On Human Analyst Performance in Assisted Requirements Tracing: Statistical Analysis

Alex Dekhtyar*, Olga Dekhtyar†, Jeff Holden*, Jane Huffman Hayes‡, David Cuddeback*, and Wei-Keat Kong‡

* *Department of Computer Science, Cal Poly, San Luis Obispo, CA, USA*
Email: {dekhtyar, jholden}@calpoly.edu, david.cuddeback@gmail.com
†*Department of Statistics, Cal Poly, San Luis Obispo, CA, USA. Email: odekhtya@calpoly.edu*
‡*Department of Computer Science, University of Kentucky, Lexington, KY, USA*
Email: hayes@cs.uky.edu, wkkong1@uky.edu

***Abstract*—Assisted requirements tracing is a process in which a human analyst validates candidate traces produced by an automated requirements tracing method or tool. The assisted requirements tracing process splits the difference between the commonly applied time-consuming, tedious, and error-prone manual tracing and the automated requirements tracing procedures that are a focal point of academic studies. In fact, in software assurance scenarios, assisted requirements tracing is the only way in which tracing can be at least partially automated. In this paper, we present the results of an extensive 12 month study of assisted tracing, conducted using three different tracing processes at two different sites. We describe the information collected about each study participant and their work on the tracing task, and apply statistical analysis to study which factors have the largest effect on the quality of the final trace.**

***Keywords*-traceability, tracing, human factors**

I. Introduction

A large non-disclosable financial corporation, NDFC, finds that it has a number of pressing issues: 1) it is being assessed fines for failure to adequately comply with Sarbanes-Oxley Act (SOX) [1] with respect to a traceability trail for its software that handles client stock transactions; 2) a recent scare has caused senior management to hire an independent assessment team from an outside firm to perform an audit to ensure that malicious code/trap doors/back doors do not exist in critical code applications; 3) a rash of software failures are being rapidly repaired as a new app, iTradeu, is being readied for its initial launch; the developers are struggling to debug and then retest the app in a timely manner. What do these three scenarios have in common? Traceability.

A traceability process and/or tool could be applied to the audit trail information to assist with SOX compliance (issue #1). The same process/tool could be used to trace all code back to requirements. If code exists that does not trace back to a requirement, it should be examined to ensure that it is not malicious code (issue #2). With a traceability process/tool, the iTradeu (issue #3) developers could trace failures (source artifact) to requirements (target artifact), design, and/or features to help locate the code faults, debug the code, and then use the trace information to determine what tests to rerun. With all the advantages that tracing could offer to NDFC, *why are they not using such a tool/process?*

First, many organizations undertake *manual* tracing, perhaps with the assistance of a word processing tool or spreadsheet. Such a process is boring, tedious, and time-consuming. As a result, it is also error prone [11]. Second, once traceability is established for a project, the project artifacts quickly change, thus necessitating traceability updates. Third, there is a lack of an industry-accepted tracing tool.

Automation of the tracing process, as studied previously [2], [19], [11], [23], [21], [20], [9], [22], could go a long way toward addressing many of the drawbacks mentioned above. Consider a process for tracing using a software tool versus a manual tracing process as described in Table I. In both scenarios, the human analyst plays a large, but qualitatively different, role in the tracing process. Each step in Table I will be performed faster in the *tracing using a software tool* scenario: software will deliver a candidate trace[1] much faster than a human analyst can read through a pair of artifacts of non-trivial size. In step three, when tracing using a software tool, the analyst is expected to mostly validate the suggestions provided by the automated method. Analyst effort on this step is expected to depend on the specifics of the software tool: how well the tool finds true links, how many false positive candidate links the tool retrieves, how much analyst effort is required to accept/reject a candidate link using the tool, etc. However, research shows that analysts working with a software tool based on any of the existing automated tracing methods [2], [19], [11], [23], [21] will examine significantly fewer candidate links than an analyst performing manual tracing [2], [11].

In this paper, we use the term ***assisted requirements tracing*** or ***assisted tracing*** to refer to *a tracing process in which a human analyst engages with an automated requirements tracing software tool to perform the assigned tracing task*. In the software processes discussed above, assisted tracing can provide the *best of both worlds*, allowing

[1]Traces, traceability matrices (TMs), and links are candidate until a human analyst vets them

978-1-4577-0924-1/11/$26.00 © 2011 IEEE

Table I
SCENARIOS FOR MANUAL TRACING AND TRACING WITH A SOFTWARE TOOL

Step	Tracing with software tool	Manual tracing
Step 1	Human launches tool to trace a pair of artifacts to each other	Analyst reads the text of a *source* artifact/document
Step 2	The tool returns a candidate traceability matrix (TM) between the artifacts	The analyst reads the text of a *target* artifact/document
Step 3	The human vets each link in the candidate traceability matrix and renders a decision This is repeated until all candidate target elements retrieved for every source element have been reviewed	The human reads the first source element, searches the target artifact for matches and records the matches This loop continues until all source elements have been processed

both humans and tracing software to do what they do best. We are interested in what constitutes a good assisted tracing process as well as ways to evaluate such a process.

Automated tracing methods are usually evaluated using *precision* and *recall* which measure the overall accuracy of the recovered traceability matrix (TM). Research in automated traceability [2], [19], [11], [23], [21] concentrates on improving precision and recall over methods studied earlier, and has as its ultimate goal reaching the "Holy Grail" of 100% precision and 100% recall.

The study of assisted tracing adds a wrinkle to the traditional evaluation methodology. While we are still interested in trace accuracy as measured by precision and recall, it is the accuracy of the *traceability matrix submitted by the human analyst* (also called the final TM) that matters. Cuddeback et al. [6] reported on the results of a preliminary study of assisted traceability, focused exclusively on making hypothetical observations on what caused specific participant performance. In that study, 26 participants in two sites were given candidate TMs of varying quality for vetting. Surprisingly, the best improvement in accuracy (comparing the vetted TM to the starting TM) was seen by the participants who were given TMs of the lowest accuracy [6].

