Assisting requirements engineering with semantic document analysis

Paul Rayson, Roger Garside and Pete Sawyer
Computing Department
Lancaster University
Lancaster, UK. LA1 4YR.
{paul, rgg, sawyer}@comp.lancs.ac.uk

Abstract

Requirements engineering is the first stage in the software life-cycle and is concerned with discovering and managing a software system's services, constraints and goals. Requirements engineers frequently face the task of extracting domain knowledge and recovering requirements from large documents. This is needed to complement the often incomplete information elicited from the people who will use or otherwise have a stake in the system to be developed. The documents that have to be analysed may vary from structured documents, such as specifications of work processes, to unstructured, verbatim reports of interviews or workplace observations. This paper shows that tools exploiting natural language processing techniques, in particular semantic analysis, are able to assist in retrieval from these documents.

1. Introduction

Requirements engineering is a vital component in the development or evolution of any software system. It comprises the systematic process of discovering, understanding, analysing, documenting and managing the requirements of a system. The requirements can include specifications of the services the system should provide, the constraints on the system and background information which is necessary to develop the system.

One of the most problematic tasks in requirements engineering is that of discovering the requirements and forming an understanding of them in their proper business context. It is seldom sufficient to elicit requirements by face-to-face meetings or interviews with the system stakeholders. These techniques are important but problematic if not supplemented with other sources of requirements information. For example, a user may simply be unable to articulate the complexities of their job and unless the requirements engineer recognises that their description is incomplete, they will be unable to specify a system that adequately supports the job. In order to fill in the missing information and to understand the information provided that the stakeholders have provided, the requirements engineer needs to supplement it with information about, for example:

- the stakeholders' work domain;
- the structure of their organisation;
- the constraints and rules to which users of a system work;
- the operational environment in which the system will execute.

One technique that has been proposed as a means to build a deeper understanding of these is ethnography (Blythin et al, 1997). Here, an organisation is treated as a social structure and is passively observed over a protracted period (perhaps several months). This can reveal subtle but important insights into how the work is really performed, situated in its organisational and social context. However, ethnographic studies generate copious field notes that have to be analysed at a

later date. This has proven an Achilles heel since it is typical for the density of significant information to be low. Hence, it is hard to extract and structure.

This paper reports some of the preliminary results of the REVERE project where we are investigating tools to support the discovery of requirements information from large natural language document bases. The paper presents the results of applying the techniques developed in REVERE to ethnographic field reports as exemplars of documentary sources of requirements information.

2. Requirements recovery from documents

A requirements engineer must use whatever information resources are available to construct conceptual models of an organisation and its business processes and from these derive the system requirements (Butler et al, 1999). This typically entails an iterative process of inferring stakeholders, roles, tasks and business objects and verifying these against the structure and behaviour of business processes to be supported.

This information has to be gathered from many different sources, both human and documentary. The elicitation of information from human stakeholders has received a great deal of attention elsewhere. The retrieval of requirements information from free text documents has, by comparison, been neglected.

The documentary sources of information that are available may span a range of document types. At one extreme are structured documents such as specifications of working practices or of the system to be replaced or enhanced. Good requirements engineering practice (Sommerville & Sawyer, 1997) recommends that requirements are carefully documented and managed. Hence, it should be straightforward to list the stakeholders' requirements, understand their motivation and consequent trade-offs, and trace them forwards (ultimately) into the operational software. Unfortunately, good requirements practice is rare and systems are still routinely constructed with minimal requirements documentation (Melchisedech, 1998). Some domains (e.g. defence) place a premium on documentation and here it is reasonable to expect requirements specifications, operating procedures, safety cases, etc. to be available. In most cases the available documentation will be less comprehensive but if any documentation does exist, it will represent a potentially important resource; particularly where human expertise is patchy.

At the other extreme of the range of document types are unstructured documents such as interview transcripts or ethnographic field reports. Ethnographic field reports are typically compiled over several months and may be very large. An ethnographic field report cannot serve as a requirements document but contributes information that may be analysed, structured and incorporated into a requirements specification. A field report may contain everything form the ethnographer's thoughts to verbatim records of conversations within the workplace.

