Judgment-Updating among Software Professionals

Magne Jørgensen
Stein Grimstad
Simula Research Laboratory, Norway

Abstract **Initial judgments related to key decisions in software projects are often based on one-sided or misleading information. The initial assessment of the benefits of introducing a new development tool may for example be based a vendor's sales demonstration or a reference client's favorable description. In this paper we study software professionals' abilities to adjust their early, biased judgments when receiving contradicting or less biased information. The first study, involving 160 software professionals, found a strong under-adjustment for the impact of misleading information and one-sided argument. A follow-up two weeks later found that this under-adjustment was not removed over time. The second study, involving 65 software professionals, found that the ability to update biased judgments may sometimes be quite good, but that it is hard to predict when. A practical consequence of our results is that software professionals should strongly emphasize the avoidance of biased and potentially misleading information and not trust that they are able to adjust their judgments and beliefs when more reliable and unbiased information are available.**

I. INTRODUCTION

Numerous studies illustrate how easily people are impacted by biased and misleading information, and, that proper adjustments when more valid information is available can be problematic [1]. This article examines to what extent these problems are present in software engineering contexts. Software engineering specific results on this topic may, for example, lead to greater awareness among software engineering researchers and professionals about the problems of biased and misleading information. Greater awareness may in turn lead to the development of better software engineering judgment and decisions processes.

As an illustration of the relevance of this topic, assume that you are supposed to select between different software development tools. Your initial judgment of a particular tool is based on the presentation of one vendor's one-sided, potentially strongly biased arguments in favor of his own tool. When presented alternative tools, or independent tool reviews, you may typically find that the first vendor's information had a much lower reliability than you initially believed or you may even assess the first information to be totally misleading. When updating your judgments about the first vendor's tool, will you tend to under-adjust, properly adjust or over-adjust? This and related questions are investigated in this paper.

The main research question of this paper is the following: *How well are software professionals able to update judgments when first exposed to misleading information or one-sided arguments?*

One reason why we find this research question interesting, is that we have previously demonstrated that irrelevant and misleading information can have a large impact judgments related to software cost estimates, see for example [2] and [3]. We see no reason that these results should not extend to other software engineering situations. We have also experienced that studies on how to avoid judgmental biases in software engineering decision and judgment processes are rare and seldom based on empirical data enabling isolation of effects [4].

Sections 2 and 3 briefly describe the design and results of two studies addressing the research question. Section 4 discusses study limitations and ethical concerns related to the design of the studies. Section 5 concludes.

II. STUDY 1

A. Participants

Study 1 was conducted at a large developer conference in Oslo, Norway (JavaZone 2007). The participants consisted of 160 software professionals attending a seminar presented by the first author of this paper.

B. Study Design

The overall purposes of Study 1 were to better understand: i) When and how much misleading information and one-sided arguments impacted the software professionals' judgments and ii) To what degree the software professionals were able to adjust properly when they were told that the information was incorrect and misleading.

The software engineering related judgment situation we selected was described as follows [translated from Norwegian]:

There is a large difference in how risk-seeking programmers are. Some programmers frequently try new ways of programming, while others stick to what they know best and know will work. Assume that we define a risk-seeking programmer as one who agrees in the statement: "I like to find own, innovative ways of solve problems" and that a programmer is better than another if she/he develops software with similar quality (measured as number of errors and perceived maintainability) more efficiently.

All situations taken into consideration, which of the statements below to you think is most correct? (Select only one.)

1) The risk-seeking always perform better

2) The risk-seeking almost always perform better

3) The risk-seeking perform better most of the time

4) The risk-seeking perform better in slightly more than half of the situations

5) The risk-seeking perform better in about half of the situations

6) The risk-seeking perform worse in slightly more than half of the situations

7) The risk-seeking perform worse most of the time

8) The risk-seeking almost always perform worse

9) The risk-seeking always perform worse

The main motivation behind the selection of the performance of risk-seeking vs. risk-averse programmers as judgment task was that we believed that the software professionals had relevant experience about this topic, but at the same time not very strong opinions. We consequently believed it would be possible to impact their judgments through study results, but that it would also be meaningful to make judgments based on own experience.

