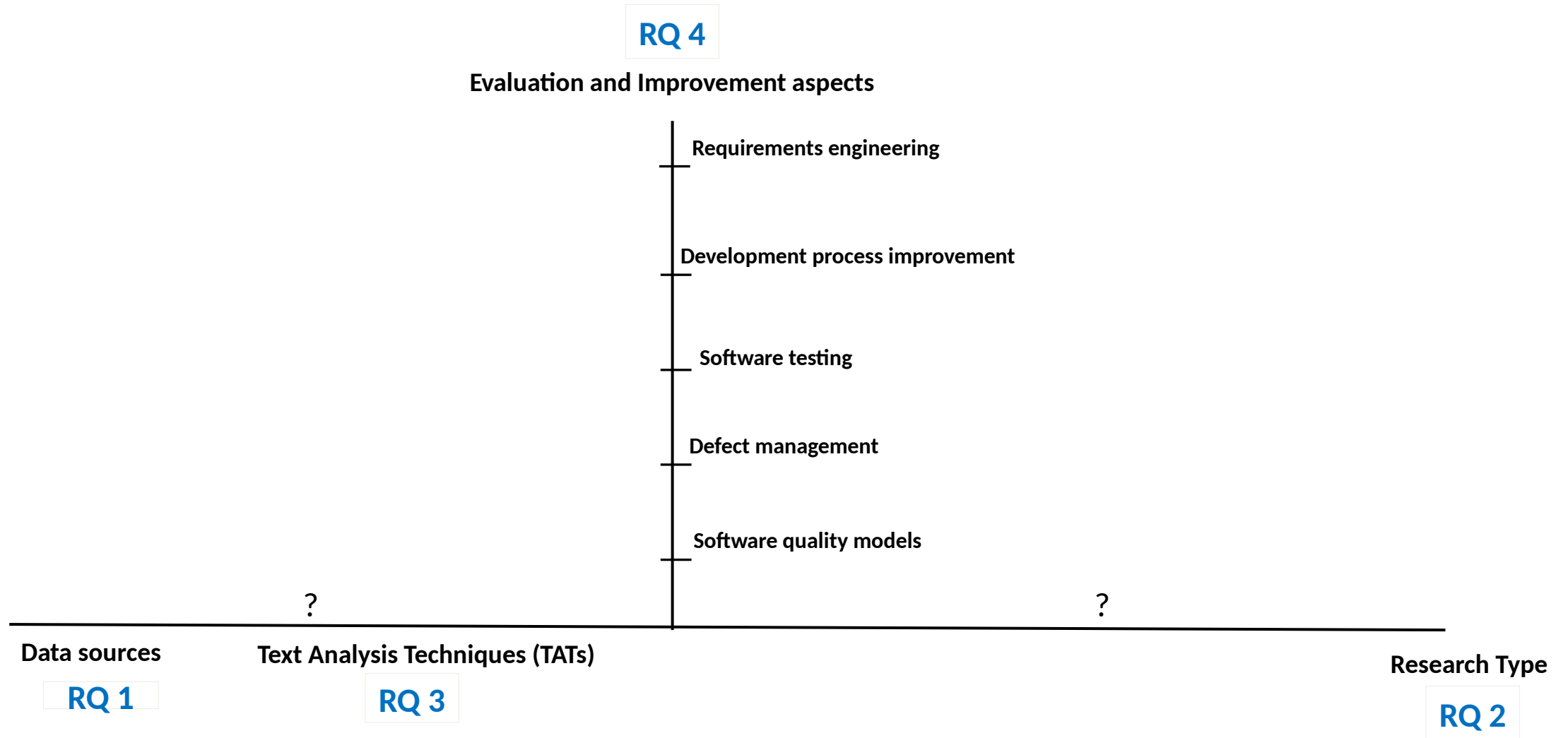
Improvement Aspects

- Software Defect Management
- Requirements Engineering
- Software Testing
- Development Process Improvement

