The Impact of Automated Support for Linking Equivalent Requirements Based on Similarity Measures

D. FALESSI, *Member, IEEE*, L.C. BRIAND, *Senior member, IEEE*, G. CANTONE

Abstract— When developing systems of systems, requirements tend to be redundant especially when running large numbers of projects, with many requirements per project, and diverse sources of requirements. It is therefore necessary to consolidate requirements by identifying the ones that are equivalent in order to avoid redundant work. The aim of this paper is to evaluate requirement similarity measurement to support analysts when linking equivalent requirements. The evaluation is conducted based on the requirements management process of an Italian company in the defense and aerospace domain. Our empirical investigation combines a controlled experiment with graduate students and an industrial case study. Results clearly show that one cannot expect any significant advantage in general. The level of support provided by similarity measures significantly depends on their level of credibility, that is the extent to which similarity measurement reliably indicates the equivalence of requirements. On average, given the credibility distribution observed in our industrial case study, showing similarity measurement to analysts is expected to: 1) improve by 20% the number of equivalence links identified per minute and 2) decrease by 40% the number of incorrect links. Finally, we investigate whether there is an effective way to combine human judgment and similarity measurement to effectively determine equivalence links. Based on machine learning, our approach yielded positive results both in terms of the correctness of the links and the speed at which they are established. Moreover, this hybrid solution is effective even when the credibility of similarity measurement is half the average we observed in our industrial case study. In conclusion, our results confirm and complement past empirical studies on the practical benefit, in terms of both quality and speed, of adopting requirement similarity measurement for linking equivalent requirements.

Index Terms— Experiment, Case study, Requirements tracing, Requirements consolidation, Similarity measure, Machine Learning.

1 Introduction

Software systems requirements engineering is "the process of discovering the purpose of software system, by identifying stakeholders and their needs, and documenting these in a form that supports analysis, communication, and subsequent implementation" [1]. When properly performed, requirements engineering is an opportunity to reduce costs and increase the quality of software systems. But when requirements are incomplete or incorrect, this can lead to the development of inadequate software products, significant delays, or project terminations.

One particularly important aspect is the identification of equivalent requirements to avoid assigning the same requirement to different developers and performing redundant tasks [2]. This is particularly common when requirements are numerous and many stakeholders are involved in defining them.

This paper focuses on assessing the use of similarity measurement between pairs of requirements in terms of correctly and efficiently linking equivalent requirements. Consistent with previous studies [3], we capture these two properties through variables referred to as the *Quality* and *Speed* of requirements linking, that will be further defined below. One important aspect is that we investigate how reliable similarity measurement is at capturing equivalence, and what this impacts quality and speed in realistic conditions. We will refer to the reliability of similarity measurement as *Credibility* and define its measurement at a later stage. In order to achieve both realism and sufficient control, our empirical investigation combines a controlled experiment with master students and a case study in collaboration with an Italian company in the defense and aerospace domain.

1.1 Research Questions

To fully investigate the impact of similarity measurement we addressed the following research questions:

R.Q. 1. Does showing the similarity measure affect the performance of linking equivalent requirements?

Performance is assessed both in terms of quality and speed. Because identifying equivalent requirements is a time-consuming task [4,5], the intent of similarity measurement is to decrease the effort required to link equivalent requirements (Speed). Furthermore, a false negative link would eventually cause doing the same job twice while a false positive link would eventually cause user dissatisfaction. This is what we will refer to as quality in this context, using several complementary measures to capture mistakes in linking requirements. Since results in [3] suggest that a similarity measure helped improve both quality and speed when applied to university project requirements, this research question aims both at replicating such results and investigating whether they generalize to industrial requirements. Investigating this question requires sufficient control to avoid confounding effects and will therefore be investigated through a controlled experiment.

- *D. Falessi and G. Cantone are with the Department of Informatics, Systems and Production engineering, at University of Rome – TorVergata, Viale del Politecnico 1, 00133 Rome, Italy. E-mail: falessi@ing.uniroma2.it, cantone@uniroma2.it.*
- *L. C. Briand is with Simula Research Laboratory, PO Box 134, 1235 Lysaker, Norway. E-mail: briand@simula.no.*

16 November 2009

R.Q. 2. What is the impact of credibility on quality and speed? Are improvements in quality or speed due to similarity measurement positively and significantly related to credibility?

In general, we expect that low credibility similarity measures would confuse analysts in linking equivalent requirements; this would eventually impact negatively both quality and speed. Therefore it is important to investigate, using the same controlled experiment as in R.Q.1, the possibility of an interaction effect between using similarity measurement and its credibility, on both quality and speed, for a realistic set of requirements pairs. Such an effect must be modeled and the minimal threshold of credibility value below which similarity measurement is counter-productive must be determined.

R.Q. 3. Given the level of credibility observed in an industrial case study, can we expect significant benefits from using similarity measurement?

Based on realistic credibility distributions in industrial requirements pairs, using our model of interaction effect developed in R.Q.2, we want to investigate if similarity measurement improves quality and speed in classifying industrial requirements.

R.Q. 4. Can we combine expert requirement linking and a similarity measure to build an optimal equivalence prediction model? Is it better than using either of the two sources of information alone?

Human classifications and similarity measures are two different sources of information regarding the equivalence of a given requirements pair. The strategy we investigate here applies machine learning to combine these two sources into a hybrid equivalence prediction model. The basic idea is that this prediction model, based on data analysis, would empirically determine how to optimally weigh the two sources of information. According to R.Q.2, the level of support provided by using similarity measurement may depend on the level of credibility of the measurement. Since in practice, we cannot predict the value of credibility of a given similarity measure for a given set of requirements pairs, it is important to investigate if we could use a hybrid equivalence prediction model to mitigate the effects of possible low credibility values.

1.2 Empirical approach

1.2.1 Linking process

The activity of linking artifacts (e.g., equivalent requirements), which is also sometimes referred to as *tracing*, can be decomposed in two sequential sub-activities: *searching* and *classifying*. The sub-activity of searching the artifacts is concerned with searching among a large number of artifacts (e.g., requirement pairs) for specific ones featuring a relevant property (e.g., equivalent pairs). Similarity measures can support this sub-activity by effectively ranking the artifacts according to their likelihood to be relevant; the top-ranked artifacts are referred to as *candidates*. The sub-activity of classifying the artifacts is concerned with establishing the type of relation between a pair of artifacts (e.g., establishing two requirements as equivalent or not) among candidates. The classification sub-activity is a human decision: the analyst, following the previously defined ranking of artifacts, classifies each pair until he believe that the remaining pairs are not worth considering.

1.2.2 Approaches for measuring the support provided by similarity measures

There are two main approaches to measure the level of support provided by a similarity measure to the linking process. They are referred to as *study of methods* and *study of humans*, respectively in [4].

- Study of methods: this post-mortem analysis relies on the principle that a pair of artifacts should be linked when the measured similarity is higher than a given threshold. Hence, the less the difference between actual and suggested links, the higher the support provided by the similarity measure. A given performance metric (e.g., Lag, Precision, Recall) measures a given aspect of support. Examples of such studies are reported in several articles [2,4,6,7,8,9,10,11]. The advantage of this approach is that, once a post-mortem analysis tool has been developed, and the actual links have been established, then the analysis process is entirely automated. This allows for the comparison of a large number of similarity measures on a given set of links. In addition, because no human analysts are involved, real requirements can be used without raising confidentiality issues. The main drawback of this approach is to entirely ignore the "classification" sub-activity. Not considering this human decision process essentially results in not accounting for human factors in the overall linking process.
- Study of humans: this consists in measuring the differences in human performance when tracing artifacts with and without the support of similarity measurement. Examples of this approach are reported in [3,12,13]. Its main strength is in observing the real phenomenon under study rather than simulating it. Moreover, the classification sub-activity is accounted for. In all past studies, such studies have been performed using students in a laboratory setting. The main problem with these studies lies in the artificial artifacts being used, rather than the subjects. It is expectedly difficult to use professionals as subjects, willing to link artifacts without the support of a similarity measure. An industrial case can therefore help study professionals, working on industrial requirements with the support of a given similarity measure. But is usually impossible to observe their performance without it and therefore obtain a baseline of comparison. The use of students does not represent a significant threat to external validity: past studies revealed that there is no significant difference between

masters students and professionals in performing activities related to requirements engineering [14,15]. However, in artificial settings, the artifacts being traced cannot usually be actual industrial ones due to non-disclosure constraints. Though there are open-source projects that provide (real) artifacts to be used, they usually do not include requirements. Additionally, a laboratory setting limits the number of artifacts to link and hence the statistical power of our subsequent analysis.

1.2.3 Our approach

Our aim is to measure the benefits of using similarity measurement when linking equivalent requirements. To do so, we combine the strengths of post-mortem analysis and human-based observation, as described above. Figure 1 sketches the activities (white background rectangles) and information (wavy-bottom rectangles) involved in the designed empirical process. The main hypothesis is that similarity measurement supports the classification sub-activity when it reliably captures equivalence (credibility). We first enacted an industrial case study (Section 3) where our tool PROUD (Section 3.2) has been used to detect equivalent requirements. This case study featured the highest level of realism as it involved real industrial requirements. In order to measure the support on the searching sub-activity we adopted the standard approach of measuring the Lag metric on the industrial requirements as proposed by Hayes et al. in [4]; this measured the ranking effectiveness. However, as discussed above, in order to properly assess the support of similarity measurement on the linking activity, we needed to take into consideration the classification sub-activity also. On this case study, an expert established the requirements links using PROUD only. Therefore, we cannot directly assess, on real requirements, the impact of using similarity measurement. We therefore decided to complement this case study with a controlled experiment aimed at measuring the impact of similarity measurement on the classification sub-activity. To produce adequate experimental objects, we modified the dataset coming from the industrial case study through sampling and modifications to meet the constraints of a controlled experiment and conform to the non-disclosure agreement with our industrial partner. Though the experiment, conducted in a laboratory setting, gave us the highest level of control, this somewhat artificial set of requirements may have biased the results and affect external validity. Therefore we decided to combine the results of a controlled experiment with the results of the case study to alleviate this problem. More specifically, we designed an experiment (Section 4) where equivalent requirements are pre-determined, defined a credibility measure for similarity measurement among requirements pairs, and used credibility as an interaction factor when assessing the impact of similarity measurement on the effectiveness (Quality) and efficiency (Speed) of classifying equivalent links. Once verified that credibility is a significant factor, using a realistic distribution of credibility from our industrial case study featuring actual requirements, we assessed the benefits that can be expected from using such similarity measures in practice. Our strategy was hence to combine a controlled experiment and an industrial case study to achieve the best balance between control and realism. The evaluation of Machine Learning to combine human judgment and similarity measures (see Section 1.1 R.Q. 4) was done following the experiment by using WEKA [16].