Given the existence of a body of documentation, the real problem is then how to process it. This will be hard if the volume is large; in extreme cases, there may be filing cabinets full. Similarly, variable quality and the structure of the documents will pose problems if, for example, they are heavily cross-referenced and version control has been poor. This is compounded by the linear structure of paper documents. Even if the documents have good tables of contents, have comprehensive indexes and are in, or can be transformed (via scanning and OCR) into, electronic form, the documents' as-written structure inevitably constrains the way in which people can read and interact with them.

Identification and assimilation of the subset of useful information contained in the documents is therefore difficult, costly and error-prone.

Our aim is to develop tools to ease these problems by exploiting mature techniques for natural language processing (NLP). Although technical documents often employ special notations, such as object models, workflow diagrams, etc., the bulk of nearly all such documents is comprised of natural language. Ryan (1993) has noted the promise of NLP for information retrieval from textual requirements (Blair & Maron, 1990). Several researchers have used rule-based NLP techniques for synthesising database conceptual schemas (Rolland & Proix, 1992), generating graphical representations of VLSI system requirements (Cyre & Thakar, 1997) and automatically abstracting requirements from requirements specifications (Goldin & Berry, 1997). However, these approaches are all hamstrung by the limitations of the rule-based NLP techniques they employ. All the techniques above depend for their efficacy on a tightly constrained subset of English.

Software analysts have no control over the documents that they must analyse so purely rule-based techniques are impractical. Our approach is therefore to exploit *probabilistic* NLP techniques that were pioneered at Lancaster and a number of other sites in the 1980s. The tools have been trained on very large corpora of free text which have already been analysed and 'tagged' (often manually) with each word's lexical, syntactic or semantic category. Extremely large corpora have been compiled (the British National Corpus consists of approximately 100 million words (Aston & Burnard, 1998)). For some levels of analysis, notably part-of-speech tagging, probabilistic NLP tools have been able to achieve levels of accuracy and robustness that rule-based techniques cannot approach.

These probabilistic tools do not attempt to automate understanding of the text. Rather, they abstract interesting properties of the text that a human user can combine and use to infer meaning. Evidence from other domains suggests that such tools can be effectively used to provide a 'quick way in' to large documents. For example, in (Thomas & Wilson, 1996) probabilistic NLP tools were used to quickly confirm the results of a painstaking manual discourse analysis of doctor-patient interaction. In this application, they were also able to reveal information that had not been discovered manually.

A further, crucial, characteristic of probabilistic NLP techniques is that they scale. The execution time of the tagging process varies approximately linearly with the document size. Once the text has been tagged, retrieval and display tools allow the user to interact with the document. These use the tags to provide views on the document that reveal interesting properties and suppress the bulk of text. They do this in a way that is largely independent of the size of the document. Hence, the user is protected from information overload by being selective about the information they want to extract.

We do not claim that NLP will solve the problem of requirements recovery from documents. The information recovered will never be complete or a perfectly accurate snapshot of the software's requirements. What is recoverable is bounded by the quality of the documents themselves. However, at present, analysts have very few useful tools to help recover this information. We believe that probabilistic NLP are sufficiently mature to make a substantial improvement if they can be integrated within a framework of guidance for what properties to look for and how to interpret them.

3. Preliminary results

During the preliminary stage of REVERE, we have adapted, and experimented with, a set of existing NLP tools developed at Lancaster for the processing of English language text. The most important of these is CLAWS (Garside & Smith, 1997). CLAWS uses a statistical hidden Markov model technique and a rule-based component to identify the parts-of-speech (POS) of words in a document to an accuracy of 97-98%. CLAWS' output is the document text annotated with POS tags and this provides the foundation for further levels of analysis. A semantic analyser (Rayson &

Wilson, 1996) uses the POS-tagged text to assign semantic tags that represent the general sense field of words from a lexicon of single words and an idiom list of multi-word combinations (e.g. ‘as a rule’). This lexicon contains approximately 52000 words or idioms and classifies words according to a hierarchy of semantic classes. For example, the tag *A1.5.1* represents words or idioms meaning *Using* which is a subclass of *general and abstract terms*. Words that would be assigned this tag (in the appropriate POS context) include *user*, *end-user* and *operator*. Similarly, the tag X2.4 is a subclass of *Psychological actions, states and processes* and would be assigned to terms meaning *Investigate*, such as *search*, *browse* and *look for*. The disambiguation techniques employed consist of a multi-layered approach. We make use of seven major techniques:

1. *POS tag*: some senses are eliminated by the prior POS tagging
2. *General likelihood ranking*: our lexicon lists senses in order of likelihood in general English
3. *Overlapping idiom resolution*: heuristics which take account of length and span of multi-word idioms resolve overlaps
4. *Domain of discourse*: current domain alters rank ordering of senses in our lexicon
5. *Text-based*: to some extent a word keeps the same meaning throughout one text
6. *Template rules*: uses immediate context in which a word is restricted to a particular sense
7. *Local probabilistic*: sense is determined by the local context

These have been reported elsewhere in more detail (Garside & Rayson, 1997).

A further tool, XMATRIX (Rayson et al, 1997) provides a means for retrieving the analysis results. XMATRIX is used to perform *frequency profiling*. At the most basic, this produces a simple concordance of individual words in context. However, POS and semantic tagging allows more useful concordances to be formed. For example, XMATRIX can show the frequency of occurrence of different parts of speech. An obvious example of how this can benefit the identification of requirements is the extraction of all the occurrences of modal verbs ('shall', 'must', 'will', 'should', etc.). Expressions of need, desire, etc., consistent with user or system requirements can therefore be located in a document very easily and without the need to construct complex regular expressions or search templates. Even this basic level of analysis goes beyond what the current generation of commercial requirements and document management tools allow.

Frequency profiling becomes even more useful when a document can be compared against a *normative corpus*: a large body of pre-tagged text from a representative domain. For this we are tagging a number of public-domain software and systems engineering standards, operating manuals, a large IBM technical documents corpus, a corpus of text from the applied sciences (a subset of the British National Corpus) and a number of technical reports and papers. Comparison with the normative corpus allows information to be extracted from a document by searching for statistically significant deviations from the frequency norm suggested by the corpus.

To provide a snapshot of the results so far, and illustrate some of the key issues, we now briefly describe two examples based upon the analysis of ethnographic field reports.

3.1 Example #1: Air-traffic control

The target documents are field reports of a series of ethnographic studies at an air traffic control centre. These consist of both the verbatim transcripts of the ethnographer’s observations and interviews with controllers, and of reports compiled by the ethnographer for later analysis by a multi-disciplinary team of social scientists and software engineers. The volume of the text is 103 pages and, significantly, the density of the information is high and the documents are not structured in a way (say around business processes or system architecture) designed to help the extraction of requirements.

The tools were applied to the document and two different normative corpora were used in the analysis. These represented two different approaches to corpus selection. The first was an attempt to introduce an element of domain filtering. In the absence of a corpus of the ATC domain, we selected (perhaps naively), one oriented to technology. This is a subset of the BNC from the pure and applied science section of the BNC related to information technology. Collectively, these form a corpus of 1.7 million words, of which about 60% are news stories relating to IT. Using this corpus, the most over-represented semantic categories in the ATC field reports are shown in table 1 (sorted by significance of deviation from the norm).

1927.7	M5	flying (‘plane’, ‘flight’, ‘airport’)
1500.8	Z8	pronouns (‘you’, ‘they’)
627.2	Z4	interjections (‘um’, ‘yeah’)
502.9	S7.1+	power, organising (‘controller’, ‘chief’)
412.8	A3+	being (‘is’)
330.1	L1+	life and living things (‘live’)
239.9	O3	electrical equipment (‘radar’, ‘blip’)
234.1	W3	geographical terms (‘Pole Hill’, ‘Dish Sea’)
184.5	Z6	negatives (‘not’)
158.6	N3.7	measurement (‘length’, ‘height’, ‘distance’, ‘levels’, ‘1000ft’)
146.8	M1	moving (‘arrival’, ‘incoming’, ‘climb’, ‘descent’)
126.3	L2	living creatures (‘wing’, ‘wings’)

Table 1 Over-represented semantic categories in the ATC domain

The figures in the left-hand column represent the semantic categories' log-likelihood (LL) figure - a measure of statistical significance. The middle column is the mnemonic for the semantic category and the right-hand column is the semantic category’s label and typical examples from the analysed text.

The LL shows the degree of deviation from the norm. A LL greater than 6.63 indicates that there is a 99% confidence in the result's reliability. Hence, the above 12 semantic categories stand out as highly unusual deviations from normal English usage, which suggest that it would be rewarding to investigate the usage of the corresponding words or idioms within the document.