The study described in this paper is a significant expansion of Cuddeback et al.'s study [6] We have conducted additional studies of assisted tracing, using two more tracing procedures (one manual and one involving a different software tool) at two experimental sites for a total of 84 participants[2]. This paper undertakes a statistical analysis to formally determine what affects human performance the most. Specifically, this paper contributes: a) two additional rounds of assisted traceability experiments at two experimental sites, b) a multi-variate analysis of 11 independent variables describing participant experience with the tracing experiment to identify statistically significant factor(s) affecting analyst performance, and c) a formal statistical re-examination of the (informal) findings from earlier work [6] studying the effect of the accuracy of the candidate traceability matrices provided to the analysts on their performance. Specifically, we study these questions:

Q1. Is the effect of the accuracy of the initial TM on the accuracy of the final TM statistically significant?

Q2. Are the effects of any observed independent variables on the accuracy of the final TM statistically significant (when controlled by the initial TM accuracy)?

Q3. Which group of independent variables has a higher effect on the accuracy of the final TM: the variables measuring accuracy of the initial TM or the observed independent variables?

[2]Including the 26 participants from Cuddeback et al. [6].

Table II
AN OVERVIEW OF PARTICIPANTS DURING THE THREE TRACEABILITY EXPERIMENTS

Cohort	Date	Location	# of participants	Tool used
1	Dec 09	University A	16	Retro
1	Dec 09	University B	10	Retro
1	Apr 09	University B	7	Retro
All 1		**A and B**	**33**	**Retro**
2	**Nov 10**	**University A**	**38**	**Manual**
3	Dec 10	University A	8	RETRO.net
3	Dec 10	University B	5	RETRO.net
All 3		**A and B**	**13**	**RETRO.net**

The rest of the paper is organized as follows. Section II provides background and related work on assisted tracing and introduces basic traceability concepts and measures. Section III describes the experiments. Section IV presents the statistical analysis and results. Section V concludes.

II. BACKGROUND AND RELATED WORK

Requirements traceability is defined as the "ability to describe and follow the life of a requirement, in both a forwards and backwards direction" [8]. The output of the tracing process is a requirements traceability matrix (RTM or TM) which specifies the connections between elements of two artifacts. Multiple studies applied information retrieval techniques to automatically generate TMs [10], [2], [3], [11], [19]. In these studies, the quality of the TM was measured primarily using *precision*, *recall*, and f*-measure* (see below). Most of the methods studied were able to achieve high recall, but with low precision.

A. Measures

Consider a tracing process consisting of a set of high-level elements H of size M and a set of low-level elements D of size N. For a particular requirement $q \in H$, let n_q be

the number of candidate links between q and the elements in D that a tracing process returns. Let r_q be the number of correct links and R_q be the actual number of correct links (from an expert-prepared answer set).

Recall is defined as the percentage of correct links that are found, while *precision* is the percentage of retrieved candidate links that are correct [11]:

$$recall = \frac{\sum_{q \in \mathcal{H}} r_q}{\sum_{q \in \mathcal{H}} R_q}; \qquad precision = \frac{\sum_{q \in \mathcal{H}} r_q}{\sum_{q \in \mathcal{H}} n_q} \tag{1}$$

F-measure is the harmonic mean of precision and recall, defined formally below. In this definition, b represents the balance between precision and recall where $b < 1$ favors precision and $b > 1$ favors recall.

$$f_b = \frac{1 + b^2}{\frac{b^2}{recall} + \frac{1}{precision}} \tag{2}$$

Contemporary studies of automated tracing methods implicitly equate TM accuracy (as calculated by precision, recall, and F-measure) with TM quality [10], [2], [3], [11], [19]. However, in mission-critical software assurance, a TM produced by an automated system must be validated by a human analyst responsible for the assurance guarantees.

B. Study of the Analyst During Tracing

In earlier work [13], [15], Hayes and Dekhtyar asked whether it is, in fact, true that more accurate initial candidate TMs lead to more accurate analyst-validated TMs. While their initial study [13] involved only four analysts, it provided anecdotal evidence that this may not be the case.

Our traceability research group has conducted a number of studies to further investigate analyst behavior during the tracing process and reported initial results [6], [5]. Two of the most important trends observed were: 1) participants were unable to recover the true TM or reach a consensus of what that TM should be, and 2) participants given the highest quality candidate TMs to validate almost uniformly degraded the TM accuracy, while participants given the lowest quality candidate TMs almost uniformly improved the accuracy greatly.

A similar recent study, conducted by Egyed et al. [7], while primarily focusing on human analyst effort, supports our overall observation that human analysts are fallible in their work with candidate traceability matrices. Our present study goes one step further and establishes that the level of human fallibility is somewhat predictable.

III. Experimental Design

In this section, we discuss the experimental design, the data collected, and threats to validity.

A. How we collected data

We conducted a series of experiments examining analyst performance in assisted tracing tasks (see Table II). The initial experiment [6] involved 26 subjects performing a tracing task using REquirements TRacing On-target (RETRO) [12], a special-purpose requirements tracing tool written in Java. Cuddeback's thesis [5] includes an extra cohort of seven subjects who used the same tracing process. We conducted two follow-up experiments, one using an improved and simplified version of RETRO called RETRO.net (written to address usability and stability issues with the original RETRO but does not differ in functionality), and the other asking the analysts to validate the TM manually using hard-copy artifacts without software assistance. In what follows we refer to these experiments as the *RETRO experiment*, the *RETRO.net experiment* and *the manual experiment*.

The RETRO and RETRO.net experiments were conducted at two sites: California Polytechnic State University and University of Kentucky. The manual study was only conducted at one of the universities; we hope to repeat it at the other site in the future. All participants in the studies were students enrolled in software engineering courses. All were provided a short introduction to requirements tracing. Most of the participants were junior, senior, or graduate students.

In RETRO and RETRO.net experiments, a pre-experiment survey was given to the participants in order to gauge prior experience and overall comfort with tracing. The research team utilized the responses to separate participants into two groups, an experienced group and a group that lacked tracing experience. In each group, participants were assigned starting TMs in a way that ensured that TMs with different accuracy were evenly distributed among participants with both levels of experience. The manual study had no pre-experiment survey, but most of the questions from it were asked in the post-experiment survey, so the same information was collected. The manual study took place in an entry-level software engineering course, and thus we did not expect (and did not observe) significant levels of tracing experience among the participants, and did not need to use pre-experiment survey data to assign starting TMs.

