Interview Review: an Empirical Study on Detecting Ambiguities in Requirements Elicitation Interviews

Paola Spoletini[1], Alessio Ferrari[2], Muneera Bano[3,4], Didar Zowghi[4] and Stefania Gnesi[2]

[1] Kennesaw State University, GA, USA, Email: pspoleti@kennesaw.edu
[2] CNR-ISTI, Pisa, Italy,
Email: {alessio.ferrari,stefania.gnesi}@isti.cnr.it
[3] Swinburne University of Technology, Melbourne, Australia, Email: mbano@swin.edu.au
[4] University of Technology Sydney, Australia,
Email: {muneera.bano, didar.zowghi}@uts.edu.au

Abstract. [Context and Motivation] Ambiguities identified during requirements elicitation interviews can be used by the requirements analyst as triggers for additional questions and, consequently, for disclosing further – possibly *tacit* – knowledge. Therefore, every unidentified ambiguity may be a missed opportunity to collect additional information. **[Question/problem]** Ambiguities are not always easy to recognize, especially during highly interactive activities such as requirements elicitation interviews. Moreover, since different persons can perceive ambiguous situations differently, the unique perspective of the analyst in the interview might not be enough to identify all ambiguities. **[Principal idea/results]** To maximize the number of ambiguities recognized in interviews, this paper proposes a protocol to conduct reviews of requirements elicitation interviews. In the proposed protocol, the interviews are audio recorded and the recordings are inspected by both the analyst who performed the interview and another reviewer. The idea is to use the identified cases of ambiguity to create questions for the follow-up interviews. Our empirical evaluation of this protocol involves 42 students from Kennesaw State University and University of Technology Sydney. The study shows that, during the review, the analyst and the other reviewer identify 68% of the total number of ambiguities discovered, while 32% were identified during the interviews. Furthermore, the ambiguities identified by analysts and other reviewers during the review significantly differ from each other. **[Contribution]** Our results indicate that interview reviews allow the identification of a considerable number of undetected ambiguities, and can potentially be highly beneficial to discover unexpressed information in future interviews.

Keywords: requirements elicitation, interviews, ambiguities, tacit knowledge, reviews.

1 Introduction

Requirements elicitation interviews are often used as starting point of the requirements elicitation process [1–4]. Interviews are often perceived by students and novice analysts

as an easy tool to use, but they can be affected by several factors, that can prevent the analyst to elicit all the relevant knowledge – including *tacit knowledge* [5] – during the elicitation process. Tacit knowledge is system relevant information that remains unexpressed often because it belongs to the unconscious level of processing of the customer or is too difficult to be properly described, and it therefore remains undocumented. Techniques were developed to facilitate the disclosure of tacit knowledge [6–9]. However, its detection is still an open problem in requirements engineering [6], and specific techniques are required to elicit it.

In our previous work [7], we have highlighted the relationship between ambiguity and tacit knowledge in requirements elicitation interviews. More precisely, we have shown that, differently from what happens in written requirements where ambiguity is a threat to the quality of requirements, ambiguity could be a powerful tool in oral synchronous communication. Indeed, when an ambiguity is detected in the words of a customer *during* an interview, the analyst asks additional follow-up questions that may lead to the identification of unexpressed, system-relevant aspects [10]. Unfortunately, given the highly interactive nature of requirements elicitation interviews, it is not always easy to recognize ambiguous statements during the interview, that are likely to be identified in second hearing of the interview.

This observation suggests conducting reviews at requirements elicitation interview process. Such a proposal would be a step forward in addressing the challenge highlighted by Salger: *"Software requirements are based on flawed 'upstream' requirements and reviews on requirements specifications are thus in vain"* [11]. Indeed, currently reviews of software process artifacts do not include any artifact before requirements documents [12]. Even if reviews are considered an effective practice to improve the quality of products [13–16], and the benefits of requirements reviews have been highlighted by several studies, especially for what concerns the identification of defects in requirements specifications [17, 14, 18], challenges remain for their widespread application [11, 19].

For these reasons, we propose to add a review of the recording of the elicitation interviews. In our proposal, we include two types of reviews: one performed by the analyst, to give her the possibility to more carefully listen to the interview, and a second one conducted by another analyst, called *reviewer*, who will analyze the interview from an additional perspective. The rationale behind the proposal is that ambiguities in the words of a customer can be perceived in different ways by different analysts, as has already been observed for ambiguities in written requirements [20, 21]. In the proposed method, the analyst performs the interview with the customer, and audio records the dialogue. The recording is then reviewed by the analyst and an external reviewer, who annotate the identified ambiguities, together with the fragment of conversation that generated it, and list the questions that they would have asked in the interview to disambiguate the annotated situation. The questions are used for further clarifications in future interactions with the customer. In [22], we have explored the feasibility and the benefits of this idea through an exploratory study that gave encouraging results. In this paper we aim at clearly defining the review protocol and assess its effectiveness through a controlled experiment performed with two independent groups of students from University of Technology Sydney (UTS) and Kennesaw State University (KSU).

The remainder of the paper is structured as follows. In Section 2.3, we summarize related works concerning ambiguity in RE with particular focus on their classification in oral communication, and review techniques, including a brief description of the result from our exploratory study. In Section 3, the controlled experiment is presented together with the developed review protocol. Sections 4 and 5 present the results of the controlled experiment and a discussion on its limitations. In Section 6, we provide final remarks and we describe the next planned step in our research.

2 Background

This section provides background information on topics relevant to our study. More precisely, Sections 2.1 and 2.2 describe the related work on ambiguities in RE in general, and in interviews in particular. Section 2.2 describes the existing work on reviews in requirements engineering, and, finally, Section 2.4 briefly presents our work on interview reviews, including encouraging results from an exploratory study.