The software professionals were randomly divided into seven groups (Group A-G). There were between 19 and 23 participants in each group. Group A was the control group, i.e., their judgments were not influenced by one-sided or misleading information. The treatment elements were as follows:

T1a: Receive the information that "*A recent Canadian study showed that risk-willing programmers performed better*".

T1b: Receive the information that "*A recent Canadian study showed that risk-averse programmers performed better*".

T2a: Provide one argument in favor of that risk-willing programmers perform better.

T2b: Provide one argument in favor of that risk-averse programmers perform better.

T3: Receive the information that "*The Canadian study was invented to see how much this information impacted your judgment.*"

The study was divided into three phases. Phase 1 and 2 were completed in sequence with no other tasks in-between, while Phase 3 was conducted two weeks later, per email. See Table I for an overview of the treatments per group and phase. "J" indicates a judgment about risk-willing vs. risk-averse on the previously described 9-point scale.

TABLE I
DESIGN OF STUDY 1

Group	Phase 1	Phase 2	Phase 3
A	J (control group)	-	-
B	T1a, T2a, then J	T3, then J	J
C	T1b, T2b, then J	T3, then J	J
D	T1a, then J	T3, then J	J
E	T1b, then J	T3, then J	J
F	T2a, then J	T1a, then J	-
G	T2b, then J	T1b, then J	-

The study design enabled several analyses on judgment impacts, including:

- The combined effect of a misleading study and a one-sided argument (comparison of Groups B and C in Phase 1).
- The effect of a misleading study (comparison of Groups D and E in Phase 1).
- The effect of a one-sided argument (comparison Groups F and G, Phase 1).
- The effect of a misleading study after a one-sided argument and judgment (comparison Group F and G, Phase 2).
- The ability to properly update from the impact of a misleading study and a one-sided argument (comparison of Groups B and C, Phase 2).
- The ability to properly update from the impact of a misleading study (comparison of Groups D and E, Phase 2)

When the Phases 1 and 2 of the study was completed, all participants were debriefed about the purpose of the study, how the different types of treatments had impacted their judgments and that the Canadian study was not real (Groups B, C, D, E did already know this).

C. Study Results

There are measurement theoretical challenges with the use of mean values to aggregate judgments on our 9-point scale. It is, for example, discussable whether we can assume an interval or even an ordinal scale. For this reason, we also conducted tabulated statistics. We found, however, that an analysis of tabulated statistics made no difference in main conclusions, and we therefore decided to present the aggregated results as mean values for communication purposes. The statistical hypothesis tests of differences in mean values should however be interpreted carefully.

First, we examined the size of the impact from the different treatments, see Table II.

TABLE II
TREATMENT EFFECTS

Group – Phase – Treatment	Mean
A: Control group.	5,0
B - Phase 1 - Misleading study, then one-sided argument. Both in favor of risk-willing programmers.	3,6
C – Phase 1 - Misleading study, then one-sided argument. Both in favor or risk-averse programmers.	5,6
D – Phase 1 - Misleading study in favor of risk-willing programmers.	4,1
E – Phase 1 - Misleading study in favor of risk-averse programmers.	5,5
F – Phase 1 - One-sided argument in favor of risk-willing programmers	4,6
G – Phase 1 - One-sided argument in favor of risk-averse programmers.	5,1
F – Phase 2 - Initial judgment in-between one-sided argument and study in favor of risk-willing programmers.	4,4
G – Phase 2 - Initial judgment in-between one-sided argument and study in favor of risk-averse programmers.	5,1

The mean value of the Group A responses represents the unbiased responses. This means that mean values lower than 5.0 ("The risk-seeking is performing better about half of the situations") suggest an impact in the direction of that the risk-seeking programmers are better, and, values higher than 5.0 the opposite. The fact that an unbiased response, on average, corresponded to the exact mid-value of the scale may be an advantage when analyzing the results. It suggests, for example, that if not impacted by our treatment, the software professionals would be quite undecided about the programmer effect of risk-willing vs. risk-averse.