1.3 Structure of the paper

The rest of the paper is structured as follows; Section 2 reports on past works related to the present study. Section 3 describes the case study and Section 4 reports the experiment planning. Section 5 reports and discusses the results. Section 6 concludes the paper.

2 Related Works

Natural Language Processing (NLP) is a field of computer science concerned with the interactions between computers and human (natural) languages. Classical applications include the search for a set of documents that are similar to a given text; NLP supports humans by providing similarity measures related to the searched texts. In our case, NLP is used to measure the similarity among requirements; the similarity measure is used to rank the requirement pairs according to their likelihood of being redundant. The role of NLP in requirements engineering has been described by Ryan in [17] as a promising approach to support the requirements engineering process due to the increasing complexity of the systems to develop and hence of the requirements to manage. NLP has been applied in different areas of software engineering and in particular to link artifacts at different levels of abstraction including high-level to low-level requirements [10,12], requirements to design [18,19,20], requirements to source code [9,13,21], functional requirements to java code [7,8], requirements to defect reports [22]. Several studies have also suggested that NLP could support several requirements engineering activities including linking customer wishes to product requirements [23], consolidating requirements [2], reasoning about inconsistencies [24], supporting reuse [25], and domain analysis [26,27].

Stierna and Rowe [25] reported an industrial experience in applying information retrieval techniques where the requirements of existing systems are matched with the requirements of new systems for identifying reuse opportunities. Our approach is similar except for the fact that we aim to identify both proactive and reactive reuse opportunities; in other words, it is important for us to identify potential reuse from developed systems (reactive reuse) but also take advantage of the commonalities among new systems being concurrently developed (proactive reuse - product line engineering).

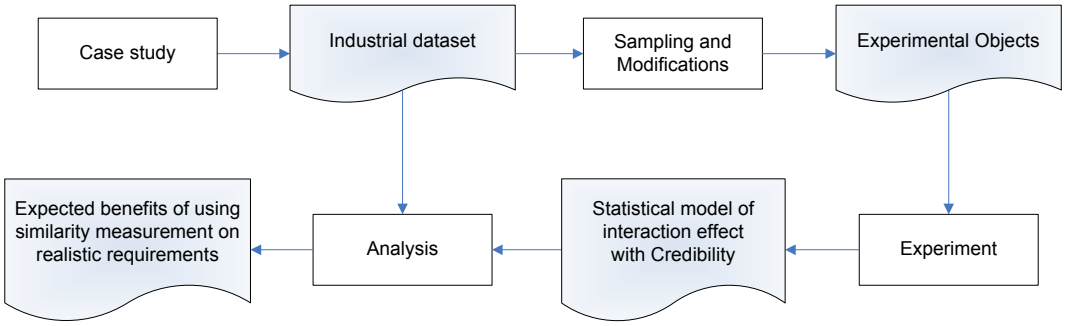

Figure 1. The activities (white background rectangles) and information (wavy-bottom rectangles) involved in the overall process to measure the impact of linking equivalent requirements based on similarity measurement.

Several studies regard the adoption of information retrieval techniques to develop feature models, which in turn support the visualization of commonalities and variability among systems. Alves et al. present in [28] an explorative study and propose a framework and a research agenda. Jhon et al. [29] report on an industrial case study where information retrieval techniques reduced expert load (i.e., Speed in our context) by more than 60%. We also aim at reducing expert effort in commonality identification through techniques that "are independent from space or time, techniques that can be applied anytime, anywhere, by almost every person without domain knowledge." [29]

Hayes and Dekhtyar [5] discussed the need for future empirical investigations on the effects of automated support in requirements linking. Hayes et al. [4] outlined two directions of research; one on methods and the other on their influence on human performance. They state: "The validation of utility requires the study of the users as much as it requires the study of methods." Hayes et al. also reported a pilot study, involving three subjects, supporting the hypothesis that "the accuracy of computer generated candidate traces affects the accuracy of traces produced by the analysts" [30]; i.e., the automated classification impacts on the human classifaction.

Natt Och Dag et al. report in [3] a controlled experiment where the adoption of the ReqSimile tool improved speed and quality in linking similar requirements. Our research focus is very similar to [3], though important differences in experimental design and measurement exist. Given their positive results, we decided to include in PROUD the same information retrieval technique developed into ReqSimile. We share with them the objective of providing effective automated support to link similar requirements when dealing with high numbers of requirements. Consistent with the authors' explicit call for replications [3], we build on their work in an attempt to confirm and extend their results. Commonalities between the present empirical study and that in [3] include:

- A controlled experiment in an academic setting.
- The use of two adapted versions of the same tool to maximize instrumentation validity.
- The dependent variables (Quality and Speed).

The main difference between the present study and that in [3] is the treatment being investigated: we study the impact of providing similarity measurement to subjects instead of providing both such measurement and a ranking of all requirement pairs. The motivation was to assess, in isolation, the effect of showing similarity measurement on subject's decisions regarding a requirement pair, and more specifically the classification sub-activity (see Section 2). The impact of ranking accuracy, measured for example with the Lag metric [4], should be studied independently. Further differences include:

- We investigate if the level of support provided by similarity measurement changes according to the requirements set on which it is used. In particular, our expectation is that similarity measurement is useful when it reliably captures similarity (high credibility), a property that may greatly vary across sets of requirements. Having modeled the impact of credibility on the effect of similarity measurement on requirement pairs' equivalence classification, we assess the likely benefits to be obtained by PROUD in realistic industrial conditions based on a case study.
- The controlled experiment adopted a cross-randomized design to avoid potential bias from subjects' characteristics.
- In the controlled experiment, we ensured that the numbers of equivalent and non-equivalent requirement pairs were the same; this to avoid the impact of unbalanced data on statistical analysis.
- Time measurement for requirement pairs' classifications was automated; this enabled the measurement of time for each classification individually.

3 Case Study

3.1 Context

3.1.1 Company profile

Finmeccanica is the main Italian industrial group operating globally in the aerospace, defense and security sectors, and is one of the world's leading groups in the fields of helicopters and defense electronics. It has revenues of 15 Billion Euros and invests 1.8 Billion Euros

(12% of turnover) a year in R&D activities. SELEX Sistemi Integrati (also known as SELEX SI) is the Finmeccanica Company focusing on the design of systems of systems; it aims to be the European leader in the definition and integration of sensors and systems for defense, coastal/maritime surveillance, and air traffic management. SELEX SI (SELENIA until 2005) has about fifty years of experience in system integration and a customer base in more than 150 countries, with plants in Italy, the UK (SELEX System Integrations Ltd), Germany (SELEX Sistemi Integrati GmbH), and the US (SELEX Sistemi Integrati Inc).

The context of this study is the development of systems of systems. A system of systems is not only large, distributed, adaptive and complex; its main peculiarity is that it is structured into components (i.e. systems) that can work independently from each other though their cooperation provides a functionality that is greater than the sum of their functionalities.

3.1.2 Causes and management of equivalent requirements

In the context of systems of systems, redundant/equivalent requirements have different implications depending on their context:

- **Mistake**: When the equivalent requirements belong to the same system, this is considered as a mistake.
- **Reuse opportunity**: When the equivalent requirements belong to different systems they represent a reuse opportunity. In the early specification stage, equivalent requirements represent a proactive opportunity to avoid redundant work. The term proactive stems from the software product line engineering community and denotes that though reuse may not already be an option, it can be achieved by a proactive approach [31,32,33,34,35,36].

Both cases above call for supporting the identification of requirements that are semantically equivalent. In the context of our case study (SELEX SI), requirements tend to be redundant due to the following reasons:

- Large number of requirements: each project includes around 500 requirements.
- Several ongoing, concurrent system development projects: five in our case study.
- Multiple sources: the requirements are elicited and written by several stakeholders that belong to different projects or different departments involved in the same project.

Experience from applying product line engineering shows that investing in reuse does not always pay off; the success factor consists in applying the right reuse strategy [37] with the right amount of investment [38] on the right asset (e.g., components, test cases, requirements) [39] without underestimating non-technical aspects (e.g., current organization). When SELEX SI started the development of six different products, it was clear that it would be wrong not to contemplate taking advantage of their commonalities. In order to reason on what and how to reuse, it is mandatory to have an understanding of the variability and commonalities of the systems to develop. This activity is called domain analysis. Among the several ways to enact domain analysis [40], linking equivalent requirements is particularly non-invasive for the actual organization, scalable, and it supports requirements consolidation [2,3,41]. Hence, in order to support the requirement engineering process, and the domain analysis phase in particular, we developed an open-source tool called PROUD, which incorporates linguistic engineering techniques for supporting commonalities identification.

3.2 A Proactive Reuse OpportUnity Discovery tool

The tool PROUD (Proactive Reuse OpportUnity Discovery) has been developed on Microsoft ActiveX COM® technology; we adopted Microsoft .NET Framework 2.0®, Microsoft® Visual Studio® 2008 as development environment and C# as programming language. PROUD works as a plug-in of a commercial CASE tool called Enterprise Architect® by Sparx Systems®.

Prior to identifying commonalities, PROUD supports requirements quality assurance. When processing the natural language in which the requirements are expressed, it checks their compliance to rules defined in the requirement specifications guidelines developed by SELEX SI. Requirements are then classified according to a pre-defined industrial taxonomy; examples of requirements categories are: user interface, simulation, and authentication.

When the requirements are found to comply with the specifications guidelines then they are analyzed for semantic equivalence. This activity is supported by PROUD in three stages: requirements selection, commonality identification, and commonality visualization.

- *Requirements selection*: in order to reduce human effort, it is important that PROUD allows the user to select the requirements that were not already analyzed for redundancy in the past. PROUD allows the user to define which requirements need to be analyzed with respect to which other according to several criteria, including creation time, correctness, classification, name of the package in which the requirement is stored, presence or absence of a particular word in the requirement text or its title. This is expected to support the user in commonality identification among a set of new projects as well as the consolidation activity between a set of consolidated requirements and new requirements.
- *Commonality identification*: given the requirements selected in the previous step, PROUD computes the similarity measure on all their possible pairs. The similarity measure is computed by applying a natural language processing technique as reported in [3]; all the requirement pairs are ranked according to the measure reported in the first column in Figure 2. At

this point, the user classifies the current requirement pair as equivalent or not.