2.1 Ambiguities in Requirements

The problem of ambiguity in RE has been widely studied over the years, with particular focus on written requirements. The existing work can be roughly separated into two groups: strategies to *prevent* ambiguities, and approaches to *detect* ambiguities in (already written) requirements.

The first set of approaches can be divided into two categories: strategies which rely on formal approaches [23–25], and strategies based on constrained natural languages [26–28]. Looking into the first sub-category, the works of Kof [23] promotes ambiguity prevention by transforming requirements into formal/semiformal models, which are easier to analyze and constrain. The approaches implemented by tools like Circe-Cico [24] and LOLITA [25] also follow a similar rationale. The second subcategory is focused on the use of constrained natural languages, which should limit the possibility of introducing ambiguity and is also easier to be analyzed. Examples of well known constrained formats for editing requirements are EARS [26] and the Rupp's template [27]. Arora *et al.* [28] defined an approach to check the conformance of requirements to these templates.

Other approaches aim to *detect* ambiguities in requirements. Most of these works stem from the typically defective terms and constructions classified in the ambiguity handbook of Berry *et al.* [29]. Based on these studies, tools such as QuARS [30], SREE [31] and the tool of Gleich *et al.* [32] were developed. More recently, industrial applications of these approaches were studied by Femmer *et al.* [33] and by Rosadini *et al.* [20]. As shown also in these studies, rule-based approaches tend to produce a high number of false positive cases – i.e., linguistic ambiguities that have one single reading in practice. Hence, *statistical* approaches were proposed by Chantree *et al.* [34] and Yang *et al.* [35], to reduce the number of false positive cases, referred to as *innocuous ambiguities*.

All these works, with the exception of Chantree *et al.* [34] and Yang *et al.* [35], focus on the *objective* facet of ambiguity, assuming that the majority of the ambiguities could

be identified by focusing on a set of typically dangerous expressions. In [10, 7], we observed that this is not the most common case in requirements elicitation interviews, in which the *subjective* and contextual facets become dominant.

2.2 Ambiguity in Interviews

Differently from the ambiguity in written documents, the term ambiguity in interviews (i.e., synchronous oral communication), covers a larger set of situations. Indeed, an ambiguity can occur not only because the words used by the speaker are meaningless for the listener or are combined in a difficult to interpret structure, but also because the information delivered by the speaker is in contrast with the knowledge that the listener already built. Other ambiguities can be generated by the fact that new information acquired in a conversation can change the knowledge on a previously acquired concept. In particular, it is possible to identify the following categories of ambiguities in requirements elicitation interviews [10]:

- *interpretation unclarity*: The fragment of the speaker's speech cannot be understood;
- *acceptance unclarity*: The fragment uttered by the speaker is understandable and there is no reason to doubt that what can be understood from it matches with the intended meaning of the customer. However, the fragment appears *incomplete* to the listener or it has some form of *inconsistency* with what previously understood, or previous knowledge of the listener.
- *multiple understanding*: multiple interpretations of the fragment uttered by the speaker are possible, and each interpretation makes sense to the listener.
- *detected incorrect disambiguation* : previously the listener perceived an acceptance unclarity, and, later in the interview, she understands that the given interpretation was not correct (i.e., it did not match with the intended meaning of the speaker).
- *undetected incorrect disambiguation*: the listener did not perceive an acceptance unclarity, but, at a certain point of the interview, she understands that her interpretation of a certain fragment of the speaker was not correct.

Notice that since during a conversation the originator of a misunderstanding situation is present, the listener – the analyst in our case – can follow up with additional questions, which not only allows for disambiguating the situation, but also for finding additional knowledge that can be relevant for the analyst.

2.3 Requirements Review

IEEE Std 1028-2008 [12] defines the standards for the review of software products and categorizes them in five types: management reviews, technical reviews, inspections, walk-throughs and audits. In our work, we focus on *inspections*, which are *systematic peer-examinations that [...] verify that the software product exhibits specified quality attributes [...]* and *collect software engineering data*. Katasonov and Sakkinen [36] provide a categorization for reading techniques to be applied in inspection reviews, distinguishing between ad-hoc, checklist-based, defect-based, perspective-based, scenario-based and pattern-based. The technique proposed in our work is *defect-based*, since it focuses on a particular type of defect, namely ambiguity.

Inspections have been already successfully used in RE. In particular, Fagan [17] and Shull *et al.* [14] provide early and successful techniques for requirements inspection. A survey on the topic was published by Arum *et al.* [37]. More recent works on requirements review are those by Salger [11] and by Femmer *et al.* [19], which focuses on the *challenges* that requirements review faces in practice. The list of challenges include aspects such as the long time required for its implementation [19] and the need to have more effective elicitation techniques [11]. This latter goal is pursued by Karras *et al.* [38], who developed a tool for video inspection of requirements workshops. Notice that the majority of related work on requirements reviews focuses on reviews applied to specifications, while our goal is to analyze the audio recording of interviews. Our work differs also from that of Karras *et al.* [38], since we suggest to analyze only the audio recording of interviews, and we focus on ambiguity, a communication defect that is not considered by this previous study.

2.4 Interview Review: an Exploratory Study

The idea of moving the review at the level of requirements elicitation interviews to detect ambiguities was first presented in [22] together with our research plan and an exploratory study. The goal was understanding whether the idea that different ambiguities may emerge when an interview is listened by different subjects is actually grounded.