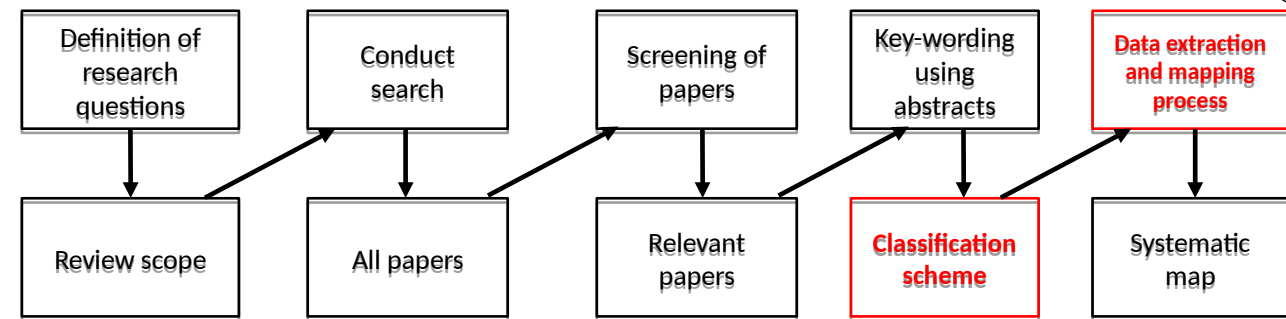
Evaluation Aspect

- Software quality models (ISO/IEC 25010)

Systematic map visualization (updated)



Classification scheme



Data sources (14 data sources)

Text Analysis Techniques

- a) Classification
- b) Clustering
- c) Concept Extraction
- d) Sentiment Analysis
- e) Information Extraction

