USING PRE-REQUIREMENTS TRACING TO INVESTIGATE REQUIREMENTS BASED ON TACIT KNOWLEDGE

Andrew Stone and Pete Sawyer
Computing Department
Infolab 21, Lancaster University, Lancaster, UK

Keywords: Tacit knowledge, Requirements, Tracing, Latent Semantic Analysis, Natural language processing.

Abstract: Pre-requirements specification tracing concerns the identification and maintenance of relationships between requirements and the knowledge and information used by analysts to inform the requirements' formulation. However, such tracing is often not performed as it is a time-consuming process. This paper presents a tool for retrospectively identifying pre-requirements traces by working backwards from requirements to the documented records of the elicitation process such as interview transcripts or ethnographic reports. We present a preliminary evaluation of our tools performance using a case study. One of the key goals of our work is to identify requirements that have weak relationships with the source material. There are many possible reasons for this, but one is that they embody tacit knowledge. Although we do not investigate the nature of tacit knowledge in RE we believe that even helping to identify the probable presence of tacit knowledge is useful. This is particularly true for circumstances when requirements' sources need to be understood during, for example, the handling of change requests.

1 INTRODUCTION

Requirements specifications are incapable of representing a problem domain in its entirety in all but the most trivial cases. One of the reasons for this is that much of the knowledge about the problem domain is tacit in nature.

The notion of tacit knowledge was first extensively explored by Michael Polanyi in his seminal book "The Tacit Dimension" (Polanyi, 1983). Polanyi briefly summarises tacit knowledge as "knowing more than you can tell", that is, knowledge that is so inbuilt within your own understanding of a process that awareness of this knowledge is neither apparent, nor explicable. Kevin Ryan (Ryan, 1993) presented a modern corollary when expressing concerns about the role of Natural Language Processing (NLP) in the requirements engineering process. Ryan's statement that "neither informal speech nor natural language text is capable of expressing unambiguously the myriad facts and behaviours that are included in large scale systems" reflects the tacit knowledge embedded within the problem domain.

Requirements often embody tacit knowledge that the analyst already has, or has uncovered from their analysis of the problem domain. The starting point for our research is that the identification of knowledge would help in two ways. Firstly, it would help the validation of requirements. Secondly, it would help in situations such as system evolution or dealing with requirement change requests, where the provenance of requirements needs to be understood. We are investigating this problem by developing tool support for a form of pre-requirements tracing designed to establish backwards traces from requirements into extant textual source material such as interview transcripts. We hypothesise that where provenance cannot be established between requirements and source material, this may indicate the influence of tacit information during synthesis of the requirements. Of course, there are other reasons for why requirements might lack identifiable provenance, but identifying a lack of provenance is interesting in itself as it permits requirements analysts to determine common sources of requirements ambiguity. This paper explains our approach to pre-requirements tracing and tacit knowledge identification and presents initial results from applying our tool.

Stone A. and Sawyer P. (2006).
USING PRE-REQUIREMENTS TRACING TO INVESTIGATE REQUIREMENTS BASED ON TACIT KNOWLEDGE.
In *Proceedings of the First International Conference on Software and Data Technologies*, pages 139-144
DOI: 10.5220/0001311701390144

2 TRACING AND TACIT KNOWLEDGE

Gotel and Finkelstein (Gotel and Finkelstein, 1994) identify both the need for and the difficulties associated with requirements tracing. They divide tracing into two classes: pre- and post- requirement specification tracing, which are analogous to high-end and low-end tracing as mentioned in (Ramesh and Jarke, 2001). Pre-requirement specification tracing is concerned with the requirement's life before it is included in the requirements specification. Post-requirements specification tracing deals with life after inclusion. Pre-requirement specification tracing is underdeveloped compared to post-requirement specification tracing. One problem standing in the way of pre-requirements specification tracing is that requirements synthesis often involves much more than a simple transformation process in which information elicited from stakeholders is re-written.