- *Commonality visualization*: When all selected requirements have been classified for equivalence, PROUD automatically delete redundant requirements (i.e., equivalent requirements pairs belonging to the same project) and it produces artifact to visualize commonalities and variability among projects including requirements matrixes, pie charts, and UML feature models [42]. Such artifacts are then refined by analysts because, though important, the equivalence between requirements is just one of several aspects of projects commonalities. Eventually, analysts and project managers can effectively observe the commonalities among the projects developed and to develop; this supports reasoning about the right reuse strategy to apply and the related projects' planning [31,32,33,34,35,36].

3.3 Sampling strategy

The external validity of our case study [43] is particularly high due to the following reasons:

1. It contains real industrial requirements.
2. It is based on a recent project.
3. The number of requirements is large.

Table 1 reports the main characteristics of our case study at SELEX SI regarding five different projects on systems of systems. In order to make the study manageable, we sampled a subset of 983 requirement pairs to study out of a population of nearly three million possible pairs. Figure 3 reports the distributions, for this sample of requirement pairs, of similarity measurement and the number of words per requirement. Among these 983 requirements pairs, 183 were equivalent.

The 983 classified requirements pairs were selected using a mix random and ranking-based sampling strategy. First, we analyzed the top 391 pairs as ranked by PROUD in order to get as many equivalent pairs as possible. Because most of the requirements pairs are non-equivalent, a pure random sampling would have yielded a very small number of equivalent requirements, thus making any analysis difficult. We stopped analyzing the ranked pairs after finding 72 nonequivalent pairs in a row. We then continued by randomly sampling 592 pairs from the remaining pairs. Sampling is done without replacement and the number of sampled pairs is driven by how much time can realistically be dedicated by an expert to manually analyze requirements.

3.4 Preliminary results

In order to evaluate the level of support provided by PROUD for linking equivalent requirements we first computed the Lag metric. The Lag metric, as introduced by Hayes et al. in [4], is a measure of ranking effectiveness that represents the number of non equivalent requirement pairs that have a higher similarity than equivalent requirement pairs, on average among all equivalent requirement pairs. In other words, it represents the number of non-equivalent requirement pairs that need to be analyzed, on average, before analyzing an equivalent pair. Using PROUD's ranking, we obtain a Lag metric value of 0.77, as opposed to 4.36 when randomly selecting requirements pairs among the sample of 983. This means that, for each equivalent requirement pair, on average, PROUD saves the analyst the effort required to classify more than three non-equivalent requirements pairs. Note that this estimated gain is a lower bound based on the sampled 983 requirements pairs, which contain a much higher percentage of equivalent requirements (18%) than the complete set of pairs. Therefore, assuming that we found all the existing equivalent requirements (183 out of more than three million, or 0.00586% of all pairs) while sampling 983 requirements pairs, using PROUD's ranking, we obtain a Lag metric value of 0.77, as opposed to 16,976 when randomly selecting requirements pairs from the complete set. Though the number of detected equivalent requirements pairs is a lower bound for the entire set, the fact that the random sampling did not find any after rank 255 suggests this is close to the actual number. Formally then, based on the lower and upper bounds computed for the estimated gain in the analysis above, for each equivalent requirements pair, PROUD saves the analyst the effort to classify between 3 to 16,975 non-equivalent pairs. However, the latter number is much closer to reality.

Another issue is that the above evaluation doesn't take into account the support provided by PROUD for the classification sub-activity. As a matter of fact, we had positive feedback from our industry partners regarding the support provided by showing similarity measurement. Neither the Lag metric nor any other information retrieval metrics (i.e. Precision, Recall, etc.) would allow us to assess the gain related to human classification. We therefore complemented our industrial case study with a controlled experiment as described in the following section.

4 Experimental Planning

4.1 Goal and task

We want to assess the impact of using similarity measurement on the speed and quality at which humans can determine the equivalence of requirement pairs but we want to do so by accounting for such measurement's credibility, as discussed above. We will achieve this goal by means of a controlled experiment that will allow us to precisely model the interaction effect of credibility. Furthermore, we want to use these results to assess the likely benefits of using similarity measurement in an industrial context, by accounting for actual credibility distributions for actual requirement pairs in a large system.

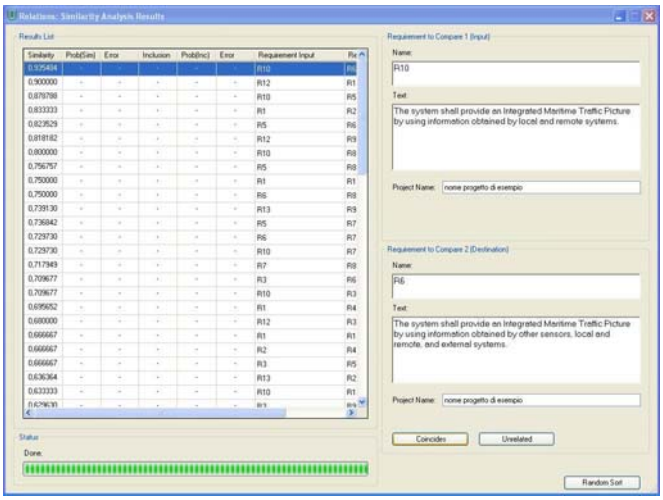

Figure 2. Commonalities identification: the requirement pairs are ranked according to similarity measurement, the analyst clicks on "coincides" when the current requirements pair is equivalent or "Unrelated" otherwise.

Table 1. Case study characteristics.

Application domain	Systems of Systems
Industry	Selex SI
Number of projects	5
Total number of requirements	2500
Total number of possible requirements pairs	3123750
Sample size (classified pairs)	983
Equivalent requirements pairs	183

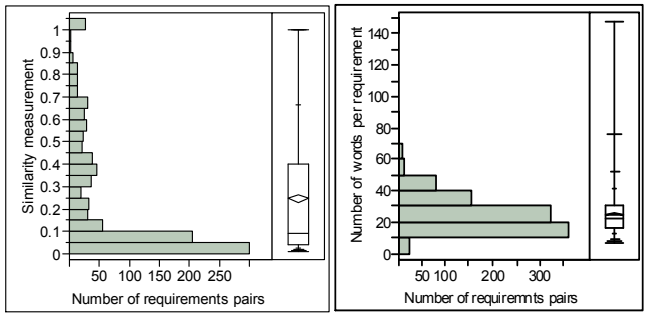

Figure 3. Case study data - distributions of similarity measurement and number of words per requirement.

4.2 Participants

Our controlled experiment was conducted at the University of Rome, Tor Vergata, and experiment subjects were 32 students taking a graduate course in empirical software engineering in the Department of Informatics, Systems and Production engineering (DISP). The course is given in the final year of a two year master-level education program in Informatics and Computer Engineering. All the students during previous bachelor and masters courses received extensive teaching on all the different phases of the software lifecycle, including requirements engineering. Most of the students had some industrial experience or worked as private consultants.

Subjects were not pressured to participate in the experiment. In fact we clearly explained that their course grade would not be related to their presence or performance during the experiment. This is an approach we have successfully adopted over several years in other experiments.

4.3 Variables

4.3.1 Treatments

The independent variable investigated is the linkage method used by subjects to deem if a requirement pair is equivalent or not. The associated treatments being compared are the following three linkage methods:

- **Pure**: Simply read the text of the two requirements of the current pair. In other words, no similarity measure is used (Figure 5 (a)).
- **Supported**: Read the text of the two requirements of the current pair and consider their similarity measurement as additional information (Figure 5 (b)).
- **Combined**: The same procedure as in Supported above is used by the human subject to establish equivalence. Then, consistent with the rationale described in Section 1.1 R.Q. 4, this human opinion is combined, using machine learning, with the similarity measure, to build a predictive model of requirement pair equivalence (Figure 5 (c)).

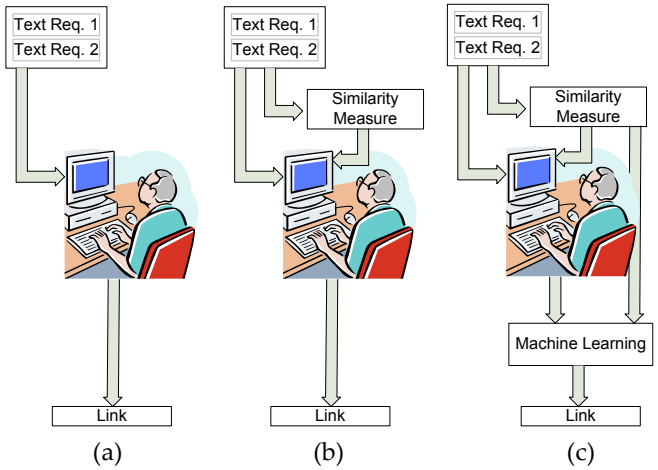

Figure 4. The different linkage methods: Pure (a), Supported (b), Combined (c).

4.3.2 Dependent variables

We are interested in two dependent variables:

- **Quality**: This captures, using different measures, the correctness of linking requirements pairs. A link is considered correct if its two requirements are indeed equivalent. A false negative would eventually lead to redundant work whereas a false positive would eventually result into client dissatisfaction. To enable the measurement of the quality of links, the correct links of all the requirement pairs were assigned beforehand by an expert, who has extensive experience in teaching requirements engineering and dealing with industrial requirements in the chosen application domain. In order to mitigate validity threats, requirements equivalence was established prior to the experiment execution. Moreover, as described in Section 5.1, we compared the equivalence links established by the expert to the decision majority

by the subjects on each of the 144 requirement pairs. Results show to be perfectly consistent between the two. We adopted the following measures of quality, which are standard when assessing classifications based on confusion matrices [44]:

- *Accuracy*: the fraction of the classifications that are correct; i.e., (true positive + true negative) / (true positive + true negative + false positive + false negative).
- *Precision*: the fraction of retrieved documents that are relevant; i.e., (true positive) / (false positive + true positive).
- *Recall*: the fraction of relevant documents that are retrieved; i.e., (true positive) / (false negative + true positive).
- *True negative rate*: the fraction of non-retrieved documents that are irrelevant; i.e., (true negative) / (false positive + true negative).
- *False positive rate*: is the fraction of retrieved documents that are irrelevant; i.e., (false positive) / (false positive + true negative).
- *False negative rate*: is the fraction of non-retrieved documents that are relevant; i.e., (false negative) / (false negative + true negative).