Our exploratory study used a preliminary version of the review method, and had two expert analysts applying it on a set of 10 *unstructured* interviews [4] performed by KSU undergraduate students. The reviewers were a researcher in requirements elicitation, and a professional analyst, respectively. The two reviewers were required to independently listen to the recording of each interview and to report ambiguous situations in a spreadsheet. They were requested to identify situations that they thought the analysts found ambiguous and situations that they found ambiguous but were not followed up by the analyst. The initial results showed not only that the reviews are very helpful in detecting ambiguities – the reviewers together found 46% that were not detected during the interview –, but also that the review process can benefit from the perspectives of different reviewers.

3 Experiment Design

The goal of our research is to analyze if reviewing requirements elicitation interviews allows the identification of additional ambiguities that were not identified during the interview by the requirements analyst. To investigate this problem in a systematic way, we set the following research questions:

RQ1: Is there a difference between ambiguities explicitly revealed by an analyst during an interview, and ambiguities identified by the analyst or by a reviewer when listening to the interview recording?

RQ2: Is there a difference between ambiguities identified by the analyst when listening to the interview recording, and ambiguities identified by a reviewer who listens to the interview recording?

RQ1 aims at exploring the contribution of the review phase in terms of ambiguities, considering the case in which the analyst performs the review and the case in which an external reviewer performs it. RQ2 focuses on the different contributions that the analyst, who performed the interview, and an external reviewer, who listens to the interview for the first time during the review, can give in the review phase. To answer these questions, we perform an experiment in which the same interview recording is reviewed by the analyst, and by an external reviewer. To provide the information to answer the questions, during the review the analyst explicitly distinguishes between ambiguities previously identified during the interview, and ambiguities found when listening. More details are given in Sect. 3.4.

3.1 Variables and Hypotheses

Variables In our study, the *independent* variable is the *perspective*, which is a combination of the *role* of the person who is working in identifying ambiguities, i.e., the analyst or an external reviewer, and the *moment* in which the identification occurs, i.e., "during the interview" or "during the review". The perspective can assume four values: analyst in the interview (AI); reviewer in the review (RR); analyst in the review (AR). Notice that the perspective value "reviewer in the interview" (RI) is not applicable, since the reviewer does not participate to the interview.

The *dependent* variables are the *performance* in identifying ambiguities ($perf$, in the following) of the three identified perspectives. The performance of the generic perspective X (with $X \in \{AI, AR, RR\}$) is measured as the combination of the description and the numbers of ambiguities identified by X. To formally define $perf_X$, we introduce the following sets:

- a_{AI}: the set of ambiguities explicitly detected by the analyst during the interview;
- a_{AR}: the set of ambiguities detected by the analyst during the review;
- a_{RR}: the set of ambiguities detected by the reviewer during the review.

So, the performance of a generic perspective X (with $X \in \{AI, AR, RR\}$) is characterized by the content and the cardinality of the correspondent a_X, i.e., $perf_X = \langle a_X, |a_X| \rangle$.

Hypotheses From **RQ1** we have derived two different null hypotheses:

H1.1$_0$: The reviewer's performance during the review is irrelevant with respect to the analyst's performance during the interview;

H1.2$_0$: The analyst's performance during the review is irrelevant with respect to the analyst's performance during the interview.

In H1.1$_0$, the perspective can assume the values AI and RR. In the light of these variables, H1.1$_0$ can be defined as $\mu_{|a_{RR} - a_{AI}|} = 0$, i.e., the mean of the number of ambiguities found in the review by the reviewer (RR) which were not found in the interview by the analyst (AI) is 0. Informally, if H1.1$_0$ cannot be rejected, it means that the ambiguities found in the review by the reviewer (RR), which were not found in the interview by the analyst (AI), were found by chance.

In H1.2_0, the perspective can assume the values AI and AR. Analogously, formalizing H1.2_0 can be defined as $\mu_{|a_{AR}-a_{AI}|} = 0$ i.e., the mean of the number of ambiguities found in the review by the analyst (AR) which were not found in the interview by the analyst (AI) is 0. Informally, if H1.2_0 cannot be rejected, it means that the ambiguities found in the review by the analyst (AR) which were not found in the interview by the analyst (AI) were found by chance.

From **RQ2**, we derive the following null hypothesis: **H2$_0$:** The reviewer's performance during the review and the analyst's performance during the review are equivalent. The independent variable assumes the values AR and RR. The dependent variable is still the performance in identifying ambiguities and can be measured in terms of found ambiguities. Notice that saying that the performance are equivalent means that the two sets of identified ambiguities are about the same not just in terms of cardinality, but also in terms of content. This hypothesis would be very difficult to analyze, so it can be reformulated in the following sub-hypotheses:

H2.1$_0$: The analyst's performance during the review is irrelevant with respect to the reviewer's performance during the review;

H2.2$_0$: The reviewer's performance during the review is irrelevant with respect to the analyst's performance during the review.

Indeed, if both the reviews are irrelevant one with respect to the other, the two reviews are equivalent.

So, H2.1_0 is formalized as $\mu_{|a_{AR}-a_{RR}|} = 0$, i.e., the additional ambiguities found by the analyst in the review (AR) with respect to those found by the reviewer during the review (RR) were found by chance. H2.2_0 is formalized as $\mu_{|a_{RR}-a_{AR}-a_{AI}|} = 0$, i.e., the additional ambiguities found by the reviewer in the review (RR) with respect to those found by the analyst during the review (AR) without considering the ones already found in the interview (AI) were found by chance. Note that in H2.2_0 we have to explicitly exclude the ambiguities found by the AI perspective: if the reviewer founds an ambiguity that was already found by the analyst during the interview, this is not taken into account in the computation. In H2.1_0 this is not needed, since a_{AR} and a_{AI} are disjoint sets.