This is particularly well illustrated by the use of contextual elicitation techniques such as ethnographic analysis. Contextual techniques result in a rich description of the problem domain. On the one hand, this makes identification of tacit knowledge easier by the analyst. However, even where a requirement is derived from explicit elicited information with minimal application of tacit knowledge, the relationship between the raw elicited material and the requirement may be hard to identify without careful reading of both. Certainly, the lexical similarities between the source material and the requirement may be very weak.

The impact of tacit knowledge makes the identification of a requirement's provenance much harder still. A previous study on the use of ethnography in systems engineering (Bentley et al., 1992) analysed the working practises of Air Traffic Controllers (ATC). Embedded within this poorly structured information are examples of tacit knowledge. When confronted with a slow aeroplane about to enter a busy sector in which all flight levels (permitted altitudes of flight) will shortly be filled, the sector chief rerouted the slow aeroplane to another sector as shown in Figure 1.

The ethnographer explicitly identified this as an example of tacit knowledge as at no point are any details about the aircraft in question mentioned, not even the originating sector, yet the chief is still able to reroute the aircraft. When questioned later the chief replied that he knew which aircraft was in question just by looking at the radar. Plausibly, therefore, an analyst experienced in the ATC domain might synthesise a requirement about the radar display that provided the information used by the chief. Since the nature of this information is only implicit in the ethnographic report, the provenance of the radar display requirement would be difficult to trace were the requirement and ethnographic report the only information available for seeking the trace. Dealing with this limited, textual information is the subject of the next section.

10.56 Wing writes a height revision on a controller's livestrip following a telephone call. (Inbound from Scottish. Much of this co-ordination is done on the wings.)
11.05 Controller PH to Controller IS: 'you can track Mac9025 to me, ...'
[Controller IS is on the telephone]: 'pardon?'
Chief: 'J...'ll take 9025'
Controller IS: 'oh ... OK ...'
11.17 SA: 'Chief theres this he wants '
Chief: 'all levels are blocked through there '
Spends a moment thinking
Chief: 'no, he's a slow one there's no way he'll be clear then so we'll take him through Liffy'

Figure 1: An example of tacit knowledge embedded in a typical air traffic control scenario.

3 IDENTIFYING TRACES IN NATURAL LANGUAGE

Requirements are typically represented in natural language. Determining any semantic meaning from natural language will require an understanding of the language that comprises it. Rule based approaches to linguistics are brittle in the face of linguistic variability and do not scale well to new problem domains which introduce unique vocabulary. Alternative approaches rely on statistical properties of the text, this gave rise to the notion that language is understandable by observation, rather than the classical theoretical linguistic approach. Statistical analysis takes place on a body of language, or corpus, and is composed of examples of natural language potentially in the scale of millions of words.

The applicability of corpus linguistics to document processing in requirements engineering has been shown in several problem domains and at different levels. Rolland and Proix provide a general background for the applicability of natural language, and therefore natural language processing, to requirements engineering (Rolland and Proix, 1992). Gervasi and Nuseibeh use automated lightweight techniques to provide automated validation of requirements in some of NASA's requirements specifications (Gervasi and Nuseibeh, 2002). Sawyer et al. (Sawyer et al., 2005) provide evidence that probabilistic natural language processing is applicable to requirements engineering processes across different domains. One such technique is Latent Semantic Analysis (LSA).

Latent Semantic Analysis (LSA) is a vector space technique that results in the formation of a multidimensional, document-word space (Deerwester et al., 1990). It is computationally intensive but allows intelligent document query and retrieval whilst overcoming the traditional problems of polysemy (multiple meanings per word) and synonymy (multiple words that mean the same thing) (Berry et al., 1995; Dumais, 1991). The number of occurrences of each word in a document determines the document's magnitude in that dimension, thereby determining the position of the document in the space. Similar documents appear to cluster together in the space. This clustering can be heightened by reduction of the space to fewer dimensions by singular value decomposition. Similarity can therefore be determined via a variety of algorithms, such as simple Euclidean distance. LSA is commonly accepted to be a shallow technique that accurately manages to approximate human expectations of linguistic comparison.