- **Speed**: Because requirements linkage is a time-consuming task [3,27], similarity measurement is also aimed at decreasing the effort required to classify two requirements as equivalent or not. We measured speed as the number of links established per minute. Since PROUD provides the initial time and the final time for each link with precision in seconds, we compute speed as (60/(*final time* – *initial time*)).

4.4 Hypotheses

The following experiment hypotheses are derived from the above mentioned research questions. We will refer to the three treatments associated with our independent variable, that is the method followed for linking requirements: Pure, Supported, and Combined.

- *R.Q. 1 Does showing the similarity measure affect the performance of linking equivalent requirements?*
  - H^1_0 *Supported* results in the same number of linked requirements per minute as Pure.
  - H^2_0 *Supported* results in the same proportion of correct links as *Pure*.
  - H^3_0 *Supported* results in the same precision as *Pure*.
  - H^4_0 *Supported* results in the same recall as *Pure*.
  - H^5_0 *Supported* results in the same true negative rate as *Pure*.
  - H^6_0 *Supported* results in the same false positive rate as *Pure*.
  - H^7_0 *Supported* results in the same false negative rate as *Pure*.
- *R.Q. 2 What is the impact of credibility on quality and speed? Does using the similarity measure improve quality or speed of linking requirements? Does it only do so when its credibility is high? Does it decrease quality and speed when credibility is low?*
  - H^8_0 The *credibility* of the similarity measure does not significantly interact with the *Supported* method in terms of *number of linked requirements* per minute.
  - H^9_0 The *credibility* of the similarity measure does not significantly interact with the *Supported* method in terms of *correctness* of the requirement links.
- *R.Q. 3 Given the level of credibility observed in an industrial case study, can we expect significant benefits from using similarity measurement?*
  - H^{10}_0 Given a realistic level of *credibility*, *Supported* results in the same number of *linked requirements per minute* as does *Pure*.
  - H^{11}_0 Given a realistic level of credibility, *Supported* results in the same proportion of correct links as *Pure*.
- *R.Q. 4 Can we combine expert requirement linking and a similarity measure to build an optimal equivalence prediction model? Is it better than using either of the two sources of information alone?*
  - H^{12}_0 *Combined* results in the same proportion of correct linkage decisions as does *Supported* and *Pure*.
  - H^{13}_0 *Combined* is as *precise* as *Supported* and *Pure*.
  - H^{14}_0 *Combined* results in the same proportion of correctly identified equivalent links (*recall*) as does *Supported* and *Pure*.
  - H^{15}_0 *Combined* results in the same proportion of correct identified nonequivalent links (*true negative rate*) as does *Supported* and *Pure*.
  - H^{16}_0 *Combined* results in the same proportion of incorrect equivalent links (*false positive rate*) as does *Supported* and *Pure*.
  - H^{17}_0 *Combined* results in the same proportion of incorrect nonequivalent links (*false negative rate*) as does *Supported* and *Pure*.
  - H^{18}_0 Given a realistic level of *Credibility*, *Combined* results in the same proportion of correct links as does *Supported* and *Pure*.

Because the combination of human decision and similarity measurement requires a negligible computational time, we assume (and don't test) that *Combined* results in the same *number of linked requirements per minute* as does *Supported*.

4.5 Objects

In order to adopt a cross randomized experimental design, we use two sets of requirements. In our experiment, each set was composed of 72 pairs; the number of pairs was decided based on the estimated number of requirements pairs the subjects could handle within the experiment time.

The requirements to link were randomly selected from the case study and then sanitized to meet the non-disclosure agreement with our industrial partner. This

gave us the possibility to use a set of realistic requirements. Moreover, in order to facilitate statistical analysis, we selected requirements pairs according to the following strategy:

- Proportion of equivalent requirements: Requirements pairs were selected in order to achieve an equal number of equivalent and non-equivalent pairs; having a balanced dataset greatly facilitates statistical analysis.
- Credibility distribution: In order to assess how reliable is our similarity measure, we defined *Credibility* as *1-|similarity measure – human link|* where human link is the subjective judgment of an expert regarding the equivalence of a requirements pair: *1* when the requirement pair is classified as equivalent and *0* otherwise. The more correlated similarity measurements and expert judgments are, the higher *Credibility* is, within a [0 , 1] range. Given this definition, we selected requirements pairs having a wide *Credibility* distribution (see Figure 5). This helped better analyze the statistical interaction of *Credibility* with the performances of linkage methods.

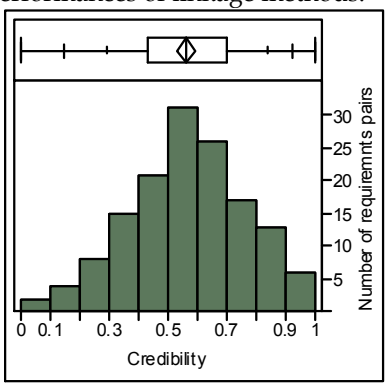

Figure 5. Distribution of *Credibility* among the requirements pairs in the experiment.

4.6 Experiment design

We adopted a cross randomized design consisting of two rounds. In each round, subjects inspected 72 pairs of requirements, in a random order, and decided about their semantic equivalence. A nominal duration for each round was suggested but not enforced.

We developed two sets of 72 requirements pairs; one set for each round. Subjects were randomly assigned to one of two groups, each working on a different set of requirements pairs and with a different version of PROUD (with and without similarity) in each round; both rounds and both groups involve the same number of subjects. This cross-randomized design helped avoid the following confounding effects on our dependent variables:

- Subjects skills: All subjects applied both treatments in a random order
- Learning effects: Both treatments were applied in equal proportion of subjects across rounds.

Experimental subjects attended a training of four hours a couple of weeks before the experiment execution in order to ensure they were competent to perform the tasks.

4.7 Experimental materials

As mentioned above, we used two versions of PROUD developed specifically for the experiment. They provided only the minimal functionalities to run the experiment and the two versions were exactly the same except for the feature showing similarity measures (see Figure 4, that is the treatment we are evaluating).

For each requirements pair linkage, PROUD records in a log-file the following information:

- Initial time: Actual starting time in seconds.
- Identifiers of each of the two requirements.
- Number of words for each of the two requirements.
- Text of the two requirements.
- Similarity measurement.
- Type of link established by human (equivalent or nonequivalent).
- Final time: Actual ending time in seconds.

Therefore the required time for a human to link a requirements pair is computed as final time less initial time.

4.8 Procedure

The experiment was run in one two-hour lab session in the fall 2008. The first 30 minutes were dedicated to present an overview of the functionality of the two versions of the tool. The differences between versions were communicated without favoring one over the other.

In order to avoid bias, no hypotheses were revealed and it was made very clear that we didn't know what results to expect from the experiment. Of course, no information was given to subjects regarding the proportion of equivalent requirements pairs. During the presentation we presented to the entire lab the procedure to follow, the tasks, and the type of material that they were going to receive and use during the experiment.

No subject was allowed to discuss his or her work with another student during the experiment. The material was both in paper and electronic forms. Paper material consisted of the general experiment instructions and rules; this was given to subjects at the beginning of the lab session. The electronic material included a set of 72 requirements pairs and a version of the tool, to use by a given subject, in a given experiment round, according to randomized assignments. The lab provided an internet connection; the material was delivered to specific subjects by email.

The following procedure was instructed to students:

1 Upload the email with experimental material and instructions from the experimenters,
2 Extract and install the version of the tool that is attached to the email.
3 While installing, overwrite any existing file.

4 Extract and open the file attached in the email containing the specific set of requirement pairs.
5 Randomize the requirement pairs order by clicking on a specific button (see the button at bottom-right side of Figure 2). Recall that such randomization helps avoid confounding effects between showing similarity measurement and ranking, the latter being measured by the Lag metric.
6 Link each requirement pair, in the given order, as equivalent or not.
7 When done or after around 45 minutes, send an email to the experimenters containing the log-file created by PROUD.
8 Wait for the acknowledgment of receipt from the experimenters.
9 If no acknowledgment is received within a few minutes, contact the experimenters. Otherwise, read the incoming email and start the second round by enacting again the steps 2 to 8 above.

The use of email prevented deviations from the experiment design. It helped avoid confusion in the delivery of the material to the right subjects and in the source of the log files received.

5 Results analysis and Discussion

5.1 Validation and Pre-processing of the data

The log-files recorded by PROUD went through pre-processing before analysis. Two Java applications were developed for that purpose, both available at www.eseg.uniroma2.it.

Past studies revealed that experts commit both errors of omission (false negative) and commission (false positive) in linking requirements when attempting to establish true equivalence links among requirements pairs [2,5]. In order to detect such errors, we compared the expert decisions with those of the participants. A first application (A1) takes as input all the subject log-files and computes in an output file the most common decision among participants (equivalent or nonequivalent) for each requirements pair. This was then compared with expert decisions and consistency was verified on each of the 144 requirements pairs. Expert judgment turned out to coincide with that of the majority of the subjects and increased our trust in the former.

Application A2 takes in input all the log-files and expert judgment files and provides as output a single file classifying all subject decisions as true positive, true negative, false positive, and false negative. The raw data was then imported into analysis tools: JMP® for statistical analysis and WEKA for machine learning [16].

In order to verify the subjects' adherence to experiment rules and procedures, and the absence of errors in treatment assignments or usages, both A1 and A2 included a validation step where sanity checks were performed on the subjects' log-files:

- No more than one link per subject per requirement pair.
- Pairs were linked in a random order.
- Subjects adopted the right version of PROUD, on the right set of requirements pairs, during the right experiment round.

5.2 R.Q. 1 Does showing the similarity measure affect the performance of linking equivalent requirements?

5.2.1 R.Q. 1 a) Quality

Table 2 reports the experiment results in terms of standard confusion matrix classification criteria for the Pure (2nd column) and Supported (3rd column) linkage methods. Results show that Pure actually outperforms Supported, though often by small margins.

Table 3 reports the p-value obtained when comparing the two linkage methods, by means of a two-tailed, Z-score test [45] for proportions, in terms of true/false negative/positive rates and precision. Results indicate that all differences in proportions, though often very small, are statistically significant ($\alpha = 0.05$).