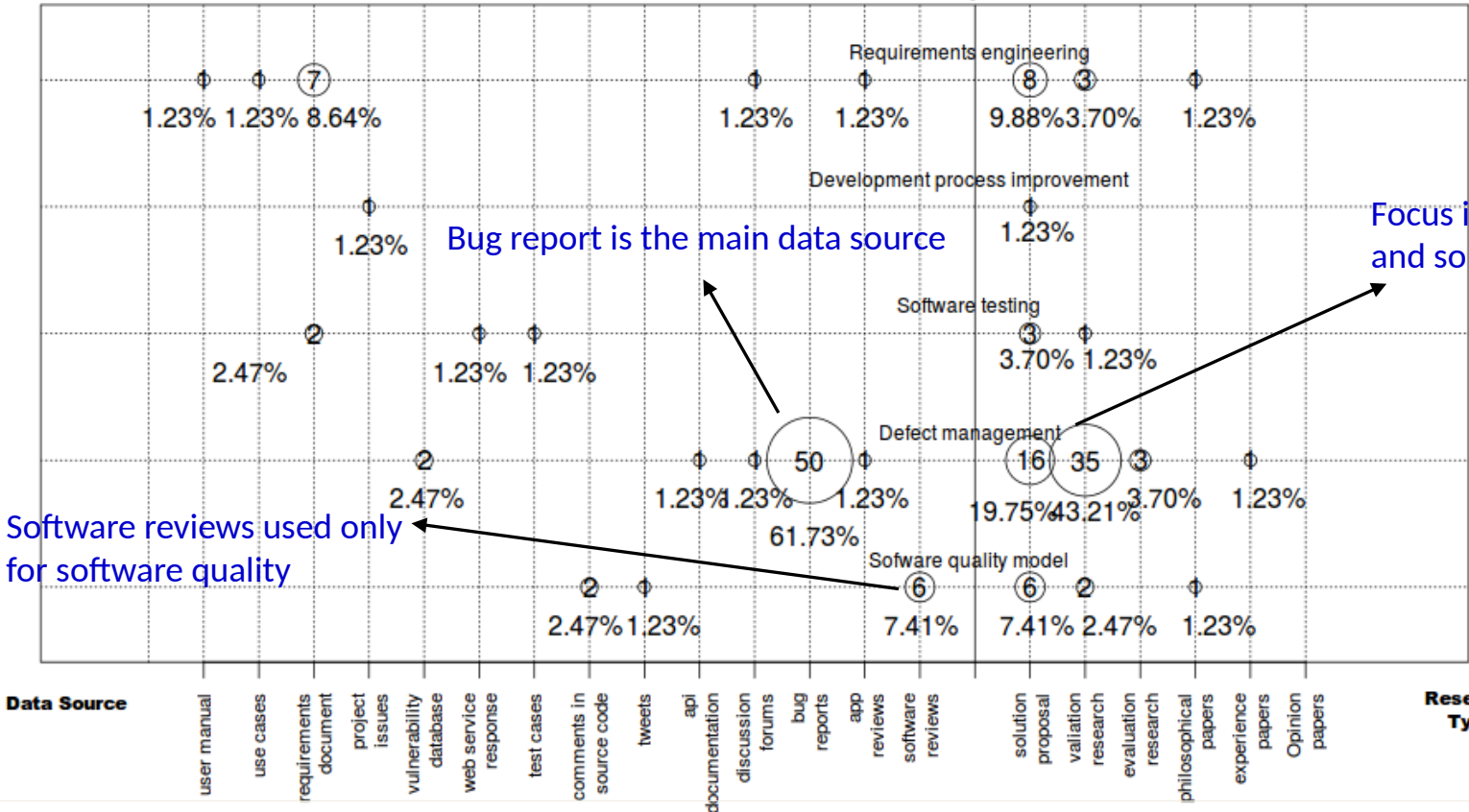
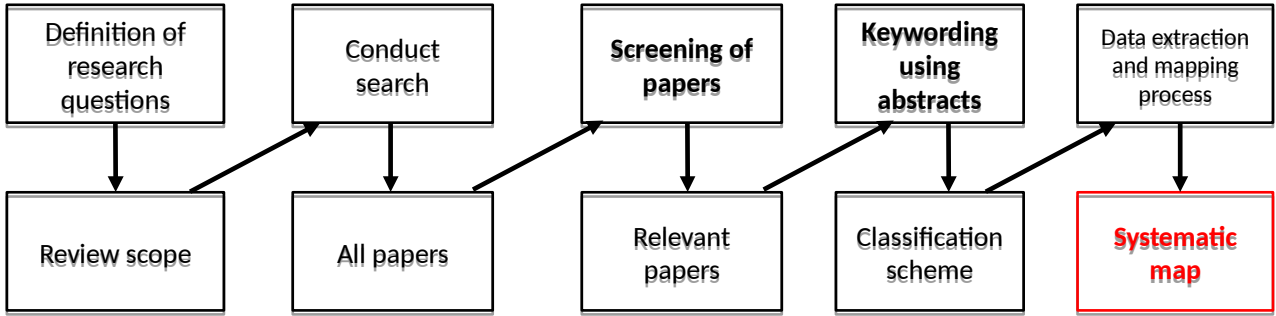
Research type

- a) Solution proposals
- b) Validation research
- c) Evaluation research
- d) Philosophical papers
- e) Opinion papers
- f) Experience papers

Classification scheme is in placed . Now, actual data extraction is performed.

- K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic Mapping Studies in Software Engineering," 12th Int. Conf. Eval. Assess. Softw. Eng., vol. 17, pp. 1-10, 2007.
- G. Miner, Practical text mining and statistical analysis for non-structured text data applications, 2012

Results and Analysis



Overlapping of data
Sources among aspects
Is very rare.