Examining Table II we find the following ranked size of impact of the treatments (measured as the difference between the mean values of treatments differing only in the risk-willing vs. risk-averse dimension):

- Misleading study + one-sided argument: 5,6 – 3,6 = 2,0
- Misleading study alone: 5,5 – 4,1 = 1,4
- Initial judgment in-between one-sided argument and misleading study: 5,1 – 4,4 = 0,7
- One-sided argument alone: 5,1 – 4,6 = 0,5

All differences in mean values are significant with $p<0,1$ (one-sided t-test, assuming unequal variance).

Interestingly, the software professionals seemed to be easier to impact when the misleading study and the one-sided argument were in favor of the risk-seeking programmers. This illustrates that there are many factors, not easy to predict, that impact the judgmental biases. Another interesting observation is that the misleading study had less impact when presented after producing a judgment, i.e., Groups F and G in Phase 2 were less impacted than Groups B and C in Phase 1 even if they were exposed to exactly the same information (see Table I). The resistance towards judgment updating, consequently, increased when forced to make a judgment before exposed to new information. Used properly, this increase in belief updating resistance can be an important means to reduce the impact of biased and misleading information. In situations with information of high validity, however, it may rather be a threat and explicit judgments in advance of exposure to that information should probably be avoided.

It is rational to be impacted by a scientific study, particularly when not possessing extensive expertise in a topic, which may have been the case in our study. The impact from the misleading study was therefore no surprise. The main issue was, however, to what degree the software professionals were able to adjust for the impact of the misleading Canadian study, when informed that it was not real but invented for the purpose of impacting their judgment. Table III shows the judgments following this information about the Canadian study. The corresponding values for the Phase 1 judgments are in brackets.

TABLE III
ADJUSTMENT WHEN INFORMED ABOUT THE VALIDITY OF THE STUDY

Group	Mean
B – Phase 2 (Phase 1: Misleading study, then one-sided argument. Both in favor of risk-willing programmers.)	4,0 (3,6)
C – Phase 2 (Phase 1: Misleading study, then one-sided argument. Both in favor of risk-averse programmers.)	5,1 (5,6)
D – Phase 2 (Phase 1: Misleading study in favor of risk-willing programmers.)	4,2 (4,1)
E – Phase 2 (Phase 1: Misleading study in favor of risk-averse programmers.)	4,9 (5,5)

The results in Table III show, not surprisingly, that the effect of the Canadian study decreased when receiving information about the information unreliability. The results show, however, also that the remaining effect of the Canadian study is still substantial. The differences after the update were as follows:

- Misleading study + one-sided argument, then debriefing: 5,1 – 4,0 = 1,1 (previously 2,0)
- Misleading study, then debriefing: 4,9 – 4,2 = 0,7 (previously 1,4)

All differences in mean values are significant with $p<0,1$ (one-sided t-test, assuming unequal variance).

Interestingly, those initially believing in the better performance of the risk-averse (Groups C and E) now gave answers similar to those provided by the control group (Group A), i.e., in the non-impacted situation. The main remaining effects in Phase 2 were consequently related to insufficient adjustment of those misled to believe in the risk-willing programmers. We are currently not able to explain this difference well. Perhaps the perceived benefits from risk-willingness are more positively loaded and easier to stay convinced about.

There are several possible direct and contributing reasons for the observed inability to adjust sufficiently for the influence from the misleading information. A contributing reason is that the judgments were based on partly unconscious mental processes, i.e., the judgments were probably based on "what feels right" rather than an analytical, explicit strategy combining previous beliefs and the results of the Canadian study. Use of unconscious judgment processes means that it is difficult to assess the size of the impact of the misleading information and, consequently, difficult to "roll back" to the initial belief and understanding. We need, however, additional elements to explain the systematic tendency towards under-adjustment. Candidate explanations include the "cognitive dissonance" theory [5], the "comprehension as accepting" theory [6], and the "primacy effect" theory [7]:

- Cognitive dissonance: Software professionals, as far as we have experienced, like to see themselves as rational individuals. Rational individuals should clearly not be strongly impacted by misleading or one-sided information. To preserve a picture of themselves as rational individuals, i.e., to avoid cognitive dissonance,

it may consequently be hard to accept that their responses were strongly impacted by a singly study with misleading information or their own one-sided argument.