Table 2 Quality of Pure versus Supported linkage methods.

	Pure	Supported
True positive	986	929
True negative	919	874
False positive	86	92
False negative	60	83
Accuracy	0.9288	0.9115
Recall or true positive rate	0.9426	0.9180
False positive rate	0.0855	0.0952
True negative rate	0.9144	0.9047
False negative rate	0.0573	0.0820
Precision	0.9197	0.9098

Table 3 Statistical tests for Pure versus Supported linkage methods.

Measure	Hypothesys	P-value
Accuracy	H^2_0	0.0000
Recall or true positive rate	H^3_0	0.0000
False positive rate	H^4_0	0.0000
True negative rate	H^5_0	0.0000
False negative rate	H^6_0	0.0000
Precision	H^7_0	0.0000

5.2.2 R.Q. 1 b) Speed

Table 4 reports the average number of links established per minute (2nd row) for Pure and Supported. These distributions depart significantly from normality and we therefore used the non-parametric, Mann–Whitney–Wilcoxon test to assess the significance of differences in speed. Table 4 also reports the p-value obtained when comparing the Pure and Supported, which shows there is no statistically significant difference. In conclusion, showing similarity measure did not significantly affect the speed of human analysts in classifying equivalent requirements.

Table 4 Comparing the speed of Supported and Pure.

	Pure	Supported
Classifications per minute	6.54	6.32
H^1_0 (P-value)	0.275	

5.2.3 R.Q. 1 c) Discussion

Against all expectations, similarity measures appeared to be counterproductive for supporting the classification of equivalent requirements. This contrasts with past studies [3] where similarity measurement yielded improvements both in terms of quality and speed. Possible reasons for this difference are considered below:

1) Differences in subjects: It would be difficult to understand why master students (current experiment) would perform significantly worse than bachelor students (experiment in [3]) when provided with similarity measurement.
2) Difference in similarity measurement: Since PROUD and reqSimile [3] use the same natural language technique to measure similarity, this is not possible.
3) Differences in proportions: Having a balanced proportion of equivalent and non-equivalent requirements should not impact the results as both treatments should be equally affected.
4) Differences in requirements pairs: Similarity measures are based on heuristics and may be more or less reliable at predicting equivalence (credibility), depending on the specific requirements considered. This seems to be the most likely reason and is therefore investigated next.

5.3 R.Q. 2 What is the impact of credibility on quality and speed? Are improvements in quality or speed due to similarity measurement positively and significantly related to credibility?

In general, we expect that unreliable similarity measures would confuse analysts in linking requirement pairs; this might eventually negatively impact both quality and speed. Hence, if our experiment is based on a set of requirements pairs for which similarity measures are poor indicators of similarity, Pure decisions might fare better than Supported ones. Given the similarity measurement provided by PROUD, we need to analyze whether for the requirements pairs with higher credibility, Supported outperforms Pure in terms of accuracy or speed. To do so, we investigate interaction effects between similarity and credibility on accuracy and speed.

5.3.1 R.Q. 2 a) Quality

We statistically tested the existence of an interaction effect between *Method* and *Credibility* on correctness by applying logistic regression. This regression technique is used as the dependent variable, *Correctness*, is binary (i.e., are two linked requirements equivalent or not?) and we therefore tackle a classification problem. The regression model in Table 5 features both a *Method* variable and an interaction term multiplying *Method* and *Credibility*. This is meant to capture a possible interaction effect. As we can see, the interaction term is statistically significant, though *Method* is not. This suggests the presence of a strong interaction effect.

To better understand what that interaction effect implies, it is better to look at an interaction plot [46], as depicted in Figure 6, which plots actual values. This helps us visualize how the probability of link correctness varies according to the type of *Method* and *Credibility*. In Figure 6 we can see that the Supported correctness curve crosses the Pure line around a *Credibility* value of 0.6, above which the similarity measure helps improves correctness (i.e., the probability for the link to be correct).

5.3.2 R.Q. 2 b) Speed

Because speed is defined on a ratio scale, we used least squares regression to study its relationship with linkage method and credibility. From Table 6 we can see that once again the interaction effect between *Credibility* and *Method* is statistically significant whereas the main effect term for *Method* is not.

As for correctness, the interaction plot in Figure 7 depicts the interaction effect between linkage method and the level of credibility on speed. Once again, Supported is beneficial above a *Credibility* threshold of roughly 0.6.

Table 5 Logistic regression for correctness

Nominal Logistic Fit for Correctness				
Parameter Estimates				
Term	**Estimate**	**Std Error**	**ChiSquare**	**Prob>ChiSq**
Intercept	-2.56	0.0859	894.72	<.0001
Method[Supported-Pure]	0.1030	0.1235	0.700	0.40
(Credibility-0,5632)* Method[Supported-Pure]	-3.15	0.4806	43.19	<.0001
H^8_0			**P-value**	<.0001

Table 6 Standard least squares regression for speed.

Standard Least Squares Regression For Speed				
Parameter Estimates				
Term	**Estimate**	**Std Error**	**tRatio**	**Prob>|t|**
Intercept	6.546	0.168	39.040	<.0001
Method[Supported-Pure]	-0.209	0.239	-0.87	0.383
(Credibility-0.56338)* Method[Supported-Pure]	6.396	0.956	6.690	<.0001
H^9_0			**P-value**	<.0001

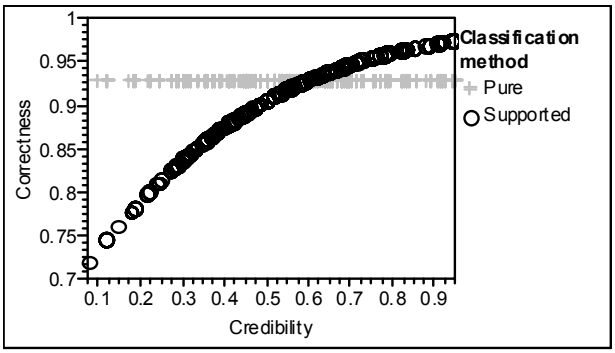

Figure 6. Interaction plots for correctness between *Credibility* and *Method* (Pure or Supported).

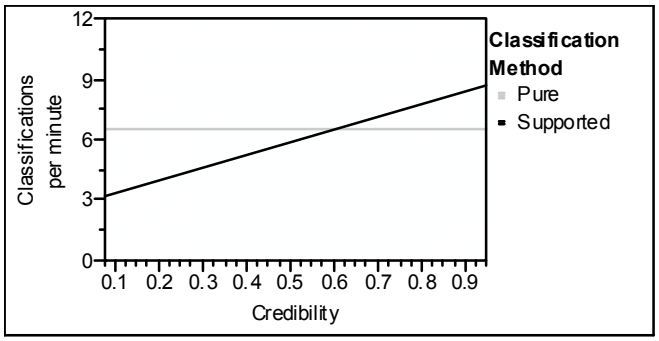

Figure 7. Interaction plots for speed between *Credibility* and *Speed* (Pure or Supported).

5.3.3 R.Q. 2 c) Discussion

There is an interaction effect between linkage method and credibility for both correctness and speed. In addition, it is interesting to note that the linkage method has no main effect neither for correctness nor speed. In other words, results suggest that for supporting the analysts in classifying equivalent requirement pairs, showing similarity measurement is beneficial, on a given set of requirements pairs, only when its credibility is above a given threshold. It would therefore be important to investigate how to write requirements in order to maximize the credibility of a used similarity measure, if at all applicable. Perhaps requirements templates can be devised to make the requirements more amenable to similarity measurement.

Moreover, given the facts that i) the average *Credibility* of the requirements pairs adopted in the experiment is around 0.55 (see Figure 5), and ii) showing similarity measurement support quality and speed only when *Credibility* is higher than about 0.6, it becomes clear why in our experiment, when not accounting for interaction effects, similarity did not seem to help improve either correctness or speed (see Table 2).

Given that in practice we do not know the distribution of credibility values for a set of requirements, we cannot assess beforehand the expected benefits of applying similarity measures. We can also draw additional conclusions:

- The impact of credibility probably explains the disagreement between the results of Table 2 and the previous study reported in [3]. Credibility needs to be measured and reported to be able to compare studies on the use of similarity measures for requirements linking.
- It is important to account for the following considerations:
  - One must adopt a natural language processing technique leading to a reliable similarity measure in a given context (e.g., requirements template),
  - One must adapt requirements templates to the used similarity measure, to ensure maximum credibility (e.g., use of a domain model to make vocabulary consistent)
  - If none of the above is possible, one must use the similarity measure only when appropriate (if it can be determined) and in a decision process using different sources for classifying requirement pairs. This is addressed by R.Q.4 in Section 5.5.5 where a machine learning model is used to combine expert linking with similarity measurement to provide an optimal classification.
- Credibility is a necessary factor to measure and account for when assessing the effect of similarity measures. In fact, standard confusion matrix criteria (e.g., precision, recall) based on rankings are not appropriate measures to account for the classification sub-activity (see Section 1.2.1).
- One must investigate and characterize, in the context of application, the level of credibility that can be expected in real requirements so as to assess the expected benefits from similarity measurement in terms of quality and speed. Thus an informed decision can be made on whether or not to use such technology, using the procedure we propose in the next section.

5.4 R.Q. 3 Given the level of credibility observed in an industrial case study, can we expect significant benefits from using similarity measurement?

In order to answer this question we must analyze credibility in a representative set of real requirements. We ran an industrial case study where a domain expert analyzed about a thousand of requirement pairs; 183 requirements turned out to be equivalent. Our goal was to ensure sufficient numbers of analyzed and equivalent requirements pairs to enable statistical analysis.

The requirements links on this case study were established by the expert following the Supported method. Therefore, we cannot use them to compare the Pure and Supported methods on real requirements. But since in R.Q. 2, based on our controlled experiment we observed a significant interaction effect between credibility and the linkage method, our approach is to assess the expected benefits of using similarity measurement by interpreting the regression model obtained via the experiment in light of the observed *Credibility* distribution in the industrial requirements set (Figure 5). The possible differences in human performance when classifying requirements in the experiment and the industrial case study should be low because the experiment requirements were drawn and adapted from the industrial requirements. Moreover, such possible differences should affect both treatments equally.

In general, since the average *Credibility* of the requirements pairs was around 0.81, and since our experiment has shown that similarity measurement improves both quality and speed when *Credibility* is higher than 0.6, we expect benefits in the industrial case study. More precisely, according to Figure 5, around 75% of requirement pairs have a *Credibility* value higher than 0.6 and should therefore benefit from the use of similarity measurement.

