IR in Software Traceability: From a Bird's Eye View

Markus Borg
Dept. of Computer Science
Lund University, Sweden
Email: markus.borg@cs.lth.se

Per Runeson
Dept. of Computer Science
Lund University, Sweden
Email: per.runeson@cs.lth.se

***Abstract*—Several researchers have proposed creating after-the-fact structure among software artifacts using trace recovery based on Information Retrieval (IR) approaches. Due to significant variation points in previous studies, results are not easily aggregated. We provide an initial overview picture of the outcome of previous evaluations. Based on a systematic mapping study, we perform a synthesis of published research. Our results show that there are no empirical evidence that any IR model outperforms another model consistently. We also display a strong dependency between the P-R values and the input datasets. Finally, our mapping of Precision and Recall (P-R) values on the possible output space highlights the difficulty of recovering accurate trace links using naïve cut-off strategies. Thus, our work presents empirical evidence that confirms several previous claims on IR-based trace recovery and stresses the needs for empirical evaluations beyond the basic P-R "race".**

***Keywords*—empirical software engineering; software traceability; information retrieval; secondary study**

I. Introduction

Software engineering is a knowledge-intensive activity, generating much information that needs to be maintained during software evolution. One state-of-practice way to structure software artifacts is to maintain *traceability*, defined as "the potential for traces to be established and used" [10] where a trace (or trace link) is "an association forged between two artifacts", representing a relation such as overlap, dependency, contribution, evolution, refinement, or conflict. To support maintenance of trace links, several researchers have proposed tool support for semi-automated *trace recovery*, i.e. proposing candidate trace links among existing artifacts. One well-studied category of tools are based on *Information Retrieval* (IR) approaches [7].

Traditional IR evaluation consists of three main elements: a document collection, a set of information needs (typically formulated as queries), and relevance judgments telling what documents are relevant to these information needs, i.e. a *gold standard*. The TREC conference[1] has been driving the state-of-the-art in traditional IR by providing large-scale evaluations. The most common way to report the effectiveness of an IR model is to use the measures *precision* and *recall*, which also applies to IR-based trace recovery. The outcome is often visualized as a Precision-Recall (P-R) curve where the average precision is plotted at fixed recall values.

We have previously conducted a Systematic Mapping (SM) study on IR-based trace recovery [4]. Our comprehensive study, conducted according to the guidelines by Kitchenham and Charters [12], contains 79 publications and the corresponding empirical data from a decade of research effort. The SM shows that evaluations on IR-based trace recovery has been dominated by technology-oriented experimentation, and are often limited to reporting P-R values.

In this paper we synthesize results from technology-oriented primary studies identified in the SM, and provide an empirical overview. Our work is guided by the following research question "Based on the published empirical data from technology-oriented experiments, what can we conclude regarding IR models and datasets?"

II. Method

We used our previous SM on IR-based trace recovery, containing studies until 2011, as input to this secondary study. The SM identified two principally different experimental setups to conduct technology-oriented evaluations of IR-based trace recovery tools, *query-based* and *matrix-based* evaluation [4]. Query-based evaluation implies that a number of queries are executed on a document set, and each query results in a ranked list of search results. In matrix-based evaluations, the result is reported as one single ranked list of candidate trace links, and the outcome is a candidate traceability matrix.

Furthermore, the primary studies differ by which sets of P-R values are reported. In addition to the traditional way of reporting precision at fixed levels of recall, different strategies for selecting subsets of candidate trace links have been proposed, i.e. different *cut-off strategies*. While there are significant variation points in previous evaluations, taking a step back to view previous results from a new perspective might reveal general trends.

We synthesize the empirical evaluations in two separate ways. First, regarding publications comparing multiple underlying IR models, we aggregate the conclusions of the original authors. We apply unweighted *vote counting analysis* [13] to explore which IR models have been considered the most successful in previous research. As a second approach to data synthesis, we *aggregate P-R values* extracted from previous studies.