- Comprehension as accepting: Cowley [6] suggest that people when comprehending information, even when accepting that the information is of low validity, start with an acceptance of the information as an unavoidable part of their comprehension process, and then try to "unaccept" it. The "unaccepting" process is, however, typically not able to completely re-adjust, which may explain, for example, the documented positive effect of obviously exaggerated advertisement claims.
- Primacy effect: The primacy effect describes the situation where the starting point of a decision or judgment process has a disproportionate effect on its outcome, perhaps caused by an unconscious desire to support the initial decision or judgment. This is, for example, reflected in the long-lasting effect of the "first impression" when meeting people.

As described earlier, two weeks after the first part of this study, we emailed the participants in Groups B, C, D and E (those impacted most) and asked them to make the judgment for the third time. Unfortunately, only 19 (of 83 in those four groups) responded. The responses may nevertheless give us useful information about the robustness of the impact, since we can track the changes of each individual. For analysis simplicity reasons we joined the responding participants into two groups:

- Group BD: Those who received treatments in the direction of that risk-willing programmers were better , i.e., Groups B and D. (n=6)
- Group CE: Those who received treatment in direction of that risk-willing programmers were worse, i.e., Groups C and E. (n=13)

Table IV displays the judgments provided in Phase 1, Phase 2, and Phase 3 (two weeks after the de-briefing). All mean values are based on the judgments on those 19 who responded in Phase 3, only.

TABLE IV
JUDGMENTS, PHASE 1, 2 AND 3 (TWO WEEKS LATER)

Group	Mean Phase 1	Mean Phase 2	Mean Phase 3
BD-Phase 1	3,3	3,5	3,5
CE-Phase 1	5,4	5,0	4,9

The differences in mean values of BD and CE in Phase 3 are still significant with $p<0.1$ (one-sided t-test, assuming unequal variance). Table IV consequently suggests that the impact from the misleading study and/or one-sided argument did not go away or decreased very much in that two week period. This is in accordance with results from other domains. Results from other domains show also that the effect sometimes even *increase* over time due to the so-called "sleeper effect", see [8] for a review. The "sleeper effect" describes a situation where the impact from low credibility sources may increase over time, because the validity of the source and the information itself get more and more disconnected over time.

III. STUDY 2

A. Participants

Study 2 was conducted at a seminar on software development effort estimation at Simula Research Laboratory, Oslo, Norway organized by the authors of this paper. There were 65 software professionals participating.

B. Study Design

The purpose of this study was to evaluate the impact of the information presentation sequence. Unlike the previous study, all information presented in study was of high validity, i.e., it was based on scientific results presented in highly ranked journals. The judgment requested was to select statement that they most agreed with:

1) I believe that software developers have a very strong tendency towards optimistic memory of own work effort.

2) I believe that software developers have a strong tendency towards optimistic memory of own work effort.

3) I believe that software developers have a weak tendency towards optimistic memory of own work effort.

4) I believe that software developers neither have tendency towards optimistic or pessimistic memory of own work effort.

5) I believe that software developers have a weak tendency towards pessimistic memory of own work effort.

6) I believe that software developers have a strong tendency towards pessimistic memory of own work effort.

7) I believe that software developers have a very strong tendency towards pessimistic memory of own work effort.

The treatment elements were as follows:

T1: Receive information about a study (with full reference) reporting the finding that people have a tendency to believe that tasks completed earlier took less time than they actually took (optimistic memory). The information includes a possible explanation for the finding.