5.4.1 R.Q. 3 a) Quality

The ratio of correct links in the experiment dataset for Pure links is 0.929 (see Figure 6). Hence, the expected gain from similarity measurement can be assessed in the case study requirements by subtracting 0.928 from the predicted correctness using the model in Table 5. Figure 9 describes the distribution of this gain in correctness for the industrial requirements pairs. According to Figure 9, the impact is positive in more than 90% of the cases.

The average gain is 0.028. Because the percentage of incorrect classifications with Pure is 7.1% (i.e., 1 - 0.929) then the expected percentage of incorrect classifications adopting Supported is 4.3% (i.e., 1 – 0.929 – 0.028). Therefore, in an industrial setting, showing similarity measurement reduced, on average, the number of incorrect links by 40% (i.e., 1 - 4.3/7.1).

Table 7 reports the p-value resulting from applying a one-tailed Z-test score comparing the proportions of the expected correctness on real requirements pairs of the two linkage methods. Results clearly show that the difference is statistically significant.

5.4.2 R.Q. 3 b) Speed

The average number of links classified per minute in the experiment dataset for Pure links is 6.54. According to the least squares regression model (Table 6) on the experiment dataset, the number of classifications per minute (when using Supported) is: 2.73 + 6.39**Credibility*. Hence, the average gain in speed expected from Supported is 1.36 links per minute, defined as: -3.81+6.39*0.81, by subtracting 6.54 from 2.73 + 6.39**Credibility*. Hence, in an industrial setting, showing similarity measurement increased speed by 20%.

Figure 10 describes the distribution of the gain among the industrial dataset of requirements. According to Figure 10, showing similarity measurement has a positive impact on speed (a positive gain) in more than 95% of the cases. The maximum gain is around 2.6 links per minutes while, though expected to be very rare in practice, showing similarity measurement can decrease speed up to three links per minute.

In order to test the difference between the expected speed of Pure and Supported on industrial requirements we adopted a one-tailed Wilcoxon Signed Rank test. The expected speed of Supported was represented by a distribution obtained by applying the regression model obtained via the experiment on the observed Credibility distribution in the industrial requirements set (Table 6). Consistent with our assumption, we are using the Pure speed distribution from the experiment as this is not available in the industrial case study. Table 8 reports the p-values, that clearly show that the difference in speed is statistically significant.

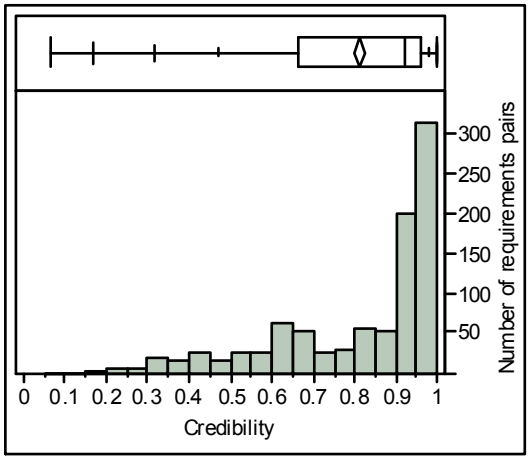

Figure 8. Distribution of *Credibility* among requirements pairs in the industrial case study.

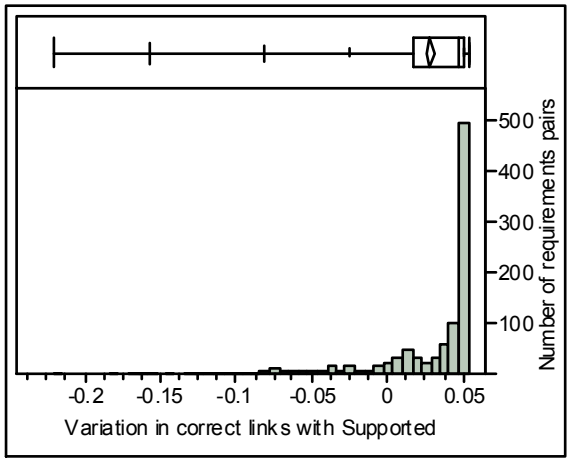

Figure 9. Expected variation in quality when using Supported, when compared to Pure, on real requirements pairs.

Table 7 Testing differences among linkage methods of expected correctness on real requirements pairs.

	Pure	Supported
Expected Correctness	0.929	0.956
H^{10}_0 (P-value)	0.0000	

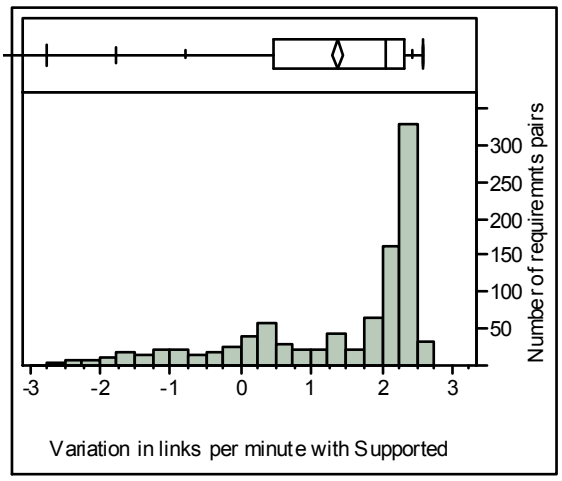

Figure 10. Expected variation in speed when using Supported, when compared to Pure, on real requirements pairs.

Table 8 Testing differences among linkage methods of expected speed on real requirements pairs.

	Pure		Supported	
	Mean	StdDev	Mean	StdDev
Expected Speed	6.55	7.75	7.91	1.30
H^{11}_0 (P-value)	0.00000			

5.4.3 R.Q. 3 c) Discussion

Given the distributions of *Credibility* in an industrial case study, using the experiment data for modeling the relationship between link correctness and credibility and linkage method, we investigated the expected improvement in both quality and speed resulting from using similarity measurement. The above results confirm what was reported in [3] and thus suggest that similarity measures should be used to help people link equivalent requirements.

5.5 R.Q. 4 Can we combine expert requirement linking and a similarity measure to build an optimal equivalence prediction model? Is it better than using either of the two sources of information alone?

Results from our experiment (R.Q. 3) and past studies [3] support the hypothesis that using similarity measure can increase both the quality and speed of humans when linking equivalent requirements. However, according to the results obtained for R.Q.2, the use of a similarity measure can also be counterproductive for requirements pairs whose similarity measurement has low credibility. Moreover, we note that the level of credibility of a given similarity measure on a given set of requirements cannot be estimated a priori. In fact, credibility may change according to: 1) the specifics of the natural language processing technique used to measure requirements' similarity, 2) the intrinsic characteristics of the natural language in which requirements are expressed in a given context. Therefore, we need to investigate whether similarity measures can support humans in linking equivalent requirements in a more robust way. The strategy we investigate here applies machine learning to combine human classification and similarity measurement into a hybrid equivalence prediction model. The basic idea is that this prediction model, based on data analysis, would empirically determine how to weigh the two sources of information when they disagree. To investigate the effectiveness of this idea we use the experiment data subset for Supported links and report the results below.

5.5.1 R.Q. 4 a) Quality

In order to investigate the quality of a hybrid equivalence prediction model using human links and similarity measurement in input, we tried several machine learning techniques including logistic regression, Bayes network, and a Decision Table/Naïve Bayes hybrid classifier (DTNB) [47]. In the present paper, we will report the results of DTNB since this is the machine learning technique that fared the best. To evaluate this hybrid equivalence prediction model we applied a 10-fold cross-validation approach [16]. The adoption of cross-validation is a standard approach for assessing prediction models in order to obtain realistic and generalizable results.

Table 9 provides various confusion matrix metrics for assessing the quality of the three linkage methods according to the experiment data. The first two columns in Table 9 were already presented in Table 2 and are reported here to facilitate comparisons among linkage methods. We can clearly see improvements with respect to all evaluation criteria for the Combined method when compared to Pure and Supported. One question is whether these differences are significant.

Table 10 reports the p-values resulting from applying a one-tailed Z-test score comparing proportions (see Section 4.3.2) related to various accuracy measures of Combined to the two other linkage methods. Results clearly show that all difference are statistically significant for all quality criteria.

The Combined linkage method provides better correctness for establishing requirements pair equivalence. The questions are then to determine its performance on industrial requirements and how dependent is this result on credibility values.

Table 9 Accuracy of linkage methods based on experimental results.

	Pure	Supported	Combined
Accuracy	0.9288	0.9115	0.9348
Recall or true positive rate	0.9426	0.9180	0.9428
False positive rate	0.0856	0.0952	0.0734
True negative rate	0.9144	0.9048	0.9266
False negative rate	0.0574	0.0820	0.0572
Precision	0.9198	0.9099	0.9289

Table 10 Testing differences among linkage methods on experimental results.

	Hypothesys	P-value	
		Pure vs Combined	Supported vs Combined
Accuracy	H^{12}_0	0.0000	0.0000
Recall or true positive rate	H^{13}_0	0.0000	0.0000
False positive rate	H^{14}_0	0.0000	0.0000
True negative rate	H^{15}_0	0.0000	0.0000
False negative rate	H^{16}_0	0.0000	0.0000
Precision	H^{17}_0	0.0000	0.0000

5.5.2 R.Q. 4 b) Expected benefits on the quality of industrial requirements

The average correctness in the experiment dataset concerning only Pure links is 0.929 (see Figure 6). Hence, the expected gain in correctness by combining similarity and human classification, can be computed on real requirements by subtracting 0.929 from the correctness predicted by the Combined model in the previous section. Figure 11 describes the gain distribution for Combined among the industrial requirements set. According to Figure 11, combining similarity and human classification shows a positive gain in quality in more than 90% of the cases. The average gain is 0.034; this means that the

percentage of incorrect classifications decreases up to 3.7% (when using Combined, the expected ratio of incorrect classifications is: 1 – 0.929 – 0.034). Hence, in an industrial setting, combining similarity measurement reduced, on average, the number of incorrect links by 48% with respect to Pure (i.e., 1 – 3.7/7.1). Table 11 reports the p-value resulting from applying a one-tailed Z-test score comparing proportions related to the expected correctness of Combined to the two other linkage methods. Results show that the difference between Combined and Pure is statistically significant while the difference between Combined and Supported is not.