While IR-based trace recovery is an IR application with its own unique characteristics, we extract P-R values that cover what is standard procedure in the general IR field. In line with what is reported at TREC, we report precision at ten recall levels from 0.1 to 1 (referred to as PR@Fix). However, while TREC established the cut-off levels 5, 10, 15, 20, 30 and 100, evaluations on IR-based trace recovery have typically not been reported at such a level of detail. As a consequence, we report

[1] trec.nist.gov

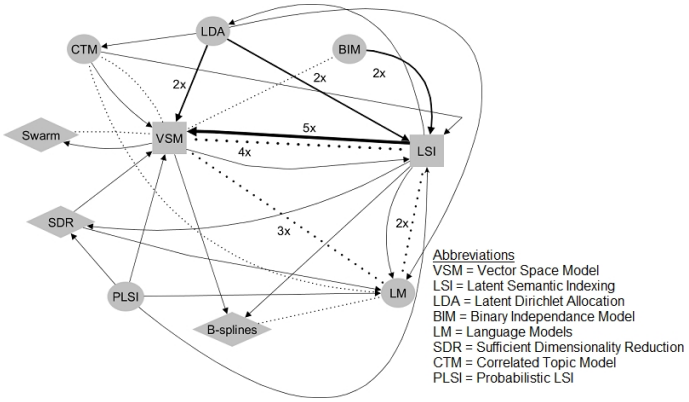

Fig. 1. Twenty-five empirical comparisons of IR models in trace recovery. Squares show algebraic models, circles represent probabilistic models, diamonds show other models. Edges point toward the better model.

P-R values from only the cut-off levels 5, 10 and 100 (referred to as PR@N).

Moreover, neither PR@Fix nor PR@N cover all P-R reporting styles in the primary publications. Thus, we also extracted P-R values beyond standard TREC practice. As several previous publications report P-R values corresponding to a set of candidate trace links with a cosine similarity ≥ 0.7 (cosine similarity is a standard way of quantifying textual similarity [14]), we extracted the corresponding P-R values (PR@Sim0.7). Finally, to ensure that all primary publications reporting P-R values contributed to our synthesis, we also report from an inclusive aggregation of all encountered P-R values (referred to as PR@Tot). This final aggregation of data shows the entire span of reported P-R values, not taking any evaluation variations into account, and thus displays an inclusive snapshot of the output space of an active research field.

III. Results and discussion

Among the 79 publications in the SM, 25 compare the output accuracy of trace recovery when applying different IR models. Figure 1 depicts the outcomes of the comparisons, based on the original authors' conclusions. An edge represents a comparison of two implemented IR models on a dataset, thus a single publication can introduce several arrows. The direction of an edge points at the IR model that produced the most accurate output, i.e. an arrow points at the better model. Accordingly, an undirected edge (dotted) shows inconclusive comparisons. Finally, an increased edge weight depicts multiple comparisons between the IR models, also presented as a label on the edge. For descriptions of the various IR models, we refer to the SM [4].

VSM and LSI are the two IR models that have been most frequently evaluated in comparing studies. Among those, and among comparing studies in general, VSM has presented the best results. On the other hand, implementing language models and measuring similarities using Jensen-Shannon divergence [5] has been concluded as a better technique in four studies and has never underperformed any other models. As it has been compared to VSM in three publications, it appears to perform trace recovery with a similar accuracy [1], [15], [9]. Also conducting retrieval based on B-splines and swarm techniques has not been reported to perform worse than other models, but has only been explored in two primary publications. Notably, one of the commonly applied IR models, the probabilistic inference network, has not been explicitly studied in comparison to other IR models within the context of trace recovery. Traceability to individual publications is provided on the accompanying website[2].

Figure 2 gives a general idea of the accuracy of candidate trace links from previous work, aggregating results from the 48 papers in the SM reporting P-R values. In the upper right corners of figures, constituting the ideal output accuracy of an IR-based trace recovery tool, we show intervals representing 'Excellent', 'Good', and 'Acceptable' as proposed by Huffman Hayes *et al.* [11]. The quality intervals were developed as an attempt to "draw a line in the sand", based on informal industrial experiences rather than solid empirical studies. While it is unclear in several publications whether a query-based or matrix-based evaluation style has been used, it is apparent that a majority of the P-R values in PR@5/10/100 originate from query-based evaluations, and that the P-R values in PR@Sim0.7/Fix/Tot are dominated by matrix-based evaluations. The P-R values reported in this section correspond to the following approaches: constant cut-point (PR@N) 12 publications, constant threshold (PR@Sim0.7) 11 publications, fixed levels of recall (PR@Fix) 10 publications, and other approaches in 24 publications (presented, together with the other approaches, in PR@Tot).

