On Integrating Orthogonal Information Retrieval Methods to Improve Traceability Recovery

Malcom Gethers, Rocco Oliveto, Denys Poshlyvanyk, Andrea De Lucia
Traceability is...

"the ability to describe and follow the artifact life-cycle"
Traceability in practice

Traceability information is still not commonplace in software projects!
IR-based traceability recovery process
IR-based traceability recovery process
State of the art
State of the art

VSM and Probabilistic Model
State of the art
State of the art

VSM and Probabilistic Model

LSI better than VSM and Probabilistic Model
State of the art

VSM and Probabilistic Model

LSI better than VSM and Probabilistic Model
State of the art

VSM and Probabilistic Model

LSI better than VSM and Probabilistic Model

JS and Lucene better than LSI
State of the art

VSM and Probabilistic Model

LSI better than VSM and Probabilistic Model

JS and Lucene better than LSI
State of the art

- VSM and Probabilistic Model
- LSI better than VSM and Probabilistic Model
- JS and Lucene better than LSI
- LDA better than LSI
State of the art

VSM and Probabilistic Model

LSI better than VSM and Probabilistic Model

Constrasting results! No clear winner among the IR methods

JS and Lucene better than LSI

DA better LSI
Our previous study...

On the Equivalence of Information Retrieval Methods for Automated Traceability Link Recovery

Russo Oliveto 1, Malcolm Gettys 2, Denys Poshyvanyk 1, Andrea De Lucia 1
1Department of Mathematics and Informatics, University of Salerno, via Ponte don Melillo, Fisciano (SA), Italy
2Computer Science Department, The College of William and Mary, Williamsburg, VA 23185, USA
reoliveto@unisa.it, gettys@cs.wm.edu, deny@cs.wm.edu, adelucia@unisa.it

Abstract—We present an empirical study to statistically analyze the equivalence of several traceability recovery methods based on Information Retrieval (IR) techniques. The analysis is based on Principal Component Analysis and on the analysis of the overlap of the set of candidate links provided by each method. The studied techniques are the Jansen-Shannon (JS) method, Vector Space Model (VSM), Latent Semantic Indexing (LSI), and Latent Dirichlet Allocation (LDA). The results show that while JS, VSM, and LSI are almost equivalent, LDA is able to capture a dimension unique to the set of techniques which we considered.

Keywords—Traceability Recovery; Information Retrieval; Empirical Studies.

I. INTRODUCTION

Extensive effort in the software engineering community (both research and commercial) has been brought forth to improve the explicit connection of documentation and source code. Promising results have been achieved using Information Retrieval (IR) techniques [1, 2] for traceability recovery (e.g., [3], [4]). IR-based methods propose a list of candidate traceability links on the basis of the similarity between the text contained in the software artifacts. Such methods are based on the conjecture that if two artifacts sharing high textual similarity share several concepts thus they are good candidates to be traced on each other.

Several IR methods have been proposed for traceability recovery — e.g., Vector Space Model (VSM) and probabilistic models [1] or Latent Semantic Indexing (LSI) [2]. In general, the retrieval accuracy of IR-based traceability recovery methods is assessed through two measures: recall, measuring the percentage of correct links that were found, and precision, measuring the percentage of found links that were correct. The results achieved are sometimes contrasting and demonstrate no clear winner among IR techniques. Indeed, it seems that all the exploited techniques so far are able to capture the same information when used to calculate the textual similarity between software artifacts.

In this paper we present an empirical study aiming at statistically analyzing the equivalence of different IR-based traceability recovery methods. The comparison is based on Principal Component Analysis (PCA) and on the analysis of the overlap of the set of candidate links provided by each of the IR methods. The studied IR techniques are the Jansen-Shannon (JS) method [5], VSM, LSI, and Latent Dirichlet Allocation (LDA) [6]. The first three methods were selected because they are widely used and seem to be the techniques that give the best results [5], [3], [4]. LDA is not as widely used for traceability link recovery though it has been used recently [16]. However, we also experiment such a technique for traceability recovery because LDA is able to capture some aspects missed by other IR methods, such as LSI, when it is used in other contexts [7].