T2: Receive information about a study (with full reference) reporting the finding that people have a tendency to believe that tasks completed earlier took more time than they actually took (pessimistic memory). The information includes a possible explanation for the finding.

The participants were randomly divided into three groups of similar size (Groups A-C). One of these groups was randomly split into two sub-groups (C1 and C2). The number of participants in C1 and C2 is consequently only the half of that in Groups A and B.

Table V shows the design of Study 2. As before, "J" means that a judgment was made. Tx + Ty means that both texts were presented at the same time, but that Tx was located before Ty in the text. As opposed to Study 1, the participants completed another, unrelated, task between Phase 1 and 2.

TABLE V
DESIGN OF STUDY 2

Group	Phase 1	Phase 2
A	T1, then J	T2, then J
B	T2, then J	T1, then J
C1	T1 + T2, then J	-
C2	T2 + T1, then J	-

If there was an information presentation sequence effect we would expect to find different judgments of Groups A and B in Phase 2, or, of Groups C1 and C2 in Phase 1.

C. Study Results

The judgments by the participants in Group A and B were, as expected, impacted by the one-sided study information. The mean of the Group A participants' judgments was 2,5, while that of the Group B participants was 3,4. The difference in mean values for these two groups were significant with $p<0.1$ (one-sided, t-test assuming unequal variance). This effect was present in spite of the extensive experience the participants had with software development projects and that the software professionals were stimulated to include their own experience, i.e., that we told them: "*The described study is conducted on other tasks than software development, and the results are not necessarily transferable.*"

Our main research question was to what extent the software professionals were able to adjust their judgment when presented with the second study, i.e., when they knew about both studies. Table VI provides the mean judgments of the different groups, with Phase 1 judgments of Group A and B in brackets.

TABLE VI
JUDGMENTS BASED ON BOTH STUDIES

Group	Mean
A – Phase 2	2,8 (2,5)
B – Phase 2	2,9 (3,4)
C1	2,4
C2	2,9

Table VI suggest that the software professionals in this case were able to adjust sufficiently when exposed to the other study and had about the same average judgment in Phase 2 (2,8 vs. 2,9). While this was somewhat surprising in light of the results of Study 1, the difference in judgments between the software professionals in Group C1 and C2 was even more surprising (a one-sided t-test on difference in mean values, assuming unequal difference gave p=0,15). Table VI suggests that the difference is caused by those believing in an optimistic memory. The participants in Group C1 (who read the study in favor of an optimistic memory before that in favor of a pessimistic memory) had about the same response as Group A in Phase 1 (who received only the information about the study in favor of an optimistic memory).

Assuming that our results point at real underlying differences in judgments, the task is consequently to explain why presenting the study in favor of optimistic memory had no impact when presented in Phase 2 with another unrelated task in-between, but had an significant effect when presented before with the study reporting the opposite result. Currently, we find it difficult to explain this difference and need more studies to better understand the reasons behind it. The surprising findings illustrate the problems we may have in predicting sequence effects on judgments.

An essential difference between Studies 1 and 2 is that the information in Study 1 was misleading, while that in Study 2 was one-sided, but still valid. If the main explanation for the finding in Study 1 is related to the "cognitive dissonance" theory, we should expect lower effects in Study 2, as observed. Understandably, it may be easier to admit, and still keep an image of one-self as rational, am impacted of valid rather than invalid information.

IV. DISCUSSION

A. Limitations

There are many types of software engineering situations where people are exposed to misleading information, one-sided arguments and potential information sequence impacts. To what extent our two studies represent and can be generalized to real-life software development situations is not obvious. In particular, real-life situations may be more decision-oriented and with stronger personal involvement than the judgment-oriented, rather artificial situation described in our experiment. While we asked about their general beliefs related to a software development relevant relationship, the software professionals' actual beliefs may be better exposed in real-life decision processes. To evaluate the robustness of our results we consequently need more studies on real-life judgment and decision processes and how they are impacted by how information is presented. The unconsciousness of many software engineering judgment processes ("what feels right" rather than an explicit analysis), however, means that it may be just as difficult to defend against the reported impacts in real-life situations. For that reason, we find it likely that the reported effects are present in real-life situations. The main limitation of our study is, we believe, not related to the existence of the reported effects, but to the effect size in real-life situations.