The following summarises the useful data from this list:

- M5 is a major class of domain objects and emerges clearly as the most over-represented category.
- S7.1+ includes the principal user roles.
- L1+ (the word ‘live’) is an attribute of a major object type (flight strip) who’s LL is too small for it to appear on this list (i.e. ‘live strip’).
- O3 includes electronic objects and properties of those objects.
- W3 are instances of geographical objects (the names of sectors of airspace and aircraft reporting ‘beacons’).
- N3.7 includes attributes and values taken by some of the objects
- M1 shows actions performed by some of the objects
- L2 refers to an area of the ATC suite and the people who work in that area (‘wingmen’)

Hence, by using XMATRIX to view the semantic categories in context, the analyst would be able to identify some of the major domain objects, roles, functions, etc. for ATC (see Figure 1). However, there is also much noise in the result. For example, Z8, Z4 and A3+ are largely a

consequence of the fact that much of the documents' contents are text quoted verbatim from users. Given this characteristic of the documents, it is perhaps unsurprising that an IT-oriented normative corpus should throw up significant deviations from the norm that are of no significance to the domain under investigation.

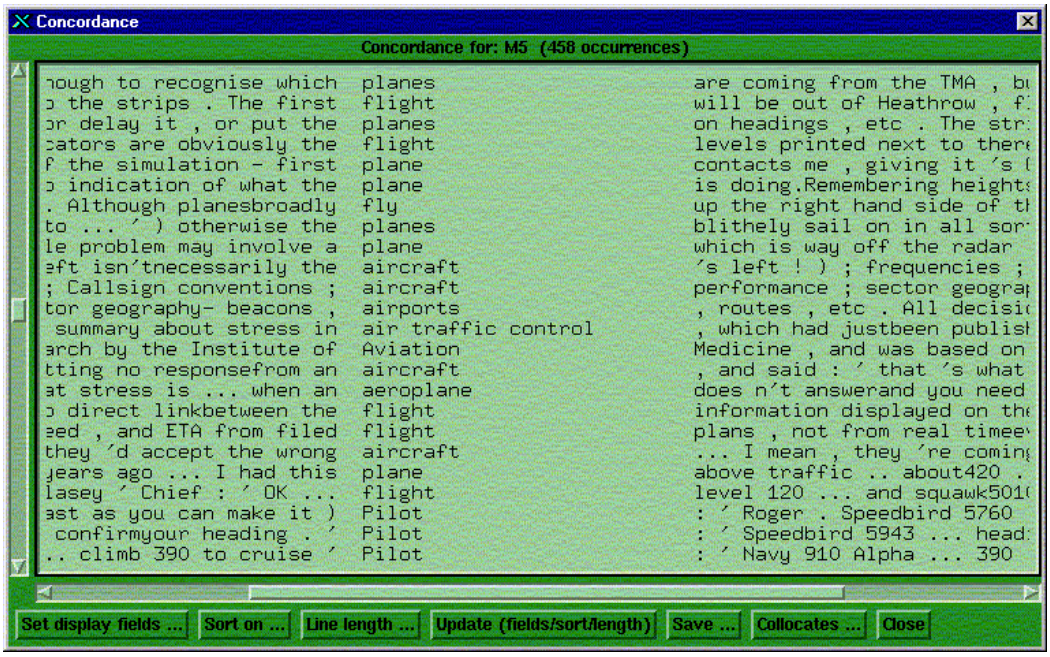

Figure 1. Browsing the semantic category *flying*

In view of this, the second normative corpus that we applied to the same documents was a 2.3 million-word subset of the BNC derived from the transcripts of spoken English. The results from this were quite different, see table 2.

3366.66	S7.1+	as above
2578.80	M5	as above
988.09	O2	general objects ('strip', 'holder', 'rack')
643.54	O3	as above
535.97	Y1	science and technology ('PH')
449.34	W3	as above
432.33	Q1.2	paper documents and writing ('writing', 'written', 'notes')
372.80	N3.7	as above
318.82	L1+	as above
310.32	A10+	indicating actions ('pointing', 'indicating', 'display')
306.90	X4.2	mental objects ('systems', 'approach', 'mode', 'tactical', 'procedure')
290.06	A4.1	kinds, groups ('sector', 'sectors')

Table 2 Over-represented semantic categories in the ATC domain

With the exception of Y1 (an anomaly caused by an interviewee's initials being mistaken for the PH unit of acidity), all of these semantic categories include important objects, roles, functions, etc. in the ATC domain.