In order to analyze the stated hypotheses, we designed and conducted a controlled experimental study which will be described in the remainder of this section.

3.2 Participants

Our controlled experiment was performed with two equivalent independent groups of participants, namely students of KSU and students of UTS. It consists of two phases: in the first phase participants performed a set of role-play requirements elicitation interviews, and in the second phase, participants reviewed the interviews. In the following we will describe the participants from both institutions and the main characteristic of the protocol. The complete protocol is available at `https://goo.gl/PI2LLy`.

The first group of participants consists of 30 students of KSU. The recruited students belonged to a User-Centered Design course, composed of undergraduate students of the 3rd and 4th year with major related to a computing discipline (software engineering, computer science, information technology, and computer game development and

design). The students were provided with a two hours lecture on requirements elicitation interviews delivered by the 1st author, in which they received an introduction on different types of interviews and general guidelines on how to conduct each of the main types. The class used a reference book [39] and additional lecture notes. While the participation to the study was on a voluntary basis, students who participated were assessed and received additional marks for their final results.

The second group of participants consists of 12 students of UTS. They were Master of Information Technology students, a two years full time postgraduate degree[5], and almost all of them were in their 1st year. The students belonged to the Enterprise Business Requirements course. To prepare for the experiment, the students attended an introductory lecture on requirements elicitation that included how to run interviews, delivered by the 4th author, and were advised to take a `Lynda.com` course online on requirements elicitation interviews. Students participated in this activity as volunteers and were not assessed for it.

3.3 Interviews

In both locations, the students were divided into 2 groups, namely analysts and customers. The creation of the two groups and the association between customers and analysts were performed randomly. One week before the interview was planned, customers were told: "Take a week to think about a mobile app for smart-phones you would like to have developed. You have a $ 30,000 budget and your idea should be feasible within your budget. If the ideas you have seems not doable with this budget look at the apps you have on your phone and try to think how you would like to modify one of them."

For both the participants groups, the interviews took place simultaneously at the reference institution, and the time slot allocated was 30 minutes in addition to the time required for setting up the experiment. The interviews were recorded at KSU in Fall 2016 and at UTS in Spring 2017. Before starting the interviews both the customers and the analysts were required to fill out a demographic questionnaires, one specific for the analyst and one specific for the customer, with the goal of knowing the proficiency of the participant with the language used in the interview (in both institution, English) and their previous experience in the role they were acting.

The students conducted *unstructured* interviews [4], which is the most suitable approach in this context. Indeed, in the experiment, the students analysts are exploring ideas for new products for which they have no background information. The interviews were audio recorded.

In order to help the students to focus, the analysts were given the goal of collecting an initial list of requirements after the interview was performed. The requirements had to be listed in the form of user stories, detailed enough to estimate the required amount of work in terms of needed time and number of developers.

[5] A full description of the degree can found at `http://www.handbook.uts.edu.au/courses/c04295.html`.

3.4 Reviews

After the interviews the participants were requested to work on the review of the interviews with the following rationale. Each student who acted as customer was requested to review an interview performed by another group. The interview to review was assigned to the customer randomly when the groups were created. Instead, analysts were requested to review the interview they conducted. This allows for two reviews: one internal, performed by the same analyst who performed the interview, one external, performed by a reviewer, who did not know anything about the interview and the product described in it before the review.

The main steps of the review protocol the reviewers were assigned are as follows:

1. Create a spreadsheet with the columns: *Time*, *Fragment*, *Question*.
2. Start the reproduction of the audio recording, start a timer, and start listening. If any external factor interrupt your work, please stop the timer and restart it when you resume your review.
3. Stop the audio when you perceive an ambiguity in the words of the customer.
4. Whenever you stop the audio for the listed cases, add a line to the spreadsheet with the following content:
   - **Time:** the moment in which the customer produces the fragment;
   - **Fragment:** the fragment of speech that triggered the ambiguity;
   - **Question:** the question that you would ask to the customer to clarify.
5. When you have finished listening, stop the timer and annotate the time that passed from the beginning of your activity. This will serve to estimate the time that you employed to perform the whole activity.

As guidelines to identify the ambiguities, participants were suggested the following: "As a rule of thumb, stop the reproduction in any case in which, if you were the analyst, you would have asked the customer one or more questions of the form:"

- *What does it mean [...]?* (You have not understood the meaning of what you heard)
- *What is the purpose of [...]?* (You have not understood the purpose of what you heard)
- *Can you discuss in more detail about [...]?* (What you heard is too general)
- *You mentioned that [...], but [...]?* (What you heard contradicts what you heard before, or your vision of the problem)
- *Do you mean <A> or <B>?* (What you heard can mean different things)
- *I thought that with [...] you meant [...], was I wrong?* (You have doubts about a previous understanding of some concept)

This review protocol allows the identification of ambiguities perceived by the reviewer (perspective RR, see Sect. 3.1). The review protocol is slightly different for the analysts, since they had to annotate their own interview, distinguishing between ambiguities perceived during the interview and ambiguities perceived during the review of the recording of the interview. In particular, steps 3 and 4 were modified as follows:

4. Stop the recording whenever the customer says something that is unclear, ambiguous or does not make sense to you. As a rule of thumb, stop the recording in any of the following two cases:

  - you asked a clarification question to the customer during the interview;
  - a new question comes to your mind now, and you regret not to have asked the question to the customer during the interview.
5. Whenever you stop listening, add a row to the spreadsheet, and write: fragment, time, question, and moment ("I" if the question was asked during the interview and "L" if the question came to your mind during the review).

In this way, the review of the analyst allowed the identification of the moments that she perceived as ambiguous within the interview (perspective AI) and the detection of additional ambiguities during the review (perspective AR).