In all three cohorts, participants were asked to review a candidate TM, referred to as the *initial* or *starting TM*, with pre-defined precision and recall values. The assignment of the TM was made by the researchers. After completing the tracing task, participants were asked to submit their final TM and complete a post-experiment survey that asked for their reactions to tracing (how prepared they were for the task, how difficult it was, if they would prefer tracing manually or with a tool, etc. [5]). Two questions in the post-survey asked the participants to identify how much effort they spent on the two main types of activities we expected: (a) validating candidate links found in the initial TM, and (b) searching for links that were missing from the initial TM. For the

Table III
BASELINE INDEPENDENT VARIABLES

Variable	Abbreviation	Scale
Initial Precision	SPrec	[0,1]
Initial Recall	SRec	[0,1]
Initial F2	SF2	[0,1]
Initial Quadrant	SQuadrant	$\{Q1, Q2, Q3, Q4\}$

RETRO and RETRO.net experiments, the participants were also asked to submit a log of their actions. In the RETRO study, the log was a hardcopy document manually created and maintained by participants. RETRO.net software implemented automatic activity logging and the participants were asked to submit the generated log file.

All three studies utilized the same dataset, a BlueJ plugin Java code formatter named ChangeStyle. This dataset contains 32 requirements and 17 system tests. The research team generated and validated the *golden standard* TM which contains 23 links from requirements to tests [6][3]. This dataset was chosen for the experiments because: (a) the domain is easily understood by participants, and (b) its size makes the validation task achievable in about one hour.

In this paper, we concentrate on analyzing common information collected from the experiments. Some of the aspects of our RETRO.net experiment, which involved tracking analyst behavior, are reported elsewhere [16].

B. What data we collected

For all studies, we assembled a rich set of meta-information from the pre- and post-experiment surveys as well as information concerning initial and final TMs for each analyst. Tables III, IV, and V provide an overview of the information that we collected, broken into three categories:

1) **Baseline independent variables.** (Table III). These variables specify the accuracy of the initial TM.
2) **Observed independent variables.** (Table IV). These variables contain is information about the experiment participants and their work on the tracing task. This information was either part of the experimental design (location, software used) or collected from the pre- and post-experiment surveys. Of the 11 variables collected, one (Time) is continuous; the remaining 10 are either nominal or ordinal (see Type column in Table IV).
3) **Response (a.k.a. dependent) variables.** (Table V). Our dependent variables measure the accuracy of the final TMs submitted by the participants. These variables fall into two groups: measures of the absolute accuracy of the final TM and "Delta" variables that measure the change between the initial and final TM.

[3]The validation process for the *golden standard* is discussed in detail elsewhere [6], [5]. In short, a candidate golden standard (answerset) was assembled from the artifacts of the software engineering course which implemented ChangeStyle; that candidate TM was then examined, link-by-link, by multiple researchers from our research group, until consensus was reached on each link.

Table V
RESPONSE (DEPENDENT) VARIABLES

Variable	Abbreviation	Scale
Final Precision	FinPrec	[0,1]
Final Recall	FinRec	[0,1]
Final F2	FinF2	[0,1]
Delta Precision	$\Delta Prec$	[-1,1]
Delta Recall	ΔRec	[-1,1]
Delta F2	$\Delta F2$	[-1,1]

In earlier work [6], the main focus was on how the baseline variables impact the dependent variables (albeit, no statistical analysis was presented). In this paper, we expand that work by: (a) presenting the results of the statistical analysis, and (b) comparing the effect of the baseline independent variables and the observed independent variables on the values of the dependent variables.

C. Threats to validity

Our study was subject to a number of threats to validity. We addressed the threat to **conclusion validity** by ensuring that all data assumptions for the statistical techniques were met and perfoming our analysis with the assistance of an experienced statistician. A threat to **internal validity** would be the use of a golden standard traceability matrix developed by a subset of the authors. This is standard practice in traceability studies as actual or true traceability matrices are rarely available. Examples of this practice can be seen in a number of previous papers in this conference (Huang et al. built answer sets for three datasets, for example [4]). There are precedents for student-built datasets in traceability research (Waterloo dataset, iTrust dataset, for example) [14], [18]. Another threat to internal validity would be the limited time given to participants to perform the task. We were constrained in the amount of time we had to undertake the experiment. We felt that it was best to use a small dataset that could be traced in the class period for this initial work. The dataset is similar in size to those used by Egyed et al. [7] (bearing in mind that their subjects had 90 minutes to work versus 60 minutes in our case). Dependent variable issues that threaten **construct validity** were reduced by the use of standard Information Retrieval measures. Our work with student participants represented a threat to **external validity**. However, Host et al. note that students can perform small tasks of judgement the same as professionals with no significant differences [17]. Also, it has been observed by Tichy et al. [24] that students can serve well for determining trends, if appropriately trained. There is also precedence in traceability work: other traceability studies have used students with low levels of industry experience to represent new people joining a company [7]. Motivation of the participants is also a threat to external validity found in all our experiments. Students were given extra credit for participating in the experiment, but the points awarded were not tied to the quality of their

Table IV
OBSERVED INDEPENDENT VARIABLES

Variable	Abbreviation	Scale	Type	Scale Details
Procedure used	Procedure	{Retro, Manual, RETRO.net}	Nominal	tracing procedure used by participant
Location	Location	{CP, UK}	Nominal	Cal Poly or University of Kentucky
Software Engineering Experience	SEExp	$\{0, 1, 2\}$	Ordinal	based on number of SE courses and industry experience
Tracing Experience	TRExp	$\{0, 1\}$	Nominal	reported use of tracing in coursework or industry
Time to preform tracing task	Time	# minutes	Cont.	number of minutes it took to complete the task
Grade Level	Grade	{F, Soph, J, S, G}	Nominal	participant grade level
Confidence with tracing	TrConf	1 – 5	Ordinal	self-reported level (1: lowest, 5: highest)
Opinion on Tool vs. Manual	Opinion	{Man, SW}	Nominal	participant's (post-task) preferred way of tracing
Effort on searching for omitted links	MissingEff	0 – 5	Ordinal	self-reported (0: never, 5: almost every link)
Effort on validating offered links	ValidEff	0 – 5	Ordinal	self-reported (0: never, 5: every link)
How prepared the analyst felt	Prepared	1 – 5	Ordinal	Self-reported post-task (1: not at all, 5: very prepared)