It is interesting that in this example, the normative corpus that was not domain-specific performed better than the normative corpus that we (wrongly, as it turned out) guessed would provide a partial match to the domain. In retrospect, it seems likely that this was due to the fact that the domain of ATC has very little to do with IT. A more relevant comparison would be to derive an ATC normative corpus and apply that to the ethnographic field reports.

However, we believe that the results using the tools in their current untuned form are potentially very useful, provided that a normative corpus is carefully chosen. Initial results indicate that where no domain-specific corpus exists, it is better to default to a non-specific corpus, and then manually separate domain-specific from application-specific variations.

3.2 Example #2: Banking

As a further illustration of the success of a comparison to a non-specific corpus, we present a second example. The corpus under consideration contains reports and notes from an ethnographic study of financial services in a major ‘high street’ bank and fieldwork that examined features of working in a major Lending Centre and, in particular, the work of the 'Technology Co-ordinator' (Hughes et al, 1998). This is the largest of the two examples, consisting of 244 pages of text, to which we applied the same process as above. On comparison with the 2.3 million-word spoken subset, we observed the significant results as shown in table 3.

3461.86	S7.1+	power, organising: ('co-ordinate', 'control' as in lending control, 'administrative')
3369.71	I3.1	work and employment generally ('work', 'working', 'employed')
3318.69	Y2	IT and Computing ('computer', 'software', 'screen')
2343.24	Q1.2	paper documents and writing ('paperwork', 'file', 'notes', 'brief' as in Customer brief)
1857.55	S5+	groups and affiliation ('Branch', 'Team', 'Unit' as in Lending Unit)
1772.26	I2.2	business selling: ('lease', 'sales', 'flogging', 'markets')
1765.56	I2.1	business generally ('business' as in business centre, 'audit')
1361.74	S8+	obligation & necessity ('teamwork', 'support', 'service', 'Assistant Manager')
1340.13	A6.2+	comparing usual ('everyday', 'routine', 'standardised')
1127.62	I1	money ('buck', 'account', 'cheques', 'interest rates', 'ledger', 'balance')
1114.05	A9-	giving ('lending', 'borrowing', 'loan')
866.82	I1.1	affluence ('credit', 'income', 'profits', 'capital', 'funds', 'pay in', 'savings', 'investments')

Table 3 Over represented semantic categories in the banking domain

Once again, this second example shows that the software is able to extract important features from the content of the text under examination. In particular, I2.2 (business selling category) highlights the new nature of the bank's business of selling financial products.

4. Future development

Our next task is to devise a large-scale experiment with our project partners and refine the semantic categories and lexicon of words and idioms (originally defined for general English). At present, the lexicon contains a number of categorisations that, in an IT context, appear anomalous.

For example, while 'browse' would be tagged with the semantic category *investigate* along with 'search' and 'look for', 'query' would be tagged as a *speech act*. We expect that it will be necessary to refine the lexicon to derive an IT-oriented lexicon and semantic classification.

Our expectation is this would underpin a core toolkit that can be tailored for particular application domains. Hence, for example, an analyst would be able to add new semantic categories for banking or railway signalling. A capability already exists to refine the lexicon by reclassifying words or idioms. This will have to be extended with the additional abilities to import and classify new words or idioms, and to define new semantic categories.

5. Conclusions

This paper has described the preliminary results of the REVERE project investigating support for the discovery of software requirements in large documents. The motivation for this is that documents often form an important, and sometimes the primary, source of information about an organisation and its business processes. These documents are often large and of variable quality so extracting information from them can be costly. They are usually written in natural language.

We plan to integrate a number of techniques to provide a set of tools to help analysts explore the documentation, and reconstruct models of the business that motivated the software. At the core of this toolset are probabilistic NLP tools to provide a 'quick way in' to large, complex and imperfectly structured documents. At present, little effective support is available for the analysis of such documents. Probabilistic NLP offers the potential to save much painstaking and error-prone manual effort. A crucial requirement of our work is that it must scale in a way that the manual analysis of documents does not. The probabilistic NLP tools that we have chosen have been proven in other domains to do so. They are also mature and can tolerate variation in the use of language contained in documents. Hence, in contrast to other attempts at applying NLP techniques to the analysis of technical documents, they are not restricted to a well-defined subset of (e.g.) English.