The empirical analysis has been conducted on two software repositories, namely EasyClinic and eTours. The studied IR methods have been used to recover traceability links between the use cases and the source code of the two software systems. The results prove that the accuracy of LDA is lower than previously used methods. However, while JS, VSM, and LSI are almost equivalent, LDA is able to capture some aspects missed by other IR methods. These considerations suggest that probably LDA can be used as a method to augment canonical methods—e.g., JS, VSM, and LSI—aiming at improving their accuracy.

The rest of the paper is organized as follows. Section II discusses related work, while Section III briefly describes an IR-based traceability recovery process. Sections IV and V provide details on the design of the case study and report the results achieved, respectively. Section VI gives concluding remarks.

II. RELATED WORK

The use of IR methods for traceability recovery was introduced by Antoniol et al. [3]. They apply probabilistic and Vector Space Models [1] to trace source code onto software documentation. Later, other IR methods (e.g., LSI, JS method and Numerical Analysis) have been proposed to recover links between different types of artifacts [4], [5], [8]. In particular, IR methods have been used to recover traceability between requirements [9], between requirements and design artifacts [10], between maintenance requests and software documents [11], and between other types of artifacts (e.g., use cases and UML diagrams) [12], [13].

All these reported case studies compare different IR-based traceability recovery approaches using recall and precision.

Which is the best IR method?

Is the ranking of these methods different?

mercoledì 12 ottobre 11
Our previous study...

Which is the best IR method?

Is the ranking of these methods different?

LDA had lowest accuracy

LDA is orthogonal as compared to VSM, JS, and LSI
Motivation

Is possible to improve the accuracy of topic-based model?

Is it still possible to capture with other topic-based models information missed by other IR models?
Relational Topic Model (RTM)

Is possible to improve the accuracy of topic-based model?
Relational Topic Model (RTM)

Is possible to improve the accuracy of topic-based model?

EasyClinic (Eng)

EasyClinic (Ita)

eTour (Eng)

JS

VSM

RTM
Relational Topic Model (RTM)

Is possible to improve the accuracy of topic-based model?

RTM provides comparable accuracy
Orthogonality of Techniques

Is still possible to capture with other topic-based models information missed by other IR models?
Orthogonality of Techniques

Is still possible to capture with other topic-based models information missed by other IR models?
**Orthogonality of Techniques**

<table>
<thead>
<tr>
<th></th>
<th>25%</th>
<th>50%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>correctINT(JS, VSM)</td>
<td>91%</td>
<td>94%</td>
<td>85%</td>
</tr>
<tr>
<td>correctDIFF(JS, VSM)</td>
<td>4%</td>
<td>5%</td>
<td>7%</td>
</tr>
<tr>
<td>correctDIFF(VSM, JS)</td>
<td>4%</td>
<td>0%</td>
<td>7%</td>
</tr>
<tr>
<td>correctINT(JS, RTM)</td>
<td>23%</td>
<td>28%</td>
<td>35%</td>
</tr>
<tr>
<td>correctDIFF(JS, RTM)</td>
<td>41%</td>
<td>35%</td>
<td>22%</td>
</tr>
<tr>
<td>correctDIFF(RTM, JS)</td>
<td>35%</td>
<td>36%</td>
<td>41%</td>
</tr>
<tr>
<td>correctINT(VSM, RTM)</td>
<td>23%</td>
<td>29%</td>
<td>32%</td>
</tr>
<tr>
<td>correctDIFF(VSM, RTM)</td>
<td>41%</td>
<td>32%</td>
<td>24%</td>
</tr>
<tr>
<td>correctDIFF(RTM, VSM)</td>
<td>35%</td>
<td>38%</td>
<td>42%</td>
</tr>
</tbody>
</table>

Is still possible to capture with other topic-based models information missed by other IR models?
**Orthogonality of Techniques**