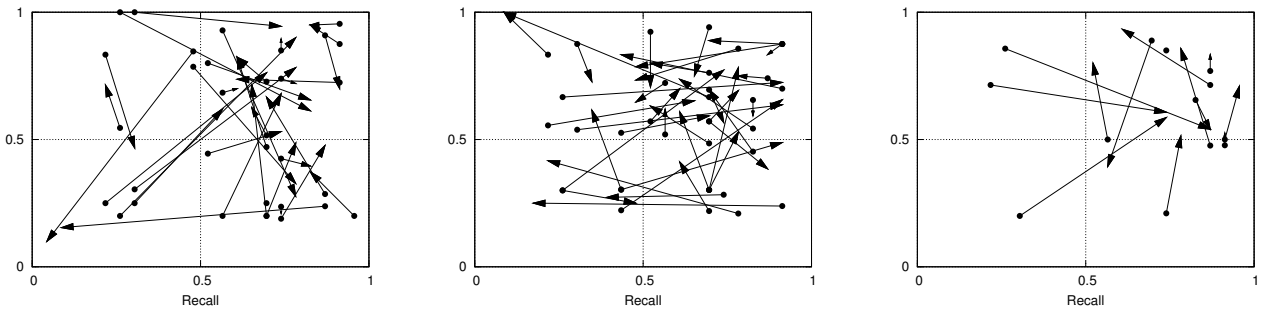

Figure 1. Results from our three studies: **(a)** used RETRO, **(b)** traced manually, **(c)** used RETRO.net.

work. Had researchers provided points based on quality of work, a different threat to validity would have been introduced (requiring mitigation of the threat versus reward dynamic). An additional external threat deals with our use of only one small, student-built dataset. Our findings may not be the same if we were to use a different dataset. The only way to overcome this threat is to repeat the work on a real project, which remains as future work.

IV. RESULTS AND ANALYSIS

We present information on analyst performance, statistical analysis undertaken, and observed results.

A. Analyst Performance

Earlier work [6] presented a collection of graphs illustrating the results of the experiment. Here, we present some of these graphs for the entire body of our experiment. The main visualization method employed in Cuddeback et al. [6] is to render, for each participant, the initial and the final TMs in the *precision–recall* space, and to draw a vector from the initial to the final TM.

Figure 1 presents the results of our three studies broken down by experiment. Figure 1(a) depicts the RETRO experiment [5], 1(b) shows the results of the manual experiment, and 1(c) shows the results of the RETRO.net experiment. Figure 2 shows the same results in two ways: graphs 2(a) and 2(d) plot the locations of all starting and final TMs, respectively. The remaining graphs show the analyst performance, for ease of visualization, by the *quadrant* of the initial candidate TM.

Cuddeback et al. [6] made the following observations:

- Analysts given *low-precision, low-recall* TMs **drastically improved** their accuracy.
- Analysts given *low-precision, high-recall* TMs tended to improve precision at the price of lower recall.
- Analysts given *high-precision, low-recall* TMs tended to improve recall, but usually at the cost of lowering precision.
- Analysts given *high-precision, high-recall* TMs tended to *slightly decrease* the overall accuracy of the TM, but they could do it in a number of different ways.
- Analysts appeared to possess good intuition about the actual size of the golden standard TM.
- No analyst recovered the golden standard TM.

As can be seen from Figures 1 and 2, with the exception of a few outliers (present in each experiment), analyst behavior observed in earlier work [6] **is informally confirmed** in this study. Participants in the manual and RETRO.net experiments appear, based on these graphs, to have exhibited essentially the same behavior as participants in the RETRO study. In 84 observed attempts, no participant recovered the true trace; *however*, every true link was found by at least one participant. We move now to **formal confirmation**.

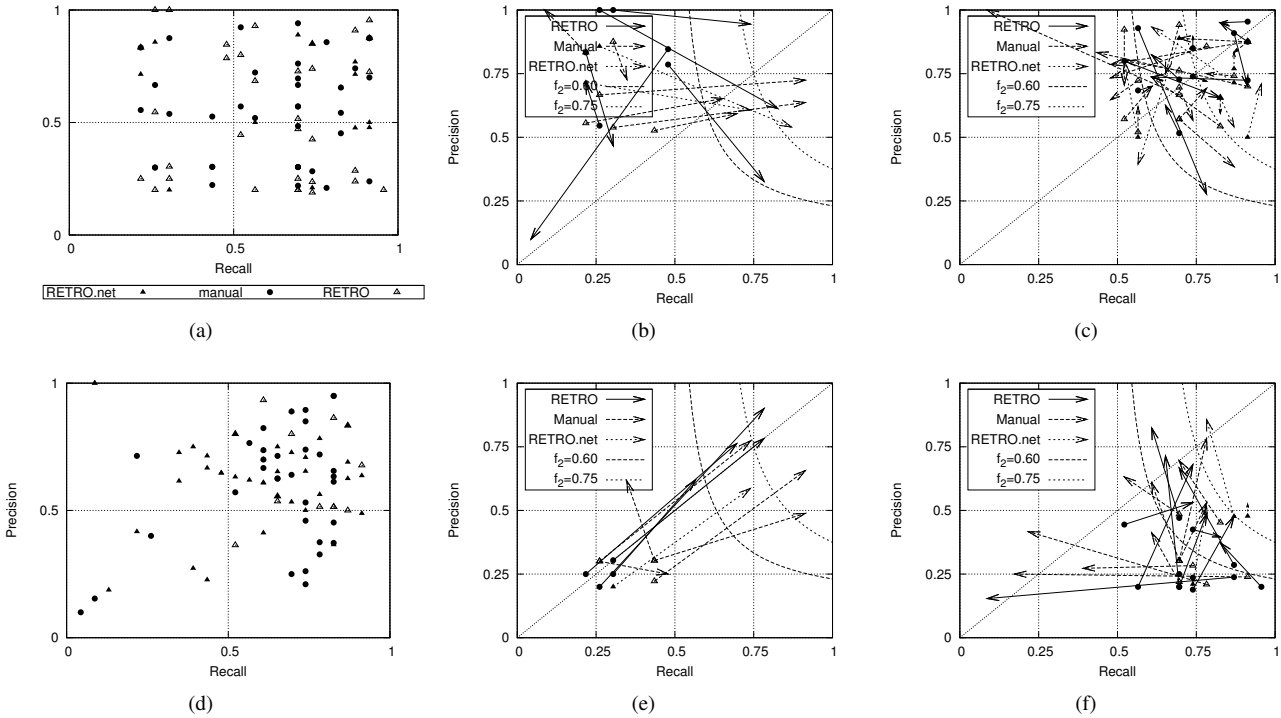

Figure 2. Results of our traceability study: **(a)**, **(d)**: distribution of assigned and submitted TMs. **(b),(c),(e),(f)**: performance of individual participants by accuracy of assigned TM.