The paper describes initial experiments on ethnographic field notes and reports from both the air traffic control and banking domains. These have been used to illustrate the use of frequency profiling, part-of-speech tagging and semantic tagging to reveal interesting properties of the text. For example, certain semantic categories of words appear in the documents with a frequency that significantly differs than a norm suggested by a corpus of English text. Examining occurrences of words of these categories reveals words that the analyst can infer to represent objects, roles and tasks.

6. Acknowledgements

The REVERE project (REVerse Engineering of Requirements) is supported under the EPSRC Systems Engineering for Business Process Change (SEBPC) programme, project number GR/MO4846. Further details can be found at: `http://www.comp.lancs.ac.uk/computing/research/cseg/projects/revere/`
We are grateful for the support of our industrial partner, Adelard, who has provided us with motivation, data and advice.

7. References

Aston, G. and Burnard, L. (1998). *The BNC Handbook: Exploring the British National Corpus with SARA*, Edinburgh: Edinburgh University Press.

Blair, D., Maron, M. (1990). Full-Text Information Retrieval: Further Analysis and Clarification, *Information Processing and Management*, 26 (3) (pp. 437--447).

Blythin, S., Rouncefield, M., Hughes, J. (1997). "Never Mind The Ethno Stuff - What Does All This Mean and What Do We Do Now?", *ACM Interactions*, 4 (3) (pp. 38--47).

Butler, K., Esposito, C., Hebron, R. (1999). Connecting the Design of Software to the Design of Work, *Communications of the ACM*. 42 (1) (pp. 38--46).

Cyre, W., Thakar, A. (1997). Generating Validation Feedback for Automatic Interpretation of Informal Requirements, in *Formal Methods in System Design*, Kluwer.

Garside, R., Smith, N. (1997). A Hybrid Grammatical Tagger: CLAWS4, in Garside, R., Leech, G., and McEnery, A. (Eds.) *Corpus Annotation: Linguistic Information from Computer Text Corpora*, (pp. 102--121) London: Longman.

Garside, R., and Rayson, P. (1997). Higher-level annotation tools, in Garside, R., Leech, G., and McEnery, A. (eds.) *Corpus Annotation: Linguistic Information from Computer Text Corpora*, (pp. 179--193) London: Longman.

Goldin, L., Berry, D. (1997). AbstFinder, A Prototype Natural Language Text Abstraction Finder for Use in Requirements Elicitation, *Automated Software Engineering*, 4 (pp. 375--412).

Hughes, J., O'Brien, J., Rouncefield, M., and Tolmie, P. (1998). Some 'real' problems of 'virtual' teams. Presented at the *Labour Process Conference*, Manchester.

Melchisedech, R. (1998). Investigation of Requirements Documents Written in Natural Language, *Requirements Engineering*, 3 (2) (pp. 91--97).

Rayson, P., Leech, G., and Hodges, M. (1997). Social differentiation in the use of English vocabulary: some analyses of the conversational component of the British National Corpus, *International Journal of Corpus Linguistics*. 2 (1) (pp. 133--152).

Rayson, P., and Wilson, A. (1996). The ACAMRIT semantic tagging system: progress report, In *Proceedings of Language Engineering for Document Analysis and Recognition (LEDAR)*, (pp. 13--20) Brighton, England.

Ryan, K. (1993). The Role of Natural Language in Requirements Engineering, In *Proceedings of IEEE International Symposium on Requirements Engineering*, San Diego (pp. 240--242).

Rolland, C., Proix, C. (1992). A Natural Language Approach for Requirements Engineering, *Lecture Notes in Computer Science*, Vol. 593.

Sommerville, I., Sawyer, P. (1997). *Requirements Engineering - A Good Practice Guide*, Chichester: John Wiley.

Thomas, J., Wilson, A. (1996). Methodologies for Studying a Corpus of Doctor-Patient Interaction, in Thomas, J. and Short, M. (Eds.) *Using Corpora for Language Research*, (pp. 92--109) London: Longman.