A simple document-word space technique, although not LSA, has been used by Johan Natt och Dag et al. (Natt och Dag et al., 2005) to determine linguistic equivalence between two different sources of requirements : market requirements and business requirements. The lexical technique used resulted in more than 50% of correct links between requirements being identified. Further, it was estimated that up to 63% of similar requirements could be identified in this manner. However, this technique is based on lexical similarity measures. It has not been determined if this technique can be used to infer semantic similarities across the wide variety of document types required for pre-requirement specification tracing.

4 PERFORMING PRE-REQUIREMENTS TRACING

By searching for traces between requirements and their respective sources it should be possible to determine requirements that are not firmly derived from source material, thereby reflecting an instance of either:-

- Poorly sourced knowledge, that is knowledge which is not clearly defined and should therefore be subject of further investigation
- A form of tacit knowledge, whose presence in the requirements specification demonstrates a description of the external behaviour of a tacit process

Note that we are not seeking to measure requirements completeness. Establishing the absence of requirements that represent information explicit in the source material or (even harder) implicit from tacit knowledge, is outside the scope of this work. The tool implements three distinct phases of analysis:

Collation All source documentation and the current version of the requirements specification are prepared here. Several steps are performed, such as collating all the documents into a single logical collection for easier processing, tokenisation, stemming and the removal of syntactic elements of speech. The source material is then split into chunks to enable comparison. As currently implemented, the size and content of chunks are determined by a heuristic boundary detection algorithm (Manning and Schütze, 2000)

Comparison The semantic equivalence of chunks is determined by use of LSA. Chunks of source material are them compared against chunks of the requirements specification; the similarities are noted. The application of LSA that we propose requires that the contents of all documents are compared to produce a document similarity matrix. The document similarity matrix contains numbers in the range [1,1], where -1 represents content that is semantically divergent, and 1 represents content that is semantically identical

Analysis Candidates of matching chunks are presented to the analyst who may filter the results to increase clarity. Only candidate matches are displayed and it is left to the analyst to finally confirm or deny a candidate match

An overview of these operations is presented in Figure 2.

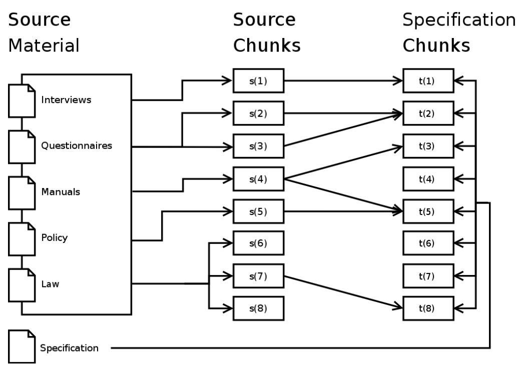

Figure 2: Identification of sources of requirements. Here chunks t(6) and t(7) are likely to be identified by the system as examples of tacit or poorly sourced knowledge as their source is not known. Note that not all source chunks may contribute to the requirements specification.

5 CASE STUDY

In order to test the validity of our approach LSA was used to trace between a concept of operations for a new system and an ethnographic report of the existing system. The ethnographic reports relate to a UK air traffic control system. The concept of operations was developed by Bentley (Bentley, 1994) for a tool to prototype ATC systems. The ethnographic report was scanned from a printed document using optical character recognition techniques. It contained scanning errors that resulted in spelling and grammatical mistakes that we left uncorrected in order to better approximate real world documents. Neither the concept of operations nor the ethnographic data are as vocabulary rich as the newspaper stories considered earlier. Therefore they were much less computationally expensive to perform LSA on. The full process took under a minute on a desktop machine.