B. Statistical Analysis

To better understand what went on in our experiments, we conducted multivariate statistical analysis designed to discover the key factors influencing the accuracy of the final TM and asked the questions found at the end of Section I.

Baseline independent variables (Q1). Table VI shows the influence of the pair of independent variables Initial Precision and Initial Recall on each of our response variables using multiple regression. We report the adjusted R-square value, R^2_{adj}, the F-value, and the significance level (p-value) for each model. As can be seen from the table, the initial accuracy of the traceability matrix has a statistically significant effect on the precision of the final TM, as well as on changes in precision, recall, and F2-measure[4]. There is no statistically significant effect on recall and F2-measure of the final TM.

We can use the Initial F2-measure as a one-dimensional surrogate for the intial precision and initial recall. We studied the influence of the Initial F2-measure on our response variables using linear regression. The results are summarized in Table VII. As can be seen from the table, initial F2-measure statistically significantly influences final precision, the change in recall, and the change in precision and the F2-measure. It does not statistically significantly influence final recall, final F2-measure and the change in precision.

Finally, we broke all our initial TMs *by quadrant* using values of 50% precision and 50% recall as boundaries. Since Initial Quadrant is a categorical variable, we used one-way ANOVA to study its relationship with each of our response variables. Table VIII shows the results of this analysis. In the table, QI is the *low-precision, low-recall* quadrant, QII is the *low-precision, high-recall* quadrant, QIII is the *high-precision, low-recall* quadrant, and QIV is the *high-precision, high-recall* quadrant. We report the mean and standard deviation for each response variable for each quadrant, as well as R^2_{adj}, F-value, and p-value of the model. As can be seen from the table, the means for the quadrants are statistically significantly different for four of our six response variables: the final precision, and the changes in precision, recall, and F2-measure. We illustrate the differences in the means for final TM precision and recall for each quadrant and the differences in changes in precision and recall in Figure 3(a) and Figure 3(b). Changes in precision and recall are illustrated as a single vector $(mean(\Delta\text{Rec}), mean(\Delta\text{Prec}))$ plotted from the center of each quadrant.

Observed independent variables (Q2). For the second question, we wanted to see how our observed independent variables (Table IV) related to the response variables. For each observed independent variable, to prevent systematic bias and reduce error variance within groups, we controlled

[4]We used significance level $\alpha = 0.05$, bolded items are statistically significant

Table VI
INFLUENCE OF INITIAL PRECISION AND INITIAL RECALL ON RESPONSE VARIABLES (DEGREES OF FREEDOM: 2, 81)

Response Variable	R^2_{adj}	F-value	Sig. (pval)
FinPrec	0.120	6.659	**0.002**
FinRec	-0.004	0.842	0.434
FinF2	0.0	1.012	0.368
ΔPrec	0.454	35.548	**0.0001**
ΔRec	0.444	34.115	**0.0001**
ΔF2	0.288	17.761	**0.0001**

Table VII
INFLUENCE OF INITIAL F2-MEASURE ON RESPONSE VARIABLES (DEGREES OF FREEDOM: 1, 82)

Response Variable	R^2_{adj}	F-value	Sig. (pval)
FinPrec	0.056	5.913	**0.017**
FinRec	0.037	3.117	0.081
FinF2	0.053	4.604	**0.035**
ΔPrec	0.036	3.02	0.086
ΔRec	0.312	37.227	**0.0001**
ΔF2	0.238	25.672	**0.0001**

Table VIII
INFLUENCE OF STARTING QUADRANT ON RESPONSE VARIABLES (DEGREES OF FREEDOM: 3, 80).

		QI	QII	$QIII$	QIV	Statistics
	N	10	26	14	34	
FinPrec	$\bar{x}$	64.46	52.94	61.03	72.96	$R^2_{adj} = 0.138$
	s	18.2	20.88	22.89	16.43	$F = 5.434$
						p= 0.002
FinRec	$\bar{x}$	64.58	60.90	52.68	64.34	$R^2_{adj} = 0.004$
	s	18.14	21.96	29.4	16.42	$F = 1.113$
						p= 0.349
FinF2	$\bar{x}$	64.27	57.71	51.08	65.09	$R^2_{adj} = 0.038$
	s	16.00	19.40	25.79	16.62	$F = 2.083$
						p= 0.109
ΔPrec	$\bar{x}$	38.14	21.03	−14.53	−2.49	$R^2_{adj} = 0.402$
	s	20.24	17.49	27.83	18.85	$F = 19.586$
						p= 0.0001
ΔRec	$\bar{x}$	33.75	−11.06	23.81	−7.35	$R^2_{adj} = 0.341$
	s	18.78	24.91	30.42	18.44	$F = 15.344$
						p= 0.0001
ΔF2	$\bar{x}$	35.5	6.32	11.1	−6.97	$R^2_{adj} = 0.253$
	s	17.64	21.52	32.09	17.56	$F = 10.356$
						p= 0.0001

for two baseline independent variables: initial precision and initial recall. That is, we statistically adjusted the dependent variable means to what they would have been if all groups had started out with equal distribution of initial precision and recall.