Figure 2 shows P-R footprints from trace recovery evaluations with constant cutpoints at 5, 10, and 100 candidate trace links respectively. Evaluations on five datasets (LEDA, CM-1, Gedit, Firefox, and ArgoUML) are marked with separate symbols, as shown in the legend. Especially for PR@5 and PR@10, the primary publications contain several empirical results from trace recovery on these datasets. The P-R values in PR@5, PR@10 and PR@100 represent evaluations using: LDA (24 results), LSI (19 results), BM25 (18 results), VSM (14 results), LM (12 results), BIM (7 results), PLSI (6 results), and SDR (6 results).

No clear pattern related to the IR models can be observed in Figure 2. Instead, *different* implementations performed *similarly* when applied to the *same* datasets. This is particularly evident for evaluations on LEDA, for which we could extract several P-R values (shown as squares in Figure 2). In the footprint PR@5, nine P-R values from four different research groups implementing VSM, BIM, and LSI cluster in the lower right corner. Also, the footprint PR@10 shows five results from evaluations on LEDA, clustered in the very right corner, corresponding to P-R values from three different research groups implementing VSM, BIM, and LSI. In line with the results on LEDA, PR@5/10/100 show clustered P-R values from trace recovery using different configurations of BM25 on the Firefox dataset (diamonds) and Gedit (triangles). Similar results can be seen regarding evaluations on CM-1 (circles) in PR@5 and ArgoUML (pluses) in PR@100. However, results on CM-1 in PR@10/100 and ArgoUML in PR@5/10 are less clear as they display lower degrees of clustering.