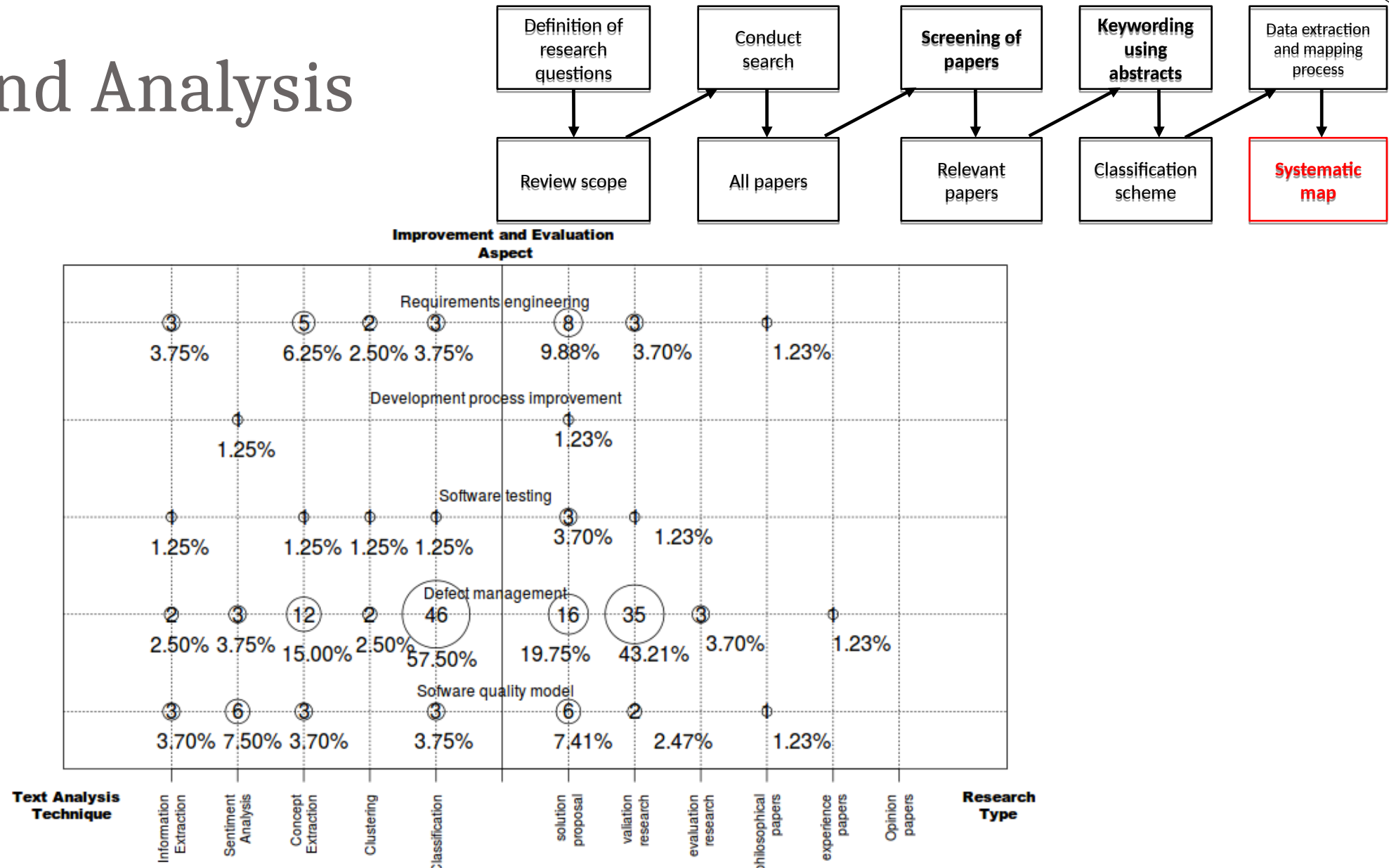
Software reviews used only
for software quality

Focus is on validation research
and solution proposals

Map of research aimed at improving and evaluating different aspects of software quality using text analysis techniques. Improvement and evaluation aspect dimension is on the y-axis. Data source dimension is on the left side of the x-axis and research type on the right side.

Results and Analysis

Classification and concept extraction are frequently mentioned techniques.



Map of research aimed at improving and evaluating different aspects of software quality using text analysis techniques. Improvement and evaluation aspect dimension is on the y-axis. Text analysis technique dimension is on the left side of the x-axis and research type dimension is on the right side.