B. Ethical concerns

In both studies we may have permanently impacted the study participants beliefs related to the performance of risk-willing vs. risk-averse programmers, and, software developers' tendency towards optimistic and pessimistic memory. We believe that our study nevertheless is ethically defendable of the following reasons:

- The beliefs about these issues are unlikely to have practical, harmful consequences for the participants.

- We de-briefed the participants about the validity of the misleading information, about the purpose of the study and the study results. The study results were integrated into software engineering seminars where they were used to illustrate how easily our opinions are impacted. The study participants probably improved their awareness of such impacts. Hopefully, this will increase the participants' abilities to make unbiased decision in the future. Our study can consequently be seen as a useful learning process for the participants.

We have received a few comments on the study by the participants. All responses have been positive, e.g., "I learned a lot", and none of them have been negative. This supports our belief that the study has an acceptable ethical standard. We acknowledge, however, that this type of study easily can be ethically problematic and should be conducted with great care to avoid harmful, impact that cannot be defended by the benefits of participating in the study.

V. CONCLUSION

The findings presented in this study support results from other domains on the vulnerability of many judgment updating processes and show their relevance in software engineering domains. We show that software professionals' judgments can be permanently distorted by one-sided, self-generated arguments and information that later is shown to have low or no validity. We also found that belief-updating when presented with new information of high validity in some cases were adequate, but that the information presentation sequence may still be essential. This indicates how difficult it is to predict when we are able to properly update judgment and when not.

We believe that our findings have several practical consequences for learning, judgment and decisions processes in software engineering. In particular, we believe that our results provide strong arguments in support of more structured software engineering judgment and decision processes. Processes ensuring the avoidance of low quality, biased information may be particularly important. The process we outline in [9], describing Evidence-Based Software Engineering, may be useful for that purpose. That process focus on collecting high quality information, evaluate the argumentation of studies and experience-based practice properly, and use structured means to summarize the evidence. Even such processes may be subject to unconscious impact from irrelevant and biased information, but probably less than ad-hoc, unstructured judgmental processes.

VI. REFERENCES

1. Wilson, T.D., D.B. Centerbar, and N. Brekke, Mental contamination and the debiasing problem, in Heuristics and biases: The psychology of intuitive judgment, T. Gilovich, D. Griffin, and D. Kahneman, Editors. 2002, Cambridge University Press: Cambridge. p. 185-200.
2. Jørgensen, M. Individual Differences in How Much People are Affected by Irrelevant and Misleading Information. Proceedings of Second European Conference on Cognitive Science. 2007. Delphi, Greece: Hellenic Cognitive Science Society: p. 347-352.
3. Jørgensen, M., A review of studies on expert estimation of software development effort. Journal of Systems and Software, 2004. 70(1-2): p. 37-60.
4. Jørgensen, M. and M. Shepperd, A systematic review of software cost estimation studies. IEEE Transactions on software engineering, 2007. 33(1): p. 33-53.
5. Festinger, L., A Theory of Cognitive Dissonance. 1957, Evanston, IL: Row, Peterson & Company.
6. Cowley, E., Processing exaggerated advertising claims. Journal of Business Research, 2006. 59: p. 728-734.
7. Bond, S.D., et al., Information distortion in the evaluation of a single option. Organizational Behavior and Human Decision Processes, 2007. 102: p. 240-254.
8. Kumkale, G.T. and D. Albarracı, The Sleeper Effect in Persuasion: A Meta-Analytic Review. Psychological Bulletin, 2004. 130(1): p. 143-172.
9. Dybå, T., B. Kitchenham, and M. Jørgensen, Evidence-based Software Engineering for Practitioners. IEEE Software, 2005. 22(1): p. 58-65.