We have not yet conducted a study to determine the effects of varying the size of each document chunk, although a trade off becomes immediately apparent. This is that small chunk sizes (e.g. single sentences) can lead to difficulty in analysts accurately interpreting results as there are too many chunks and relations to concurrently track. Larger chunk sizes abstract a lot of the information and result in an overly granular comparison. We decided to use 5 sentences per chunk for this experiment. This is somewhat arbitrary and future versions will use variable size chunks, so for example, the analyst can investigate individual requirements clauses or steps in a scenario. This chunk size was used on both the concept of operations and the ethnographic report.

5.1 Evaluation

Two measures that can be used to demonstrate that LSA is matching human expectation are recall and precision. In order to calculate these measures, it is first necessary to manually determine the correct links between the concept of operations and the ethnographic reports. The recall and precision may then be calculated as follows:-

1. Compute the similarities between chunks
2. Select a threshold, α in the range [-1,1]
3. Select a chunk of the concept of operations, i
4. Manually compare i to all chunks of the ethnographic report to produce a set of matches, r_{man}
5. For chunk i determine all the chunks of the ethnographic report that have a similarity value greater than α to produce a set of matches r_{lsa}
6. Calculate the recall as

$$recall = \frac{|r_{man} \cap r_{lsa}|}{|r_{man}|} \tag{1}$$

7. Calculate the precision as

$$precision = \frac{|r_{man} \cap r_{lsa}|}{|r_{lsa}|} \tag{2}$$

Essentially, recall can be seen as the percentage of correct associations in the current list with respect to the total number of correct associations, i.e. how many correct associations have been discovered at this point. Precision is the percentage of correct associations with respect to the size of the associations list, i.e. how many of the results are correct. It is therefore expected that the recall of LSA will be high when the threshold is low. By setting the threshold to -1 (the lowest threshold possible) all documents will be included in R_{lsa}, ensuring total recall. In other words, every chunk in the concept of operations will appear to be derived from every chunk in the ethnographic report. However, this will result in poor precision as the number of incorrect associations in R_{lsa} is high. As the threshold tends towards 1 precision should increase as the weak and noisy candidate matches are eliminated.

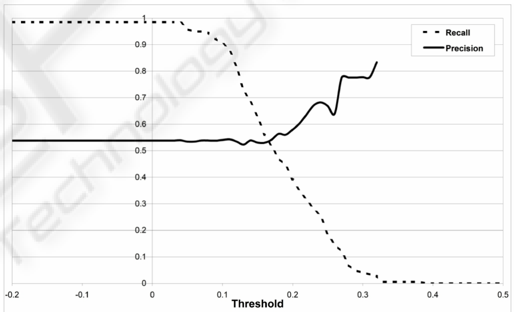

Figure 3: Recall and precision as a function of threshold.

In order to test that LSA can be used to perform semantic level comparison on these sorts of documents, the associations between 4 of the 25 chunks of the concept of operations were recorded against the 85 chunks of the ethnographic report. These manual associations were then used to plot the recall and precision against threshold, as shown in figure 3. This figure is made from a population sample; corresponding confidence interval plots are presented in figures 4 and 5. These plots show the 95% confidence interval for each sampled point, i.e. the range in which 95% of all members of the population are contained within assuming a normally distributed sample, calculated as $\bar{x} \pm 1.96(\frac{\sigma}{\sqrt{n}})$.

Figure 3 clearly shows that as the minimum threshold of relatedness increases the recall decreases and the precision increases. This provides evidence that LSA is approximating human expectations of semantic equivalence for the documents being considered.

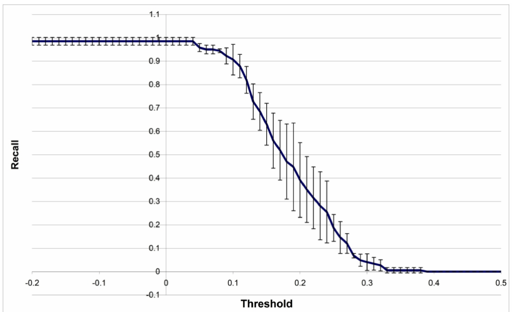

Figure 4: Recall and associated 95% confidence intervals.