4 Evaluation

To evaluate the results of this study and answer to our research questions, we analyzed the spreadsheets of the the analysts and of the reviewers, and we created a_{AI}, a_{AR}, and a_{RR}. From these sets, we derived other relevant sets that will be used in the following analyses:

- $both_{AI,RR} = a_{AI} \cap a_{RR}$: the set of detected ambiguities in common between the analyst during the interview and the reviewer;
- $both_{AR,RR} = a_{AR} \cap a_{RR}$: the set of detected ambiguities in common between the analyst during the review and the reviewer;
- $ao_{AI} = a_{AI} - both_{AI,RR}$: the set of ambiguities detected only by the analyst during the interview. Notice that $both_{AI,AR}$ is not considered since it is empty by construction;
- $ao_{AR} = a_{AR} - both_{AR,RR}$: the set of ambiguities detected only by the analyst during the review (again $both_{AI,AR}$ is not considered since it is empty by construction);
- $ao_{RR} = a_{RR} - both_{AI,RR} - both_{AR,RR}$: the set of ambiguities detected only by the reviewer during the review.

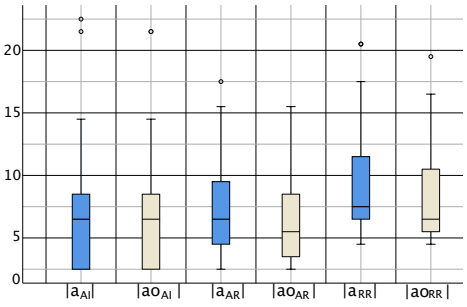
The sum of the cardinalities of these sets forms the total number of ambiguities identified in the whole process. In the following, the data of KSU and UTS are combined together. At the end of this section, we will briefly discuss them separately.

Overall Evaluation In order to have an initial idea of the performance of each perspective, we have computed the classic descriptive statistics (minimum, maximum, mean, and median) for the number of ambiguities found by each perspective and for the number of ambiguities found only by a perspective. These values and the corresponding box plots are reported in Figure 1. It is worth noting that each perspective contributes to the identification of ambiguities by identifying on average at least 4 ambiguities that were not found by any other perspective (Fig. 1, for each ao_X the Mean value is above 4).

To look at the distribution of the detected ambiguities on the different combinations of roles and situations, we can refer to Figure 2a. The figure considers the following cases of detection: only during the interview ($|ao_{AI}|$), only during the review performed by the analyst ($|ao_{AR}|$), only during the review performed by the reviewer ($|ao_{RR}|$),

	Min	Max	Median	Mean	Std dev
$\|a_{AI}\|$	0	20	4	5.1905	5.91286
$\|ao_{AI}\|$	0	19	4	4.8095	5.80189
$\|a_{AR}\|$	0	15	4	4.9524	4.58777
$\|ao_{AR}\|$	0	13	3	4.1905	3.88097
$\|a_{RR}\|$	2	18	5	7.619	5.04456
$\|ao_{RR}\|$	2	17	4	6	4.58258

(a)

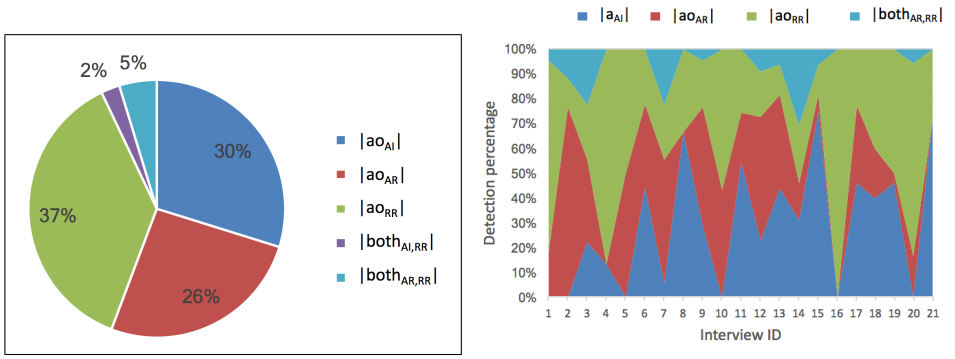


(b)

Fig. 1: Descriptive statistics and box plots for the main metrics of the performance

common to the interview and the review performed by the reviewer ($|both_{AI,RR}|$), and common to the reviews ($|both_{AR,RR}|$). These numbers are evaluated with respect to the total number of ambiguities, which is the sum of all these contributions. The number of ambiguities detected *only* during the interview – blue area, ($|ao_{AI}|$) – is 30%, and increases only to 32% if we consider also the ones that were also detected in the review of the reviewer ($|both_{AI,RR}|$) – purple area. Hence, the overall review activity identified 68% of the total number of ambiguities. Analogously, Figure 2b shows the distribution of the detection of ambiguities for the performed interviews separately. Analyzing the data from the figure, we can observe that in most of the cases the majority of ambiguities are detected during the reviews – red, green and light blue areas – rather than during the interview – blue area. Specifically, it is possible to observe that in more than 75% of the cases the ambiguities detected during the interview ($|a_{AI}|$) are less than 50% of the total number of detected ambiguities – i.e., the blue area plot is below 50% for 75% of the interviews. Moreover, in 50% of the cases this percentage drops below 30%. These data are an interesting result *per se*, because they highlight that there is a considerable number of ambiguities that is not identified during the interview and can be detected with a further analysis. Indeed, regardless of the subject who performs the review process – either the analyst or reviewer –, this analysis suggests that the review is useful to spot a significant number of ambiguities not identified during the interview.