In the footprint PR@Sim0.7 in Figure 2, showing P-R

[2]sites.google.com/site/tracerepo/

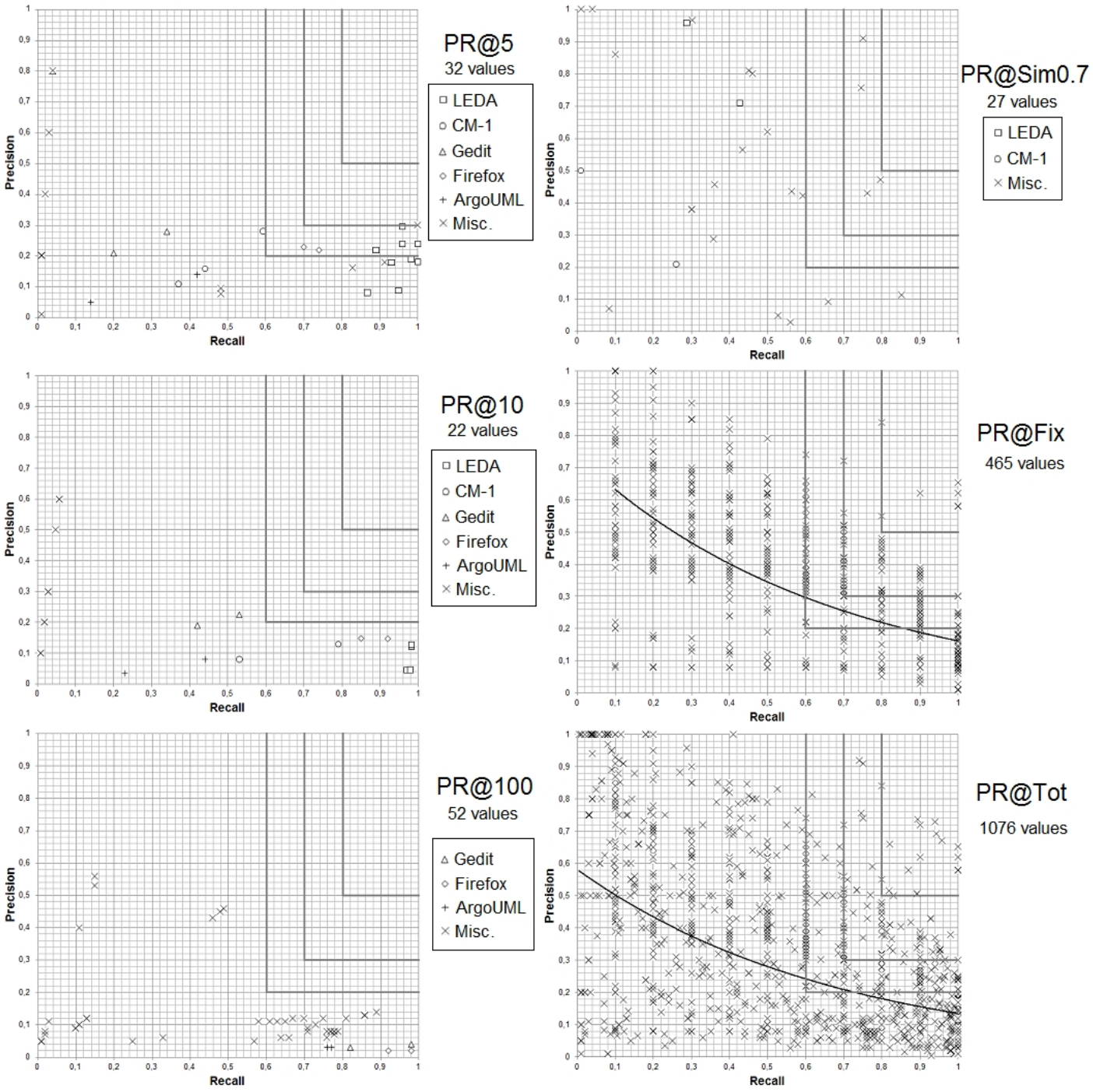

Fig. 2. P-R footprints for IR-based trace recovery tools. The figures to the left show P-R values at the constant cut-offs PR@5, PR@10 and PR@100. The figures to the right show P-R values representing a cut-off at the cosine similarity 0.7 (PR@Sim0.7), precision at fixed recall levels (PR@Fix), and an aggregation of all collected P-R values (PR@Tot). The figures PR@Fix and PR@Tot also present a P-R curve calculated as an exponential trendline.

values corresponding to candidate trace links with a cosine similarity of ≥ 0.7, P-R values are located in the entire P-R space, displaying its inappropriateness as a generic cut-point. In PR@Fix, the expected P-R tradeoff is evident as shown by the trendline. Several primary publications report both 'acceptable' and 'good' P-R values. Six P-R values are even reported within the 'excellent' zone, all originating from evaluations of trace recovery based on LSI. However, all six evaluations were conducted on datasets containing around only 150 artifacts, CoffeeMaker (5 results) and EasyClinic (1 result).

In PR@Tot, we show 1,076 P-R values from 48 publications. In total, we extracted 270 P-R values (25.2%) within the 'acceptable' zone, 129 P-R values (12.0%) in the 'good' zone, and 19 P-R values (1.8%) in the 'excellent' zone. The average (balanced) F-measure for the P-R values in PR@Tot is 0.31 with a standard deviation of 0.07. As this average F-measure is higher than the F-measure of the lowest 'acceptable' P-R value (0.24, corresponding to recall=0.6, precision=0.2), this reflects the difficulty in achieving reasonably balanced precision and recall in IR-based trace recovery. Also, among the 100 P-R values with the highest F-measure in PR@Tot, 69 have been reported when evaluating trace recovery on the EasyClinic dataset, extracted from 9 different publications.

Our synthesis of 25 comparative studies on IR-based trace recovery show that there is no empirical evidence that any IR model outperforms another consistently, wrt. the accuracy of the candidate trace links. Hence, our results confirm previous findings by Oliveto *et al.* [15] and Binkley *et al.* [3]. Instead, our results suggest that the classic VSM performs better or as good as other models. Our findings are also in line with the claim by Falessi *et al.* [8], that simple IR techniques are typically the most useful. Thus, as also pointed out by Ali *et al.* [2], there appears to be little value for the traceability community to continue publishing studies that solely hunt improved P-R values without considering other factors that impact trace recovery, e.g. the validity of the dataset and the specific work task the tools are intended to support.