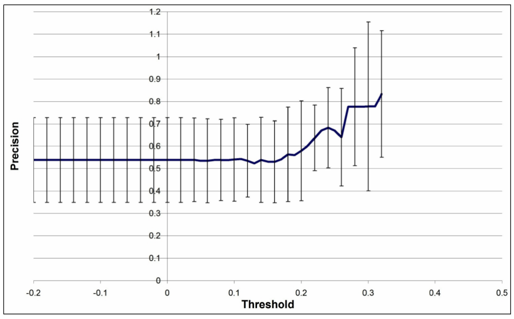

Figure 5: Precision and associated 95% confidence intervals.

If LSA was providing the opposite of human expectation we would expect to see the precision drop as a function of threshold. If LSA was producing random results we would expect to see no trend at all in the precision and recall curves.

5.2 Badly Sourced Material

We define any chunk as being badly sourced if it has no relatedness to chunks belonging to other documents for $\alpha > 0.1$. An examination of the chunks of the concept of operations that were poorly sourced fell in to two main categories:

1. A detailed description of the semantics of shared user displays. These were requirements invented by Bentley as part of his work on shared displays.
2. Chunks where Bentley has used knowledge from his own field work at the ATC centre and knowledge elicited by him from the ethnographer. Neither type of information were explicitly represented in the ethnographic report.

Other, less significant examples of poorly sourced text were due to us erroneously scanning too much of one of the leading pages in the document that contained the concept of operations, but was unrelated to the concept of operations. LSA correctly identified this material as not being associated with the ethnographic report. The results also include examples of the tool correctly identifying poorly sourced chunks of Bentley's concept of operations as potentially tacit in nature. One example of this is a chunk chunk of text that contains the lexical term 'strip'. Strip is a common word in both documents, but despite this the chunk is correctly identified as poorly sourced. The chunk deals primarily with a description of the pragmatics of different views of the airspace, such as a written strip view or a radar view. Similarly, despite many instances of the word 'radar' in the ethnographic document no strong link is made with this chunk, as LSA has correctly identified that this chunk is primarily concerned with a concept not covered in the ethnography.

6 LIMITATIONS & FUTURE WORK

Our approach assumes that a significant proportion of requirements are derived relatively directly from elicited problem domain information. If most of the requirements are invented rather than derived the number of candidate matches will be too low for the tool to offer any useful insights into requirements provenance. In addition, there are four factors that constrain the circumstances in which our approach is usable:

Media are not necessarily in text form. Video, audio and pictorial sources of information may be used to inform a requirements specification.

Media availability reduces the accuracy of the system if not all source media are available. The system is likely to identify many cases of tacit knowledge if the amount of source material is relatively small.

Inconsistent vocabulary reduces the accuracy of techniques such as LSA. There is potential to incorporate tools such as WordNet (Miller et al., 1990) to determine lexical similarity via synonym sets.

Document evolution may result in new associations appearing and old associations being removed.

Within these constraints we believe that our preliminary results demonstrate the potential of LSA to offer insights into requirements provenance and the influence of tacit knowledge. However, as noted above, we need to provide greater flexibility over chunk size. In particular, chunks must map onto the requirements, use cases, business events, or whatever is the natural unit of traceability in the requirements document under analysis. This will inevitably require some manual pre-processing by the analyst.

We also plan to evaluate LSA against other techniques that may yield similar or better results. In particular, text reuse algorithms used in plagiarism detection technologies may provide meaningful output, such as n-gram overlap (Clough et al., 2002), substring matching via greedy string tiling (Wise, 1996) and sentence alignment (Piao et al., 2002).