RQ1: Contribution of the Review Activity To answer **RQ1**, we look into the contribution of the review activity in detecting ambiguities with respect to the ones identified by the analyst during the interview. Looking at Figure 2a, we see that the percentage of ambiguities that were common between the analyst (during the interview), and the reviewer is only 2% ($|both_{AI,RR}|$, purple area) of the total number of ambiguities identified in the whole process. It is also possible to notice that the reviewers contribute by identifying on average 37% ($|ao_{RR}|$, green area) of the total number of ambiguities. Looking only at the ambiguities detected by the analyst during the interview and by the analysts in the review ($|a_{AI}| + |a_{AR}|$ – notice that has pointed out at the beginning of Section 4 there is no overlapping between a_{AI} and a_{AR}), the contribution of the analyst's review in detecting ambiguities ($|a_{AR}|$) is on average more than 49% (not shown in the figures). Analogously, looking only at the ambiguities detected by the analyst



(a) Average Detection Distribution

(b) Detection Distribution ($|a_{AI}| = |ao_{AI}| + |both_{AI,RR}|$)

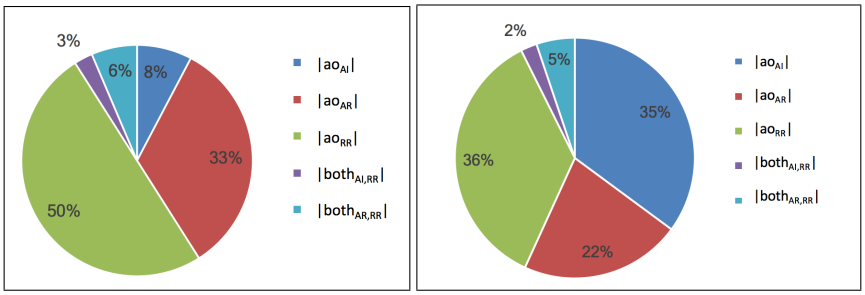
Fig. 2: Distribution of ambiguities

during the interview and by the reviewer in the review ($|a_{AI}| + |a_{RR}| - |both_{AI,RR}|$), the contribution of the reviewer in detecting ambiguities ($|a_{RR}| - |both_{AI,RR}|$) is on average more than 56% (not shown in the figures). Among all the ambiguities detected by the reviewers only 4.45% ($|both_{AI,RR}|$, not shown) were identified also by the analysts during the interview. Notice that the reviewer's work always positive contributed to the detection of ambiguities. Indeed, in all the interviews the reviewer detected at least a couple of additional ambiguities with respect to those detected during the interview.

To more precisely answer to RQ1, we evaluate $H1.1_0$ and $H1.2_0$ by using the (student) paired t-test, which provides an hypothesis test of the difference between populations for pair of samples whose differences are approximately normally distributed. $H1.1_0$ is formalized as $\mu_{|a_{RR}-a_{AI}|} = 0$, where $|a_{RR} - a_{AI}$ is $|a_{RR}| - |both_{AI,RR}|$, and $H1.2_0$ is formalized as $\mu_{|a_{AR}-a_{AI}|} = 0$, where $|a_{AR} - a_{AI}|$ is $|a_{AR}| - |both_{AI,AR}| = |a_{AR}|$. The paired t-test is applicable in these cases since both $|a_{RR}| - |both_{AI,RR}|$ and $|a_{AR}|$ are normally distributed with a skewness of .958 (standard error = 0.501) and kurtosis of 0.01 (standard error = 0.972) and a skewness of 1.088 (standard error = 0.501) and kurtosis of -0.032 (standard error = 0.972), respectively. In both cases it is possible to reject the null hypotheses with significance level 5% since t_0 is greater than the tabular reference value. Indeed, we have 21 samples, which correspond to 20 degrees of freedom and a tabulated reference value $t_{0.025,20} = 2.086$, and, $S_d = 8.9944$ and $t_0 = 3.6877$ for $|a_{RR}| - |both_{AI,RR}|$ and $S_d = 5.0883$ and $t_0 = 6.5187$ for $|a_{AR}|$.

RQ2: Contribution of Different Reviews To answer **RQ2**, we compare the ambiguities detected during the reviews performed by the analysts with those detected by the reviewers. Considering the ambiguities that were common between the analyst during the review and the reviewer, we have that these amount solely to 5% ($|both_{AR,RR}|$, light blue area in Figure 2a) of the total number of ambiguities. On average the ambiguities that are common to both reviews is 7.14% (not shown in the figures) of the total number of ambiguities detected in the review phase ($|a_{AR}| + |a_{RR}| - |both_{AR,RR}|$). Furthermore, Figure 2b shows that the set of ambiguities detected in both the reviews always contains less than 30% of the total number of detected ambiguities (the light blue area plot is always above 70%).

Analogously to what done for RQ1, to answer to RQ2, we evaluate H2.1_0 and H2.2_0 by using the (student) paired t-test. H1.1_0 is formalized as $\mu_{|a_{AR}-a_{RR}|} = 0$, where $|a_{AR}-a_{RR}|$ is $|ao_{AR}|$, and H2.2_0 is formalized as $\mu_{|a_{RR}-a_{AR}-a_{AI}|} = 0$, where $|a_{RR}-a_{AR}-a_{AI}|$ is $|ao_{RR}|$. Both $|ao_{AR}|$ and $|ao_{RR}|$ are normally distributed with a skewness of .902 (standard error = 0.501) and kurtosis of 0.01 (standard error = 0.971) and a skewness of 1.14 (standard error = 0.501) and kurtosis of 0.2 (standard error = 0.971), respectively. In both cases it is possible to reject the null hypotheses with significance level 5% since t_0 is greater than the tabular reference value. Indeed, we have 21 samples, which correspond to 20 degrees of freedom and a tabulated reference value $t_{0.025,20} = 2.086$, and, $S_d = 5.269$ and $t_0 = 5.4968$ for $|ao_{AR}|$ and $S_d = 3.881$ and $t_0 = 4.8288$ for $|ao_{RR}|$.



