Linguistic Processing of Natural Language Requirements: the Contextual Exploration Approach

Jorge J. García Flores *

LaLICC
Université de Paris 4
96, bd Raspail,
75006 Paris, France
jorge.gflores@paris4.sorbonne.fr

Abstract. This paper studies linguistic approaches of requirements engineering. It proposes the contextual exploration method as a linguistic annotation technique to support requirements engineering activities, and specifically requirements analysis and validation. Our approach makes use of linguistic markers to extract, within large natural language requirements documents, those statements considered as relevant from requirements engineering perspective: concepts relationships, aspecto-temporal organisation, cause and control statements.

1 Introduction

Natural language plays an important role in requirements analysis. A recent study [1] shows that 73% of these documents are written in natural language. Natural Language Processing (NLP) provides useful techniques to extract information from textual requirements descriptions (TRD), like use cases [2], scenarios, user stories, transcriptions of conversations for requirements elicitation [3] and even rough sketches [4].

The purpose of this paper is to explore the use of a linguistic technique, the Contextual Exploration (CE) Method [5] to support RE activities, in particular requirements analysis and validation. This method organises linguistic knowledge under the form of hypothesis that could confirm or refute the attribution of a certain semantic value to a sentence, according to the presence of linguistic clues. This paper proposes that the following semantic viewpoints may be useful to support RE analysis and validation: 1) relations between concepts, 2) aspecto-temporal organisation, 3) control and 4) causality.

The paper is organised as follows: section 2 analyses linguistic approaches of RE, section 3 introduces the CE Method, section 4 is devoted to the discussion of how the CE method would assist RE analysis and validation, and section 5 presents the conclusions and sketches future work.

* PhD research funded by CONACYT (Mexican Government)

2 Linguistic approaches of RE: state of the art

The use of NLP tools for software development is not new. Already in the eighties, Chen [6] associated entities to nouns and relations to verbs in order to build E-R models from natural language sentences. With the emergence of the idea that software development cannot be performed effectively without a full understanding of requirements, NLP tools were used as a way to deal with the substantial amount of natural language documents produced during the requirements process. Table 1 shows some Natural Language Requirements Engineering (NLRE) systems that have been developed to assist RE activities with NLP tools .

Table 1. RE activities supported by NLRE approaches

NLRE system	Elicitation	Analysis-modelling	Documentation	Lexical/synt. validation	Semantic validation
Rolland & Proix [7]		•			
Macias & Pulman [8]			•		
Goldin & Berry [9]	•				
Osborne & MacNish [10]				•	•
Ambriola & Gervasi [11]		•			•
Ben Achour [12]		•	•		•
Burg [13]		•			•
Rayson et al [14]	•				
Natt och Dag et al [15]					•
Fabbrini et al [16]				•	
Gervasi & Nuseibeh [17]					•
Fantechi et al [18]				•	•
Fliedl et al [19]		•			

Requirements elicitation is supported by extracting relevant lexical entries from the mass of textual information produced by elicitation techniques (stakeholder interviews, group meetings, protocol analysis or participant observation [20]). Statistical annotation techniques (faster but less accurate) prevails in these systems over linguistic based ones, mainly because of the mass of textual information they deal with.

NLRE systems that support requirements analysis and modelling receive NL requirements specifications as an input. Input text can be written in free NL language or in a controlled form. Then, they build formal ([21, 13, 19]) or semiformal ([7, 11]) representations which are the base for further processing, like conceptual schema generation or automatic analysis. Linguistic annotation techniques can be used in this process. In Burg's [13] approach, WORDNET [22] based annotation is used to tackle lexical ambiguity and to specify the semantic roles of verbs. Ambriola and Gervasi's approach [11] uses semantic annotation of glossary entries to build semi-formal models by shallow parsing of NL requirements. Ben Achour [21] annotates NL scenarios with Fillmore's [23] cases to discover new scenarios.

In this paper, two kind of requirements validation activities are distinguished (see Table 1): on the one hand those activities that aim at progressively improving the linguistic quality of NL specification documents by lexical and syntactical means; on the other hand those activities pointing towards the validation of NL requirements semantic properties (e.g. completeness, coherence, consistency) by formal means. For instance, in Fabbrini et al. [16] the linguistic quality of NL specification documents is evaluated according to lexical/syntactical criteria (e.g. the word "clear" is an indicator of vagueness), so lexical and syntactical tags are necessary. In contrast, Gervasi and Nuseibeh [17] use CIRCE's [11] semantic annotation engine to support a formal validation model. They attach semantic tags to glossary entries (e.g. "1.1 sec" would have the *time span* tag, and "less than" the *condition* tag).