7 CONCLUSION

We propose a method of pre-requirements tracing that uses a corpus linguistics technique to achieve semantic-level comparison. By splitting up requirements specifications and the source material from which they were derived into chunks and comparing their semantic similarities, it is possible to determine likely sources for each chunk of the requirements specification. Further, this permits us to identify requirements not firmly derived from the supplied source material. We argue that these requirements represent either poorly sourced knowledge or instances of tacit knowledge embedded in the problem domain or the analyst's mind. We have demonstrated that LSA, a linguistic technique designed to overcome the problems of polysemy and synonymy, can approximate human expectations of semantic relatedness between chunks of source material and their resulting specification. The source material contains less rich text than found in other domains, such as newspaper articles, but is still able to match human expectation. We plan to show that this technique can be used to identify instances of tacit processes and enable pre-requirements tracing on an on-going software development project to update the student registry system at Lancaster University.

REFERENCES

Bentley, R. (1994). *Supporting Multi-User Interface Development for Cooperative Systems*. PhD thesis, Lancaster University.

Bentley, R., Hughes, J. A., Randall, D., Rodden, T., Sawyer, P., Shapiro, D., and Sommerville, I. (1992). Ethnographically-informed systems design for air traffic control. In *Proceedings of ACM CSCW'92 Conference on Computer-Supported Cooperative Work*, Ethnographically-Informed Design, pages 123–129.

Berry, M. W., Dumais, S. T., and O'Brien, G. W. (1995). Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595.

Clough, P. D., Gaizauskas, R., Piao, S. L., and Wilks, Y. (2002). Measuring text reuse. In *Proceedings of the 40th Anniversary Meeting for the Association for Computational Linguistics*.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *J. Am. Soc. for Inf. Sci.*, 41(6):391–407.

Dumais, S. T. (1991). Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments and Computers*, 23:229–236.

Gervasi, V. and Nuseibeh, B. (2002). Lightweight validation of natural language requirements. *Software Practice and Experience*, 32(2):113–133.

Gotel, O. C. Z. and Finkelstein, A. C. W. (1994). An analysis of the requirements traceability problem. In *First International Conference on Requirements Engineering (ICRE)*, pages 94–101. IEEE Computer Society Press.

Manning, C. D. and Schütze, H. (2000). *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, England.

Miller, G. A., R., B., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database. *Journal of Lexicography*, 3(4):234–244.

Natt och Dag, J., Gervasi, V., Brinkkemper, S., and Regnell, B. (2005). A linguistic-engineering approach to large-scale requirements management. *IEEE Software*, 22(1):32–39.

Piao, S. S. L., Gaizauskas, R., Clough, P. D., and Wilks, Y. (2002). Detecting measuring text reuse based on alignment. *Natural Language Engineering (submitted)*.

Polanyi, M. (1983). *The Tacit Dimension*. Paul Smith Publishing. ISBN 0-8446-5999-1.

Ramesh, B. and Jarke, M. (2001). Toward reference models of requirements traceability. *IEEE Trans. Software Eng*, 27(1):58–93.

Rolland, C. and Proix, C. (1992). A Natural Language Approach For Requirements Engineering. In Loucopoulos, P., editor, *Proceedings of the Fourth International Conference CAiSE'92 on Advanced Information Systems Engineering*, volume 593, pages 257–277, Manchester, United Kingdom. Springer-Verlag.

Ryan, K. (1993). The role of natural language in requirements engineering. In *Proceedings of the IEEE Int. Symposium on RE*, pages 80–82.

Sawyer, P., Rayson, P., and Cosh, K. (2005). Shallow knowledge as an aid to deep understanding in early phase requirements engineering. *IEEE Trans. Software Eng*, 31(11):969–981.

Wise, M. J. (1996). YAP3: Improved detection of similarities in computer program and other texts. *SIGCSE Bulletin (ACM Special Interest Group on Computer Science Education)*, 28.