(a) UTS Average Detection Distribution (b) KSU Average Detection Distribution

Fig. 3: Comparing UTS and KSU experiments

KSU vs UTS Data If we separate the data of UTS (Figure 3a) and KSU (Figure 3b), we can notice that while both cases suggest that there is a benefit in both the review performed by the analysts and the one performed by the external reviewers, there is a considerable discrepancy in the percentage of ambiguities detected only in the interview (8% in the case of UTS, 35% in the case of KSU – blue areas in the figures). This discrepancy might be caused by the fact that KSU students received a different training, with a higher focus on ambiguity, with respect to UTS students, and were therefore more focused on ambiguity detection already during the interview. However, this result does not change the validity of the above performed analysis, which focuses on the data regarding the *common* cases of ambiguity, which, on average, do not substantially vary among the two groups.

Another aspect that is relevant to our study and needs to be evaluated is the *time* employed by the reviewers for their task, with respect to the duration of the interviews. Unfortunately, the data collected by the students, especially the KSU ones, are incomplete. The 45% and the 18% of the data regarding the review time of analysts and reviewers, respectively, are missing. However, from the data collected, we observe that on average the reviews take about twice the time needed for the interviews. This is a reasonable time for an activity which contributes considerably to the detection of ambiguities.

5 Threats to Validity

In this section, we list the main threats to the validity of our study. Notice that this controlled study has been developed to overcome the limitations of the exploratory study presented in Section 2.4 and was designed preventing most of the problems of that experiment.

Internal Validity The students participating in the experiments had slightly different backgrounds. In particular, UTS students were graduate students, while KSU students were undergraduate students. Even if their learning experience on requirements elicitation was similar, being at a different degree level could influence the attitude of the students towards the learning process. However, we argue that the fact that KSU students were mostly 3rd and 4th year students and they were evaluated, while the graduate students were not, may have mitigated this maturation threat. Furthermore, since UTS students were in the first semester of their first year of their degree, they can be considered nearly graduate. As collected in the survey that was distributed before the experiments, we noticed that a few of the students already experienced being part of an elicitation interview while others did not. This can represent an history threat. However, the participants with experience had in general a very limited experience, which classifies them all as unexperienced analysts, equivalent with respect to our experiment.

Construct Validity We argue that there are no construct validity threats in our study. Indeed, our research questions (and consequently our hypotheses) maps very straightforwardly to the collected data: the questions are related to the number of detected ambiguities and we evaluated them directly using this measure, which represent the performance of the perspectives.

External Validity The population validity is the major threat in this study, since we use students instead of practitioners to perform our interviews. Although according to Höst *et al.* [40] students with a good knowledge of computer science appear to perform as well as professionals, there is always a difference between the industrial world, and a role-playing settings. This limit will be addressed by our next research step with will be discussed in Section 6.

6 Conclusion and Future Work

In our previous work [22], we proposed to define a review method for requirements elicitation interviews, with the goal of identifying ambiguities in the conversation. Indeed, identified ambiguous situations can be used to suggest further clarifying questions, that can help in finding additional relevant (possibly *tacit*) knowledge. In this paper we presented a protocol to apply interview reviews in practice and a controlled experiment to evaluate the effectiveness of the protocol. The protocol consists in having both the analyst and an external reviewer to review performed interviews. The method aims to exploit both a more reflective attitude of the analyst during the review phase with respect to the interview phase, and the different perspective of the external analyst. Our experiment involved 42 students in two Higher Education Institutions, KSU and UTS, and measured the contribution of the reviews in detecting ambiguities. The experiment showed that reviews help to detect a considerable number of additional ambiguities and both the reviews were helping in different ways, suggesting the needs of both of them.

As a future work we aim to prove the correlation between the questions generated by detected ambiguities and the quality of the information that they allow to find. In particular, we want to address the following research question: Can the ambiguities identified during interview review be used to ask *useful* questions in future interviews? To answer to it, we plan to perform a case study in industry, in which the method will be applied, and the impact of the questions will be monitored along the development. The idea is to gather qualitative data about the *perceived* usefulness of the questions produced after the first interview, and their *actual* usefulness observable after the delivery of the products. It is worth mentioning that our approach can also help in requirements engineering education, since, by enabling students to listen to each others' interviews, can let them learn from the observed successful elicitation strategies and mistakes.