Of the eleven observed independent variables, only *time to complete the tracing task* (Time) is continuous. We used multiple linear regression analysis for it. The remaining 10 variables are categorical; we used one-way ANCOVA to analyze them. Table IX shows the results of the analyses. For each model, we report the R^2_{adj}, the F-value, and the p-value. We also report the baseline R^2_{adj} value from Table VI for each response variable's effect with initial precision and initial recall. As can be seen from the table, **only one** observed independent variable, ValidEff, has statistically significant effect on any of our response variables.

When performing tracing tasks, participants spent their time engaging in two different types of activities: vetting candidate links from the initial TM, or searching the artifacts for missing links. Variable ValidEff quantifies the amount of effort participants put into vetting candidate links from the initial TM. This information was collected in the post-experiment survey on a 0 – 5 scale, where 0 meant *"never performed this type of activity"* and 5 meant *"performed this type of activity for every single link."* When looking at the performance of participants based on the value of ValidEff variable, the key reason for the statistically significant influence on final recall and change in recall can be seen from Table X. Of 84 participants, 62 specified values of 0, 1, 2, or 3 in response to the post-experiment question. Thirteen participants gave a response of 4 and one participant gave a response of 5[5]. As can be seen from Table X, the average recall for those whose response was 4 or 5 is 20.5% less than the average recall of those whose responded 0—3. We also noted that those who responded 4 or 5 were the only group of participants whose mean change in recall was negative: an overwhelming -24.22%. In Figure 3(c), we plot the performance of the participants who gave responses of 4 or 5. As can be seen from the graph, the majority of participants received initial TMs with relatively high recall and varying precision, and most of them wound up significantly reducing recall. This behavior is consistent with the self-reported effort spent on validating candidate links: participants did almost nothing but link validation, but they wound up making many incorrect judgment calls, which lead to many true links being rejected.

Comparing the influences (Q3). Based on the analyses shown above, we conclude that the accuracy of the initial TM in our experiments was the *best predictor* for the *change in the TM accuracy*. Initial precision and initial recall jointly account for over 40% of variability of each of ΔPrec, ΔRec, and ΔF2 response variables. In fact, even the much coarser, Starting quadrant of the initial TM accounts for 33%—39% variability for these response variables. Of the 11 observed independent variables in our study (see Table IX), only ValidEff had statistically significant effect on ΔRec and ΔF2, explaining an additional 7–8% of variability – much less than our baseline variables.

As can be seen from Figure2(d), the majority of final TMs submitted by the study participants have precision and recall between 50% and 70%. Our study found that except for ValidEff, the effort spent validating candidate links, no other independent variable (baseline or observed) had significant effect of the final TM recall. In fact, ValidEff itself shows

[5]The remaining participants did not provide an answer.

Table IX
ANALYSIS FOR OBSERVED INDEPENDENT VARIABLES CONTROLLING FOR INITIAL TM PRECISION AND RECALL

Response		Location	Procedure	SEExp	TRExp	Time	Grade	TrConf	Opinion	MissingEff	ValidEff	Prepared
FinPrec	R^2_{adj}	0.12	0.109	0.107	0.121	0.127	0.148	0.083	0.129	0.049	0.116	0.102
$R^2_{adj} = 0.12$	F	1.012	0.510	0.876	2.034	1.025	1.668	0.045	1.111	0.091	1.02	1.003
	p	0.318	0.602	0.421	0.158	0.315	0.166	0.833	0.335	0.965	0.413	0.423
FinRec	R^2_{adj}	0.006	0.001	-0.001	0.002	-0.012	0.017	-0.022	-0.017	-0.016	**0.115**	-0.053
$R^2_{adj} = -0.004$	F	1.789	1.18	1.028	1.306	0.126	1.423	0.001	0.373	0.847	**2.810**	0.34
	p	0.185	0.313	0.362	0.257	0.724	0.234	0.978	0.690	0.522	**0.023**	0.887
FinF2	R^2_{adj}	-0.006	0.01	0.019	0.016	-0.008	-0.025	-0.022	0.0	-0.024	**0.153**	-0.022
$R^2_{adj} = 0.0$	F	0.496	1.383	1.765	0.2.284	0.013	0.503	0.077	0.784	0.717	**3.428**	0.741
	p	0.483	0.257	0.178	0.135	0.910	0.734	0.782	0.46	0.613	**0.008**	0.595
ΔPrec	R^2_{adj}	0.461	0.448	0.466	0.475	0.475	0.472	0.460	0.462	0.427	0.465	0.459
$R^2_{adj} = 0.454$	F	1.012	0.510	0.876	2.034	1.025	1.668	0.045	1.111	0.191	1.02	1.003
	p	0.318	0.602	0.421	0.158	0.315	0.166	0.833	0.335	0.965	0.413	0.423
ΔRec	R^2_{adj}	0.449	0.446	0.443	0.445	0.416	0.455	0.445	0.413	0.444	**0.493**	0.424
$R^2_{adj} = 0.444$	F	1.789	1.18	1.028	1.306	0.126	1.423	0.001	0.373	0.847	**2.810**	0.34
	p	0.185	0.313	0.362	0.257	0.724	0.234	0.978	0.69	0.522	**0.023**	0.887
ΔF2	R^2_{adj}	0.284	0.297	0.322	0.297	0.243	0.270	0.291	0.245	0.265	**0.326**	0.268
$R^2_{adj} = 0.288$	F	0.541	1.565	2.53	1.17	0.021	0.521	0.001	0.571	0.598	**2.582**	0.653
	p	0.464	0.216	0.086	0.283	0.885	0.721	0.98	0.568	0.702	**0.034**	0.66