5.5.3 R.Q. 4 c) Robustness

Though both Combined and Supported are affected by requirements' credibility, our expectation is that Combined is more robust than Supported to low credibility values. This is justified by the fact that it combines two sources of information, human links and similarity measurement. Figure 12 shows the interaction plot in terms of logistic regression, of the three linkage methods. The Pure and Supported curves correspond to the model in Figure 6 and the Combined curve corresponds to a logistic regression model predicting correctness based on the DTNB model output and credibility, trying to model the interaction between the two. Figure 12 shows that the Combined curve crosses the Pure line earlier on the *Credibility* range than the Supported curve. Combined therefore requires a lower level of credibility to provide benefit when compared to Supported, thus making similarity measurement applicable to a wider set of contexts. In other words, Combined is more robust to low credibility than Supported.

5.5.4 R.Q. 4 d) Discussion

Combined improves both the quality of links (Accuracy, Recall, False positive rate, True negative rate, False negative rate, Precision) when compared to both Pure and Supported linkage methods. Hence, though the level of credibility of the experiment dataset is significantly lower than for the industrial case study, when using Combined, similarity measurement is nevertheless beneficial (Table 9). In other words we went from a situation where similarity measurement was at best not beneficial (Supported) to a situation where it is significantly valuable in terms of correctness improvement, when used in combination with human decisions (Combined).It is interesting to observe that Combined outperformed Supported both in the experiment and in the industrial datasets. This result is confirmed by analyzing Figure 12; Combined outperforms Supported on the entire credibility range. However, the higher the credibility, the lower the differences between the performances of Combined and Supported. In fact, according to Table 11, their difference is statistically significant on the experiment dataset while it is not significant on the industry dataset (where the credibility is higher).

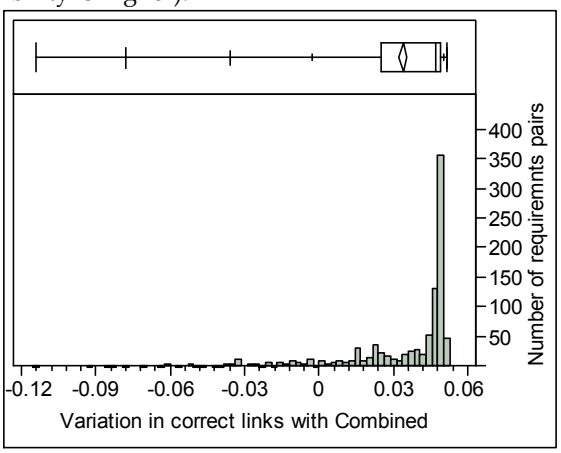

Figure 11. Expected variation in quality from Combined when compared to Pure on real requirements pairs.

Table 11. Testing differences among linkage methods of expected correctness on real requirements pairs.

	Pure Vs Combined	Supported vs Combined
Expected correctness H^{18}_0 (P-value)	0.0000	0.1873

Results from Table 9 are consistent with Figure 12: Combined is better regarding correctness than Pure when *Credibility* is higher than 0.47, that is significantly lower than the average (0.55) observed in the experiment. In other words, when using the hybrid prediction model described above, the minimum *Credibility* from which the use of similarity measurement provides benefits decreases from 0.6 to 0.47, thus making similarity measurement more robust to low values of credibility.

In conclusion, combining human classification with similarity measure is expected to provide higher performances in respect to single sources. Moreover, such performances are less sensitive to credibility thus making the application of similarity measurement applicable to a wider context.

5.6 Threats to Validity

In the following we discuss the threats to validity related to our empirical procedure using the terminology and concepts reported by Wohlin et al. in [48].

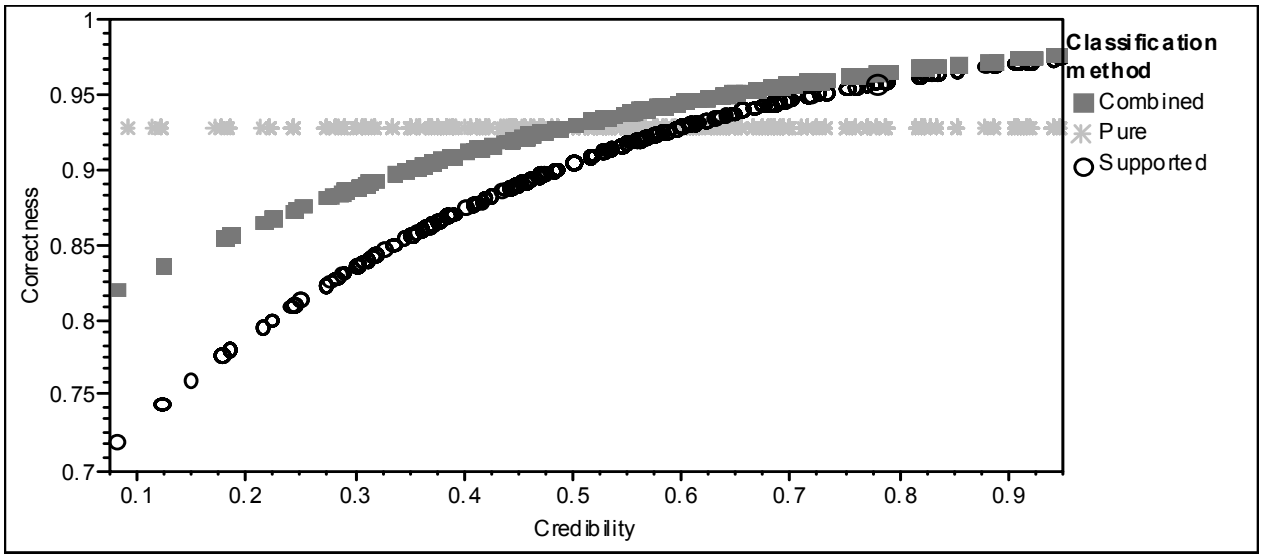

Figure 12. Correctness versus *Credibility* and type of linkage method – Logistic regression.

5.6.1 Conclusion validity

Conclusion validity is concerned with the experimenter's ability to draw conclusions about relations between the treatment and the outcome of the experiment [48]. The threats related to statistical assumptions and type-I error rates are negligible due to the fact that we used non-parametric tests and obtained very small p-values on most statistical tests. Since measurement was automated and the data validated (Section 5.1), in our view threats related to the reliability of measurement are unlikely. The adopted cross-randomized design allows subjects to apply all the treatments in a random order. This is expected to mitigate the threats related to reliability of treatment implementation, random irrelevance in experimental setting, and random heterogeneity of subjects.

5.6.2 Internal validity

Internal validity issues arise from variables that are confounded with independent variables, thus making the effect of the latter on the dependent variable difficult to analyze. The ability of the experimenter to establish causality between treatments and the observed effect on the dependent variable is then compromised. Threats related to maturation should be unlikely because of our experimental design. Regarding the ability of subjects to deal with requirements written in English, we are pretty confident about the subjects' ability to deal with technical English because many of the books in courses they passed were written in English. Social threats should be low as well because subjects had nothing to gain from the actual outcome of the experiment.

5.6.3 Construct validity

Construct validity is concerned with the validity of measurement. Our measurement was very similar to that in [3]. We avoided mono-operation bias by adopting a large amount of requirement pairs to link and an equal amount of equivalent and nonequivalent pairs. Threats of mono-method bias should be unlikely since we adopted six different measures for correctness and they all yielded similar results. This also mitigates the fishing threat. Regarding experimenter expectancies, we have seen the use of PROUD was on average counterproductive; moreover, we weren't interested in demonstrating that the usage PROUD was beneficial but we wanted to investigate how the support provided by similarity measurement was dependent on its reliability to properly measure equivalence. Though the adoption of a balanced proportion of equivalent requirements is not realistic, this balanced design facilitated the experiment data analysis and yielded a better correctness prediction model, that could then yield more realistic results when applied to the case study industrial requirements.

5.6.4 External validity

External validity is related to generalization of the results. Though controlled experiments with students present general challenges with respect to external validity, the students were suitably trained for the tasks. The controlled experiment was relied upon to model the relationship between the correctness and speed of establishing requirements equivalence and similarity, while accounting for the credibility of similarity measurement. However, an industrial case study was used to obtain realistic distribution for credibility, and thus assess in a realistic fashion what benefits to expect from using similarity measurement.

6 Conclusions and Future Work

The application context of our work is the defense/aerospace domain where a significant proportion of requirements tend to be redundant due to large numbers of projects, requirements per project, and diverse sources of requirements.

This paper makes the following contributions:

1. **We propose a novel approach to evaluate the benefit of showing requirement similarity measurement to**

analysts in terms of effectiveness and efficiency for linking equivalent requirements: The approach consists in combining an industrial case study and a controlled experiment to obtain both sufficient control and realism. One important point is our investigation of how the level of support provided by similarity measurement can be affected by the requirements set on which it is used and the specific properties of the similarity measure. Hence, we designed an experiment with known equivalence links among requirements and we defined a "credibility" measure to assess the reliability of similarity measurement, that is how accurately it reflects the actual correctness of links between requirements. We then used credibility as an interaction factor with similarity in our data analysis. Once the interaction effect of credibility with similarity is modeled using the experiment data, we used a realistic credibility distribution from an industrial case study to assess the likely benefits of using similarity measurement on realistic requirements. Our controlled experiment involved thirty graduate students and an industrial case study with one thousand requirements pairs with known equivalence links and computed similarity and credibility values.

2. **We show that similarity measurement has an interaction effect with credibility**: Results suggest that similarity measurement must be above a given credibility level to be beneficial. When credibility is low, it may even be counterproductive and negatively affect the correctness of linking equivalent requirements. Such a result can be used by future evaluations of similarity measures; they can estimate the likely benefits of showing similarity measurements for classifying requirements by adopting the interaction effects observed in our experiment.
3. **We evaluate the benefits of similarity measurement on industrial requirements**: Given the level of credibility observed on a large industrial case study, using the similarity measurement is expected, on average: 1) to improve by 20% the amount of equivalence links established per minute and 2) to decrease by 40% the amount of incorrectly established links. This confirms and extends past empirical studies regarding the practical benefit of using similarity measurement when establishing equivalence links among requirements, both in terms of correctness and speed.
4. **We propose and evaluate an approach for combining human decisions and similarity measurement**: The idea is to combine and weight measurement and human decisions in such a way to alleviate their individual weaknesses. We empirically evaluated the benefits of building prediction models based on these two sources of information by using various statistical and machine learning techniques. A practical result is that such hybrid models make the use of similarity measurement beneficial for lower values of credibility, both for correctness and speed, thus making the application of similarity measurement applicable to a wider context. Moreover, such hybrid models outperform the performance of showing similarity measurement, and significantly so for low levels of credibility. In conclusion, the combination of human decisions and similarity measurements makes the use of the latter more effective, efficient and more widely applicable.