References

1. Davis, A., Dieste, O., Hickey, A., Juristo, N., Moreno, A.M.: Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. In: RE'06, IEEE (2006) 179–188
2. Hadar, I., Soffer, P., Kenzi, K.: The role of domain knowledge in requirements elicitation via interviews: an exploratory study. REJ **19**(2) (2014) 143–159
3. Coughlan, J., Macredie, R.D.: Effective communication in requirements elicitation: a comparison of methodologies. Requir. Eng. **7**(2) (2002) 47–60
4. Zowghi, D., Coulin, C.: Requirements elicitation: A survey of techniques, approaches, and tools. In: Engineering and managing software requirements. Springer (2005) 19–46
5. Gervasi, V., Gacitua, R., Rouncefield, M., Sawyer, P., Kof, L., Ma, L., Piwek, P., De Roeck, A., Willis, A., Yang, H., et al.: Unpacking tacit knowledge for requirements engineering. In: Managing requirements knowledge. Springer (2013) 23–47
6. Sutcliffe, A., Sawyer, P.: Requirements elicitation: towards the unknown unknowns. In: RE'13, IEEE (2013) 92–104
7. Ferrari, A., Spoletini, P., Gnesi, S.: Ambiguity cues in requirements elicitation interviews. In: RE'16, IEEE (2016) 56–65
8. Rugg, G., McGeorge, P., Maiden, N.: Method fragments. Expert Systems **17**(5) (2000) 248–257
9. Friedrich, W.R., Van Der Poll, J.A.: Towards a methodology to elicit tacit domain knowledge from users. IJIKM **2**(1) (2007) 179–193
10. Ferrari, A., Spoletini, P., Gnesi, S.: Ambiguity as a resource to disclose tacit knowledge. In: RE'15, IEEE (2015) 26–35
11. Salger, F.: Requirements reviews revisited: Residual challenges and open research questions. In: RE'13, IEEE (2013) 250–255
12. : IEEE Std 1028-2008 - IEEE Standard for Software Reviews and Audits (2008)
13. Laitenberger, O., DeBaud, J.M.: An encompassing life cycle centric survey of software inspection. JSS **50**(1) (2000) 5–31
14. Shull, F., Rus, I., Basili, V.: How perspective-based reading can improve requirements inspections. Computer **33**(7) (2000) 73–79
15. Bacchelli, A., Bird, C.: Expectations, outcomes, and challenges of modern code review. In: ICSE'13, IEEE (2013) 712–721
16. Rigby, P.C., Bird, C.: Convergent contemporary software peer review practices. In: FSE'13, ACM (2013) 202–212
17. Fagan, M.E.: Design and code inspections to reduce errors in program development. **15**(3) (1976) 182–211

18. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Are the perspectives really different?: Further experimentation on scenario-based reading of requirements. In: Experimentation in Software Engineering. Springer (2012) 175–200
19. Femmer, H., Hauptmann, B., Eder, S., Moser, D.: Quality assurance of requirements artifacts in practice: A case study and a process proposal. In: PROFES 2016, Springer (2016) 506–516
20. Rosadini, B., Ferrari, A., Gori, G., Fantechi, A., Gnesi, S., Trotta, I., Bacherini, S.: Using NLP to detect requirements defects: an industrial experience in the railway domain. In: REFSQ'17. Volume 10153 of LNCS. Springer (2017) 344–360
21. Massey, A.K., Rutledge, R.L., Anton, A.I., Swire, P.P.: Identifying and classifying ambiguity for regulatory requirements. In: RE'14, IEEE (2014) 83–92
22. Ferrari, A., Spoletini, P., Donati, B., Zowghi, D., Gnesi, S.: Interview review: Detecting latent ambiguities to improve the requirements elicitation process. In: RE@next!, IEEE (2017)
23. Kof, L.: From requirements documents to system models: A tool for interactive semi-automatic translation. In: RE'10. (2010)
24. Ambriola, V., Gervasi, V.: On the systematic analysis of natural language requirements with Circe. ASE **13**(1) (2006)
25. Mich, L.: NL-OOPS: from natural language to object oriented requirements using the natural language processing system LOLITA. NLE **2**(2) (1996) 161–187
26. Mavin, A., Wilkinson, P., Harwood, A., Novak, M.: Easy approach to requirements syntax (ears). In: RE'09, IEEE (2009) 317–322
27. Pohl, K., Rupp, C.: Requirements engineering fundamentals. Rocky Nook, Inc. (2011)
28. Arora, C., Sabetzadeh, M., Briand, L., Zimmer, F.: Automated checking of conformance to requirements templates using natural language processing. TSE **41**(10) (2015) 944–968
29. Berry, D.M., Kamsties, E., Krieger, M.M.: From contract drafting to software specification: Linguistic sources of ambiguity (2003)
30. Gnesi, S., Lami, G., Trentanni, G.: An automatic tool for the analysis of natural language requirements. IJCSSE **20**(1) (2005)
31. Tjong, S., Berry, D.: The design of SREE — a prototype potential ambiguity finder for requirements specifications and lessons learned. In: REFSQ'13. Volume 7830 of LNCS. Springer (2013) 80–95
32. Gleich, B., Creighton, O., Kof, L.: Ambiguity detection: Towards a tool explaining ambiguity sources. In: REFSQ'10. Volume 6182 of LNCS., Springer (2010) 218–232
33. Femmer, H., Fernández, D.M., Wagner, S., Eder, S.: Rapid quality assurance with requirements smells. JSS **123** (2017) 190–213
34. Chantree, F., Nuseibeh, B., Roeck, A.N.D., Willis, A.: Identifying nocuous ambiguities in natural language requirements. In: RE'06. (2006) 56–65
35. Yang, H., Roeck, A.N.D., Gervasi, V., Willis, A., Nuseibeh, B.: Analysing anaphoric ambiguity in natural language requirements. Requir. Eng. **16**(3) (2011) 163–189
36. Katasonov, A., Sakkinen, M.: Requirements quality control: a unifying framework. REJ **11**(1) (2006) 42–57
37. Aurum, A., Petersson, H., Wohlin, C.: State-of-the-art: software inspections after 25 years. Software Testing, Verification and Reliability **12**(3) (2002) 133–154
38. Karras, O., Kiesling, S., Schneider, K.: Supporting requirements elicitation by tool-supported video analysis. In: RE'16, IEEE (2016) 146–155
39. Sharp, H., Rogers, Y., Preece, J.: Interaction Design: Beyond Human Computer Interaction, 4th edition. John Wiley & Sons (2015)
40. Höst, M., Regnell, B., Wohlin, C.: Using students as subjects, a comparative study of students and professionals in lead-time impact assessment. ESE **5**(3) (2000) 201–214