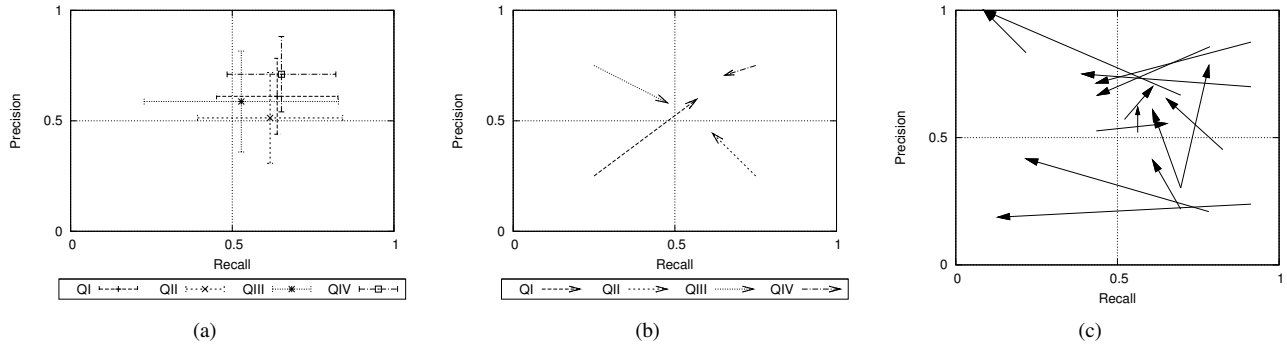

Figure 3. Graphs showing: (a) means and standard devisions of final recall and precision by Initial Quadrant; (b) mean changes in final recall and precision by Initial Quadrant, and (c) performance of participants who spent much effort validating candidate links (values "4" or "5" for variable ValidEff).

statistically significant difference only between those who put all their effort into link validation (and rejected many true links) and those who did not. For final precision, initial TM accuracy provided some predictive power, accounting for about 12% of the variability.

C. Discussion

Two of the observed independent variables, Procedure and Location, represent where and how participants took the study. As can be seen from Table IX, neither variable has a statistically significant effect on the response variables. That is, *participants in both locations* and *in all three experiments* (RETRO, manual, RETRO.net) performed in roughly the same way when controlled by the initial TM accuracy. This means that the results we observed *were repeatable in our studies* between two locations and between three procedures used for tracing.

A number of observed variables assess "personal qualities" of study participants: software engineering experience, prior tracing experience, grade level, confidence level, preparedness level, and opinion on whether manual tracing is better than tracing with a software tool. As can be seen from Table IX, **none** of these variables have statistically significant effect on any response variables. This means that in our experiments, the final TM accuracy was not affected in any major way by the prior experiences of the participants or by their opinions. This is an interesting observation: in general, one expects more experienced analysts to perform better on various tasks than those with less experience. In our experiments, this did not happen.

Returning to the questions of interest, based on these studies, the answers are:

Q1. Yes. The effect of the accuracy of the initial TM on the accuracy of the final TM, and especially on the change in the accuracy is statistically significant.

Q2. Of all the examined variables, **only one**, *self-reported effort validating offered links*, was in statistical significance

with four of our response variables.

Q3. The variables measuring accuracy of the initial TM have a higher effect on the *change in the TM accuracy* than any observed independent variable. The most interesting observed result is that **low** initial TM accuracy lead to the best overall improvement in accuracy.

This result (Q3) begs the question "why?" It might seem intuitive that starting with a low initial quality TM provides ample opportunities for improvement – removing incorrect links and finding missing links. It should be noted that these "mistakes" in the TM are not necessarily so easy to detect. Follow–on work to this study has shown that many participants incorrectly confirmed false links (often the same problematic links) as well as incorrectly added links to the TM [16]. Though our investigation into "why" is very preliminary, it appears that all participants had periods of work where many correct decisions were made in a row: the difference in participants was how long it took them to get to that "constructive" period of work and how long that period lasted. This clearly could be tied to the data set, though data captured with our logging tool indicated that many participants did not work in a sequential order (rather, they "jumped around" in the dataset). Further study must be undertaken with additional datasets in order to understand "why" low intial TM accuracy leads to the best overall improvement in accuracy.

Initial TM accuracy had statistically significant, although weaker and only partial, effect (on final precision but not on the final recall) on the accuracy of the final TM. We observe that the lack of significant effect on the final recall is chiefly due to the fact that the majority of final TMs had recall in the 50%–70% range. The only significant interaction with final recall came from the 14 participants who reported spending much of effort on link validation: they were the only group with a significantly lower recall.

V. Conclusions and Future Work

Initial examination of data from the Cuddeback et al. study [6] led us to observe that: (a) participants failed to recover the true TM, (b) participants given lower accuracy TMs tended to show more significant improvement, and (c) regardless of starting TM accuracy and size, participants tended to guess the size of the true TM. This was a surprising finding that led to 12 months of continued experimental studies as well as statistical analysis to understand why. This paper presents a look at 11 independent variables which may account for the change in final TM accuracy. Interestingly enough, statistical analyses show that analyst's tracing experience, amount of effort applied to look for missing links, comfort level with tracing, etc. do not affect final TM accuracy. Rather, the initial TM accuracy is the most important factor impacting final TM accuracy. The only other factor that had a statistically singificant interaction with final TM accuracy was the amount of time an analyst spent vetting links provided by the tool.

Table X
Influence of ValidEff on response variables

Response		0–3	4–5
	N	62	14
FinRec	$\bar{x}$	65.25	**44.72**
	s	19.98	23.28
FinF2	$\bar{x}$	62.36	**45.66**
	s	17.98	22.25
ΔRec	$\bar{x}$	5.18	$\mathbf{-24.22}$
	s	26.71	30.73
ΔF2	$\bar{x}$	7.06	$\mathbf{-14.55}$
	s	22.33	23.59

In the introductory example, NDFC lacks tracing processes that could assist with their three looming issues. If they select a fully manual process, errors and analyst discontent will surely ensue. If a totally automated solution is selected, a large number of false positive links in the TM could lead to dismissal of the tool as faulty. Assisted tracing, an analyst working with the results of an automated tool, suits their needs the best. In applying such a process, NDFC would probably like to know how to select analysts for the job (years of software engineering experience, years of tracing experience, comfort level with the tool, etc.). Imagine their surprise to learn that the only statistically significant factor that impacts the quality of the final TM in assisted tracing is the initial quality (which has negative correlation) and the amount of time spent vetting links. The analyst's experience, effort applied, etc. do not matter.

Our key, formally confirmed finding that lower initial TM accuracy leads to better analyst performance *significantly alters* our overall approach to assisted tracing. We can no longer rely on the automated tracing methods to produce high-accuracy results and expect these results to translate into even higher-accuracy ones in assisted tracing settings. While we still consider the quest for high-precision, high-recall automated tracing methods important, we must acknowledge that it will not provide a panacea for assisted tracing. We have established that analysts performing assisted tracing tasks are fallible and predictably so. Assisted tracing procedures must account for this. As such, we plan to run a follow-on experiment using data from a real project to further understand this behavior.