<table>
<thead>
<tr>
<th></th>
<th>25</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>correctINT(JS,VSM)</td>
<td>91%</td>
<td>94%</td>
<td>85%</td>
</tr>
<tr>
<td>correctDIFF(JS,VSM)</td>
<td>4%</td>
<td>5%</td>
<td>7%</td>
</tr>
<tr>
<td>correctDIFF(VSM,JS)</td>
<td>4%</td>
<td>0%</td>
<td>7%</td>
</tr>
<tr>
<td>correctINT(JS,RTM)</td>
<td>23%</td>
<td>28%</td>
<td>35%</td>
</tr>
<tr>
<td>correctDIFF(JS,RTM)</td>
<td>41%</td>
<td>35%</td>
<td>22%</td>
</tr>
<tr>
<td>correctDIFF(RTM,JS)</td>
<td>35%</td>
<td>36%</td>
<td>41%</td>
</tr>
<tr>
<td>correctINT(VSM,RTM)</td>
<td>23%</td>
<td>29%</td>
<td>32%</td>
</tr>
<tr>
<td>correctDIFF(VSM,RTM)</td>
<td>41%</td>
<td>32%</td>
<td>24%</td>
</tr>
<tr>
<td>correctDIFF(RTM,VSM)</td>
<td>35%</td>
<td>38%</td>
<td>42%</td>
</tr>
</tbody>
</table>

Is still possible to capture with other topic-based models information missed by other IR models?
Is still possible to capture with other topic-based models information missed by other IR models?

<table>
<thead>
<tr>
<th></th>
<th>25</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>correctINT(JS, VSM)</td>
<td>91%</td>
<td>94%</td>
<td>85%</td>
</tr>
<tr>
<td>correctDIFF(JS, VSM)</td>
<td>4%</td>
<td>5%</td>
<td>7%</td>
</tr>
<tr>
<td>correctDIFF(VSM, JS)</td>
<td>4%</td>
<td>0%</td>
<td>7%</td>
</tr>
<tr>
<td>correctINT(JS, RTM)</td>
<td>23%</td>
<td>28%</td>
<td>35%</td>
</tr>
<tr>
<td>correctDIFF(JS, RTM)</td>
<td>41%</td>
<td>35%</td>
<td>22%</td>
</tr>
<tr>
<td>correctDIFF(RTM, JS)</td>
<td>35%</td>
<td>36%</td>
<td>41%</td>
</tr>
<tr>
<td>correctINT(VSM, RTM)</td>
<td>23%</td>
<td>29%</td>
<td>32%</td>
</tr>
<tr>
<td>correctDIFF(VSM, RTM)</td>
<td>41%</td>
<td>32%</td>
<td>24%</td>
</tr>
<tr>
<td>correctDIFF(RTM, VSM)</td>
<td>35%</td>
<td>38%</td>
<td>42%</td>
</tr>
</tbody>
</table>
Orthogonality of Techniques

<table>
<thead>
<tr>
<th></th>
<th>25%</th>
<th>50%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>correctINT(JS,VSM)</td>
<td>91%</td>
<td>94%</td>
<td>85%</td>
</tr>
<tr>
<td>correctINT(VSM,RTM)</td>
<td>4%</td>
<td>5%</td>
<td>7%</td>
</tr>
<tr>
<td>correctINT(RTM,JS)</td>
<td>0%</td>
<td>7%</td>
<td>7%</td>
</tr>
<tr>
<td>correctDIFF(JS,VSM)</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
</tr>
<tr>
<td>correctDIFF(VSM,RTM)</td>
<td>23%</td>
<td>29%</td>
<td>24%</td>
</tr>
<tr>
<td>correctDIFF(RTM,JS)</td>
<td>23%</td>
<td>28%</td>
<td>41%</td>
</tr>
<tr>
<td>correctDIFF(RTM,VSM)</td>
<td>41%</td>
<td>41%</td>
<td>42%</td>
</tr>
</tbody>
</table>

RTM provides orthogonal results

Is still possible to capture with other topic-based models information missed by other IR models?
Is still possible to capture with other topic-based models information missed by other IR models?
Is still possible to capture with other topic-based models information missed by other IR models?
Other Results

There is a statistically significant interaction between IR method and artifact type.

There is a statistically significant interaction between IR method and language.
Conclusions

RTM provides accuracy comparable to existing IR techniques

RTM is able to capture orthogonal information

RTM combined with other IR techniques provides improved accuracy for traceability recovery
Conclusions

Thank You!

Questions?