Though the presented study focuses only on linking equivalent requirements, there is no reason why the above results should not be independent of the specific type of similarity measure or artifacts (expressed in natural language) being linked.

Future works include the evaluation of other similarity measures and the use of machine learning to predict the number of remaining artifacts to link.

Acknowledgement

The work has been partially supported by SELEX SI under the research grant: SSI-DISP/07/08. The authors thank the engineering department of SELEX SI for the provided support and insightful interactions, in particular we thank: Emanuela Barbi, Vincenzo Sabatino and Emiliano Pandolfi. Lionel Briand was supported by Det Norske Veritas in the ModelME! Project.

References

[1] B. Nuseibeh, and S. Easterbrook, “Requirements engineering: a roadmap,” in Proceedings of the Conference on The Future of Software Engineering, Limerick, Ireland, 2000.

[2] J. Natt och Dag, B. Regnell, P. Carlshamre, M. Andersson, and J. Karlsson, “A Feasibility Study of Automated Support for Similarity Analysis of Natural Language Requirements in Market-Driven Development,” *Requirements Engineering journal,* vol. 7, no. 1, 2002.

[3] J. Natt och Dag, T. Thelin, and B. Regnell, “An experiment on linguistic tool support for consolidation of requirements from multiple sources in market-driven product development,” *Empirical Software Engineering,* vol. 11, no. 2, pp. 303-329, 2006.

[4] J. H. Hayes, A. Dekhtyar, and S. K. Sundaram, “Advancing candidate link generation for requirements tracing: the study of methods,” *IEEE Transactions on Software Engineering,* vol. 32, no. 1, pp. 4-19, 2006.

[5] J. H. Hayes, and A. Dekhtyar, “Humans in the traceability loop: can't live with 'em, can't live without 'em,” in Proceedings of the 3rd international workshop on Traceability in emerging forms of software engineering, Long Beach, California, 2005.

[6] D. Poshyvanyk, A. Marcus, R. Ferenc, and T. Gyimty, “Using information retrieval based coupling measures for impact analysis,” *Empirical Software Engineering,* vol. 14, no. 1, pp. 5-32, 2009.

[7] G. Antoniol, G. Canfora, G. Casazza, A. De Lucia, and E. Merlo, “Recovering Traceability Links between Code and Documentation,” *IEEE Transactions on Software Engineering,* vol. 28, no. 10, pp. 970-983, 2002.

[8] G. Antoniol, A. Cimitile, and G. Casazza, “Traceability Recovery by Modeling Programmer Behavior,” in Proceedings of the Seventh Working Conference on Reverse Engineering (WCRE'00), 2000.

[9] J. Cleland-Huang, R. Settimi, C. Duan, and X. Zou, “Utilizing Supporting Evidence to Improve Dynamic Requirements

Traceability," in Proceedings of the 13th IEEE International Conference on Requirements Engineering, 2005.
[10] A. De Lucia, R. Oliveto, and P. Sgueglia, "Incremental Approach and User Feedbacks: a Silver Bullet for Traceability Recovery," in Proceedings of the 22nd IEEE International Conference on Software Maintenance, 2006.
[11] X. Zou, R. Settimi, and J. Cleland-Huang, "Improving automated requirements trace retrieval: a study of term-based enhancement methods " *Empirical Software Engineering,* vol. To Appear, 2009.
[12] A. De Lucia, R. Oliveto, and G. Tortora, "Assessing IR-based traceability recovery tools through controlled experiments," *Empirical Software Engineering; An International Journal,* vol. 14, no. 1, pp. 57-92, 2009.
[13] A. Marcus, and J. Maletic, "Recovering documentation-to-source-code traceability links using latent semantic indexing," in Proceedings of the 25th International Conference on Software Engineering, Portland, Oregon, 2003.
[14] M. Svahnberg, A. Aurum, and C. Wohlin, "Using students as subjects - an empirical evaluation," in Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, Kaiserslautern, Germany, 2008.
[15] M. Host, B. Regnell, and C. Wohlin, "Using Students as Subjects; A Comparative Study of Students and Professionals in Lead-Time Impact Assessment," *Empirical Software Engineering Journal,* vol. 5, no. 3, pp. 201-214, 2000.
[16] I. Witten, and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2005.
[17] K. Ryan, "The role of natural language in requirements engineering," in Proceedings of IEEE International Symposium on Requirements Engineering, 1993, pp. 240-242.
[18] C. Duan, and J. Cleland-Huang, "Clustering support for automated tracing," in Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering, Atlanta, Georgia, USA, 2007.
[19] X. Zou, R. Settimi, and J. Cleland-Huang, "Term-based Enhancement Factors for Improving Automated Requirement Trace Retrieval," in ACM International Symposium on Grand Challenges of Traceability, 2007.
[20] M. Lormans, and A. van Deursen, "Can LSI help Reconstructing Requirements Traceability in Design and Test?," in Proceedings of the Conference on Software Maintenance and Reengineering, 2006.
[21] M. Di Penta, S. Gradara, and G. Antoniol, "Traceability Recovery in RAD Software Systems," in Proceedings of the 10th International Workshop on Program Comprehension, 2002.
[22] Yadla S., J. H. Hayes, and A. Dekhtyar, "Tracing requirements to defect reports: an application of information retrieval techniques," *A NASA Journal, Information Systems Software Engineering*, 2005.
[23] J. Natt och Dag, V. Gervasi, S. Brinkkemper, and B. Regnell, "Speeding up Requirements Management in a Product Software Company: Linking Customer Wishes to Product Requirements through Linguistic Engineering," in Proceedings of the Requirements Engineering Conference, 12th IEEE International, 2004.
[24] V. Gervasi, and D. Zowghi, "Reasoning about inconsistencies in natural language requirements," *ACM Transaction on Software Engineering Methodologies,* vol. 14, no. 3, pp. 277-330, 2005.
[25] E. Stierna, and N. Rowe, "Applying information-retrieval methods to software reuse: a case study," *Inf. Process. Manage.,* vol. 39, no. 1, pp. 67-74, 2003.
[26] N. Niu, and S. Easterbrook, "On-Demand Cluster Analysis for Product Line Functional Requirements," in Proceedings of the 2008 12th International Software Product Line Conference, 2008.
[27] I. John, "Capturing Product Line Information from Legacy User Documentation " *Software Product Lines*, T. Kakola and J. C. Duenas, eds., pp. 127-159, Heidelberg: Springer Berlin 2006.
[28] V. Alves, C. Schwanninger, L. Barbosa *et al.*, "An Exploratory Study of Information Retrieval Techniques in Domain Analysis," in Proceedings of the 2008 12th International Software Product Line Conference, 2008.
[29] I. John, J. Knodel, K. Robby, and T. Schulz, "Efficient scoping with CaVE - a case study," in IESE-Report, 039.07/E, 2007.
[30] J. H. Hayes, A. Dekhtyar, and S. Sundaram, "Text mining for software engineering: how analyst feedback impacts final results," *SIGSOFT Softw. Eng. Notes,* vol. 30, no. 4, pp. 1-5, 2005.
[31] K. Pohl, G. Böckle, and F. J. v. d. Linden, *Software Product Line Engineering: Foundations, Principles and Techniques* Springer, 2005.
[32] J. Bosch, *Design and Use of Software Architecture: Adopting and Evolving a Product-Line Approach*, Boston: Addison-Wesley, 2000.
[33] M. Jazayeri, A. Ran, F. V. D. Linden, and P. V. D. Linden, *Software Architecture for Product Families: Principles and Practice*, Boston: Addison-Wesley, 2000.
[34] F. van der Linden, K. Schmid, and E. Rommes, *Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering*: Springer-Verlag New York, Inc., 2007.
[35] P. Clements, and C. Kreuger, "Point - Counterpoint: Being Proactive Pays Off - Eliminating the Adoption " *IEEE Software,* vol. 19, no. 4, pp. 4, 2002.
[36] P. Clements, and L. Northrop, *Software Product Lines: Practice and Patterns*, Boston: Addison-Wesley, 2002.
[37] J. Bosch, "On the Development of Software Product-Family Components," *Proceedings of the Third International Conference on Software Product Lines*, R. L. Nord, ed., Boston, MA, USA: Springer LNCS 3154, 2004.
[38] P. C. Clements, J. D. McGregor, and S. G. Cohen, *The Structured Intuitive Model for Product Line Economics (SIMPLE),* CMU/SEI-2005-TR-003, 2005.
[39] I. John, J. Knodel, T. Lehner, and D. Muthig, "A practical guide to product line scoping," in Software Product Line Conference, 2006 10th International, 2006, pp. 3-12.
[40] K. Kang, S. Cohen, J. Hess, W. Novak, and A. Peterson, "Feature-Oriented Domain Analysis (FODA) Feasibility Study."
[41] J. Natt och Dag, V. Gervasi, S. Brinkkemper, and B. Regnell, "A Linguistic-Engineering Approach to Large-Scale Requirements Management," *IEEE Software,* vol. 22, no. 1, pp. 32-39, 2005.
[42] H. Gomaa, *Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures*: Addison Wesley Longman Publishing Co., Inc., 2004.
[43] P. Runeson, and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering,* vol. 14, no. 2, pp. 131-164, 2009.
[44] C. Manning, P. Raghavan, and H. Schtze, *Introduction to Information Retrieval*: Cambridge University Press, 2008.
[45] J. L. Fleiss, B. Levin, and M. C. Paik, *Statistical Methods for Rates and Proportions*, New York: Wiley-Interscience, 2003.
[46] N. Juristo, and A. M. Moreno, *Basics of Software Engineering Experimentation* Springer, 2006.
[47] M. Hall, and E. Frank, "Combining Naive Bayes and Decision Tables," in In Proc 21st Florida Artificial Intelligence Research Society Conference, Miami, Florida, 2008.
[48] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslen, *Experimentation in Software Engineering: an Introduction*: Springer, 2000.