The synthesized P-R values highlights the evident challenge of reaching 'acceptable' precision and 'acceptable' recall, as it is only achieved in about a quarter of the reported P-R values. Some published results are 'acceptable', a few are even 'good' or 'excellent', while a majority of the results are 'unacceptable'. While the appropriateness of the proposed quality levels cannot be validated without user studies, we acknowledge them as a starting point for the synthesis of empirical results. On the other hand, as Cuddeback *et al.* [6] rather controversially highlighted, human subjects vetting entire candidate traceability matrices do not necessarily benefit from more accurate candidate trace links. Thus, there is a need for more empirical work on how humans interact with the tool output to validate the quality levels proposed by Huffman Hayes and Dekhtyar [11], and to understand in which contexts they are applicable.

IV. Summary

We have synthesized the empirical results from a previously conducted SM on trace recovery between software artifacts, using IR approaches. We identified that there are two principally different evaluation approaches, query-based and matrix-based evaluations. Also, there are a significant number of variations regarding which P-R values on model performance are reported. Based on the primary studies from our previous SM, vote counting shows that there is no evidence of any IR model consistently outperforming another. On the contrary, we see a clear interaction between IR model and the data used for the empirical evaluation. Further, the general level of P-R values is rather low, which may or may not be critical, depending on the use situation of the recovered traces. As a consequence, we propose that further research on trace recovery should be conducted in more realistic use situations, rather than hunting small improvements in the "race" for P-R values on synthetic benchmarks.

References

[1] A. Abadi, M. Nisenson, and Y. Simionovici. A traceability technique for specifications. In *Proceedings of the 16th International Conference on Program Comprehension*, pages 103–112, 2008.

[2] N. Ali, Y-G. Guéhéneuc, and G. Antoniol. Factors impacting the inputs of traceability recovery approaches. In J. Cleland-Huang, O. Gotel, and A. Zisman, editors, *Software and Systems Traceability*. Springer, 2012.

[3] D. Binkley and D. Lawrie. Information retrieval applications in software maintenance and evolution. In J. Marciniak, editor, *Encyclopedia of software engineering*. Taylor & Francis, 2 edition, 2010.

[4] M. Borg, P. Runeson, and A. Ardö. Recovering from a decade: A systematic mapping of information retrieval approaches to software traceability. *Empirical Software Engineering*, 2013.

[5] S-H. Cha. Comprehensive survey on Distance/Similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 4(1):300–307, 2007.

[6] D. Cuddeback, A. Dekhtyar, and J. Huffman Hayes. Automated requirements traceability: The study of human analysts. In *Proceedings of the 18th International Requirements Engineering Conference*, pages 231–240, 2010.

[7] A. De Lucia, A. Marcus, R. Oliveto, and D. Poshyvanyk. Information retrieval methods for automated traceability recovery. In J. Cleland-Huang, O. Gotel, and A. Zisman, editors, *Software and Systems Traceability*. Springer, 2012.

[8] D. Falessi, G. Cantone, and G. Canfora. A comprehensive characterization of NLP techniques for identifying equivalent requirements. In *Proceedings of the International Symposium on Empirical Software Engineering and Measurement*, 2010.

[9] M. Gethers, R. Oliveto, D. Poshyvanyk, and A. De Lucia. On integrating orthogonal information retrieval methods to improve traceability recovery. In *IEEE International Conference on Software Maintenance, ICSM*, pages 133–142, 2011.

[10] O. Gotel, J. Cleland-Huang, J. Huffman Hayes, A. Zisman, A. Egyed, P. Grünbacher, A. Dekhtyar, G. Antoniol, J. Maletic, and P. Mader. Traceability fundamentals. In J. Cleland-Huang, O. Gotel, and A. Zisman, editors, *Software and Systems Traceability*, pages 3–22. Springer, 2012.

[11] J. Huffman Hayes, A. Dekhtyar, and S. Sundaram. Advancing candidate link generation for requirements tracing: the study of methods. *Transactions on Software Engineering*, 32(1):4–19, 2006.

[12] B. Kitchenham and S. Charters. Guidelines for performing systematic literature reviews in software engineering. *EBSE Technical Report*, 2007.

[13] R. Light and P. Smith. Accumulating evidence: Procedures for resolving contradictions among different research studies. *Harvard Educational Review*, 41(4):429–471, 1971.

[14] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[15] R. Oliveto, M. Gethers, D. Poshyvanyk, and A. De Lucia. On the equivalence of information retrieval methods for automated traceability link recovery. In *International Conference on Program Comprehension*, pages 68–71, 2010.