VI. Acknowledgments

Our work is funded in part by a grant from Lockheed Martin and in part by the National Science Foundation under grant CCF-0811140. We thank John Dalbey for the ChangeStyle dataset, and David Janzen, Clark Turner, and Gene Fisher for allowing us to conduct the studies in their courses.

References

[1] The Sarbanes-Oxley act 2002. http://www.soxlaw.com/.

[2] G. Antoniol, G. Canfora, G. Casazza, A. De Lucia, and E. Merlo. Recovering Traceability Links Between Code and Documentation. *IEEE Transactions on Software Engineering*, 28(10):970–983, Oct 2002.

[3] J. Cleland-Huang, W. Marrero, and B. Berenbach. Goal-Centric Traceability: Using Virtual Plumblines to Maintain Critical Systemic Qualities. *IEEE Transactions on Software Engineering*, 34(5):685–699, Sep–Oct 2008.

[4] J. Cleland-Huang, R. Settimi, C. Duan, and X. Zou. Utilizing supporting evidence to improve dynamic requirements traceability. In *Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on*, pages 135 – 144, aug.-2 sept. 2005.

[5] D. Cuddeback. Automated requirements traceability: the study of human analysts. *Master's Thesis and Project Reports*, May 2010.

[6] D. Cuddeback, A. Dekhtyar, and J. Hayes. Automated Requirements Traceability: The Study of Human Analysts. In *Requirements Engineering Conference (RE), 2010 18th IEEE International*, pages 231–240, Sydney, Australia, Oct 2010. IEEE.

[7] A. Egyed, F. Graf, and P. Grunbacher. Effort and quality of recovering requirements-to-code traces: Two exploratory experiments. In *Proceedings of the 18th International Conference on Requirements Engineering, 2010.*, pages 94–101. IEEE, Sept. 2010.

[8] O. Gotel and C. Finkelstein. An Analysis of the Requirements Traceability Problem. In *Proceedings of the First International Conference on Requirements Engineering, 1994*, pages 94–101. IEEE, Apr 1994.

[9] M. Grechanik, K. S. McKinley, and D. E. Perry. Recovering and using use-case-diagram-to-source-code traceability links. In *Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, ESEC-FSE '07, pages 95–104, New York, NY, USA, 2007. ACM.

[10] J. Hayes, A. Dekhtyar, and J. Osborne. Improving requirements tracing via information retrieval. In *Proceedings of the 11th IEEE International Requirements Engineering Conference, 2003.*, pages 138–147. IEEE, Sept. 2003.

[11] J. Hayes, A. Dekhtyar, and S. Sundaram. Advancing Candidate Link Generation for Requirements Tracing: the Study of Methods. *IEEE Transactions on Software Engineering*, 32(1):4–19, Jan 2006.

[12] J. Hayes, A. Dekhtyar, S. Sundaram, E. Holbrook, S. Vadlamudi, and A. April. REquirements TRacing on target (RETRO): improving software maintenance through traceability recovery. *Innovations in Systems and Software Engineering*, 3(3):193–202, Sept. 2007.

[13] J. H. Hayes and A. Dekhtyar. Humans in the Traceability Loop: Can't Live With 'Em, Can't Live Without 'Em. In *TEFSE '05: Proceedings of the 3rd International Workshop on Traceability in Emerging Forms of Software Engineering*, pages 20–23, New York, NY, USA, 2005. ACM.

[14] J. H. Hayes, A. Dekhtyar, A. Holbrook, S. Sundaram, and O. Dekhtyar. Will Johnny/Joanie make a good software engineer: are course grades showing the whole picture? *International Conference on Software Engineering Education and Training, Conference (CSEET)*, pages 175–184, 2006.

[15] J. H. Hayes, A. Dekhtyar, and S. Sundaram. Text Mining for Software Engineering: How Analyst Feedback Impacts Final Results. In *MSR '05: Proceedings of the 2005 International Workshop on Mining Software Repositories*, pages 1–5, New York, NY, USA, 2005. ACM.

[16] W.-K. Kong, J. Hayes, A. Dekhtyar, and J. Holden. How do we trace requirements? an initial study of analyst behavior in trace validation tasks. In *CHASE '11: Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering*, Honolulu, HI, USA, 2011.

[17] B. R. M. Host and C. Wohlin. Using students as subjects - a comparative study of students and professionals in lead-time impact assessment. *Empirical Software Engineering: An International Journal*, 5(3):201–214, 2000.

[18] A. Mahmoud and N. Niu. Source code indexing. In *TEFSE 2011: 6th International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE'11)*, Honolulu, HI, USA, 2011.

[19] 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, 2003*, pages 125–135. IEEE, May 2003.

[20] P. Mdder, O. Gotel, T. Kuschke, and I. Philippow. tracemaintainer - automated traceability maintenance. In *International Requirements Engineering, 2008. RE '08. 16th IEEE*, pages 329 –330, 2008.

[21] R. Oliveto, M. Gethers, D. Poshyvanyk, and A. D. Lucia. On the equivalence of information retrieval methods for automated traceability link recovery. In *Proc. of 18th IEEE International Conference on Program Comprehension (ICPC'10)*, pages 68–71, June 2010.

[22] I. Omoronyia, G. Sindre, M. Roper, J. Ferguson, and M. Wood. Use case to source code traceability: The developer navigation view point. In *Proceedings of the 2009 17th IEEE International Requirements Engineering Conference, RE*, RE '09, pages 237–242, Washington, DC, USA, 2009. IEEE Computer Society.

[23] T. Savage, B. Dit, Gethers, and D. Poshyvanyk. Topicxp: Exploring topics in source code using latent dirichlet allocation. In *Proc., 26th IEEE International Conference on Software Maintenance (ICSM'10)*, September 2010. Formal Research Tool Demonstration.

[24] W. F. Tichy and F. Padberg. Empirical methods in software engineering research. In *Companion to the proceedings of the 29th International Conference on Software Engineering*, ICSE COMPANION '07, pages 163–164, Washington, DC, USA, 2007. IEEE Computer Society.