Results and Analysis (Improvement Aspect)

Table : Activities improved in the defect management aspect

TATs	Activities					
	Bug classification	Bug severity assignment	Bug quality assessment	Bug assignment	Duplicate Bugs	Bug localization
Classification	(P2,P5,P15,P16,P19,P20-P24,P26,P28,P29-P31,P33-P35,P73,P81)	(P4,P36-P48)	(P49)	(P25,P50-P52,P54)	(P55,P57)	(P59-P61)
Clustering	(P18,P32)	-	-	-	-	-
Concept Extraction	(P17,P19,P26,P27,P32,P34)	-	-	(P25,P53)	(P55)	(P59,P62)
Sentiment Analysis	(P28,P29)	-	-	-	-	(P62)
Information Extraction	-	-	-	-	(P57)	(P58)

Bug classification and bug severity assessment activities are mostly Improved using TATs.

Classification and Concept extraction are popular text analysis techniques.

Table : Activities improved in the requirements engineering aspect

TATs	Activities			
	Functional Requirements	Non Functional Requirements	Requirements Evolution	Software Verification
Classification	(P69)	(P75,P80)	-	-
Clustering	(P70)	(P70)	(P78)	-
Concept Extraction	(P71)	(P71,P74,P76,P77)	(P3)	-
Sentiment Analysis	-	-	-	-
Information Extraction	(P70)	(P70,P74)	-	(P79)

Non functional requirements activity Is frequently attempted to improve Using TATs.

Concept extraction is the commonly used TAT.

Results and Analysis (Improvement Aspect)

Table : Activities improved in the software testing aspect

TATs	Activities			
	Static black box test case prioritization	Robustness testing	Test case generation	Bug assignment
Classification	-	(P65)	-	-
Clustering	-	-	-	(P67)
Concept Extraction	(P64)	-	-	-
Sentiment Analysis	-	-	-	-
Information Extraction	-	-	(P66)	-

Results and Analysis (Evaluation aspect)

Table : Quality characteristics evaluted in the software product quality model aspect

TATs	Quality Characteristics							
	Functional	Reliability	Performance	Operability	Security	Compatibility	Maintainability	portability
Classification	-	(P10)	-	(P12)	(P12)	-	-	-
Clustering	-	-	-	-	-	-	-	-
Concept Extraction	-	-	-	-	-	-	-	-
Sentiment Analysis	-	(P10)	-	(P12)	(P12)	-	-	-
Information Extraction	-	-	-	(P13,P14)	-	-	(P11)	-

Studies focused on a small set of quality characteristics.

Quality characteristic **“Operability”** is addressed In 4 out of 5 primary studies.

Table : Quality characteristics evaluted in the software quality in use model aspect

TATs	Quality characteristics				
	Effectiveness	Efficiency	Satisfaction	Safety	Usability
Classification	(P9)	(P9)	(P9)	(P9)	-
Clustering	-	-	-	-	-
Concept Extraction	(P6,P8)	(P6,P8)	(P6)	(P6,P8)	(P6)
Sentiment Analysis	(P6,P8,P9)	(P6,P8,P9)	(P6,P9)	(P6,P8,P9)	(P6,P7,P9)
Information Extraction	-	-	-	-	-

Quality characteristics are equally well covered by the primary studies .

Conclusions

- Bug management aspect is the primary focus of the research.
- Within defect management, text analysis is mostly used for the improvement of bug classification and bug severity assignment activities.
- Other aspects which also gain some attention are requirements engineering and software quality model.
- Quality in use model focuses on evaluation of usability and operability.
- Published research is centered on solution proposals followed by validation research.
- Improvement and evaluation aspect uses different set of text analysis techniques.

Future research directions

- Analyzing app reviews to assess quality characteristics
- “multi-repositories fusion” could become an intriguing research direction.
- Extract “actionable information”
- Evaluation research is required in defect management

Thanks!