Table 2. NLP profile of NLRE approaches

NLRE	NLP strategy	NLP technique	Input lanuage	Analysis level	Large input support
Rolland & Proix [7]	FF	ling.	free	sentence	
Macias & Pulman [8]	SP	ling.	controlled	sentence	
Goldin & Berry [9]	-	stat.	free	-	•
Osborne & MacNish [10]	FF	ling.	controlled	sentence	
Ambriola & Gervasi [11]	SP	ling.	free	sentence	
Ben Achour [12]	FF	ling.	free	sentence	
Burg [13]	FF	ling.	controlled	discourse	
Rayson et al. [14]	-	stat.	free	-	•
Natt och Dag et al [15]	-	stat.	free	-	•
Fabbrini et al [16]	FF	ling.	free	sentence	
Gervasi & Nuseibeh [17]	SP	ling.	free	sentence	
Fantechi et al. [18]	FF	ling.	free	discourse	
Fliedl et al. [19]	FF	ling.	free	discourse	

From the analysis of the NLP profiles from NLRE systems (see Table 2)[1] we can make the following observations:

- Except for statistical approaches, few NLRE systems take into account the size of the input text. Large documents raise interesting questions for NLRE systems. What should be the processing's scope? Is it useful to process the

[1] NLP profiles are classified under the following criteria: 1) full-fledged (FF) vs shallow parsing (SP) for the NLP strategy; 2) statistical vs linguistic for the NLP technique; 3)free vs controlled for the input text language; 4) sentence and discourse for the analysis level and 5) yes or no for the large text support.

whole mass of documents, or is it better to limit it to some extent? What are the effects of large TRD on formal representations? Gervasi and Nuesibeh [17] suggest that, when using formal methods for requirements validation, partial processing is more effective than translating the entire document into a formal representation.

- Linguistic approaches work mainly at sentence level. Few NLRE systems take into account discursive relations. Dependencies and references between sentences are particularly important for semantic validation. Fantechi et al. [18] propose to go beyond the sentence level by extracting functional relations between semantically annotated use cases (e.g. to extract the collections of use cases where the *action* trace is present).
- Full fledged NLP techniques are necessary to improve the linguistic quality of NL specification documents, but not for semantic validation, which is perform using full-fledged (lexical + morphological + syntactic) analysis, shallow parsing or even statistical NLP techniques, like in Natt och Dag [15] et al. approach, where lexical statistics are used to organises requirements by clusters and look for redundancy. Generally speaking, full fledged techniques are used mostly by NLRE systems supporting requirements modelling and/or linguistic validation of NL requirements, while shallow parsing is used mainly for requirements validation and statistical approaches for requirements elicitation.

The novelty of using the Contextual Exploration method for RE is that:

1. It is oriented to the exploitation of large NL requirements documents by linguistic means.
2. It organises NL requirements according to semantic relationships established between requirement's arguments. Four semantic viewpoints are used: relations between concepts, aspecto-temporal relations, control, and causality.
3. It has rule-based structure that could allow the definition of semantic validations beyond the limits of a sentence, and even between requirements that are far away from each other in the requirements document.

Because of these characteristics, the CE method may be particularly useful to support requirements analysis and semantic validation of large textual requirements descriptions (TRD).

3 The Contextual Exploration Method

In the frame of the Cognitive and Applicative Grammar (GAC in French) linguistic theory [24], the contextual exploration (CE) method [5] was originally designed to solve lexical and grammatical polysemy problems. The method rests on the principle that domain independent linguistic elements structure text meaning. Its purpose is to access the semantic content of texts in order to extract relevant information according to a certain task.

3.1 Contextual Exploration Rules

According to contextual exploration, all signs occurring in a text (the textual context) must be taken into account to determine the semantic value of a sentence. This example illustrates how indeterminacy is solved[2]:

(1) ***In spite*** *of all precautions, he was* ***captured*** *the day after*
(2) ***Without*** *all precautions, he was* ***captured*** *the day after*

From the aspecto-temporal point of view, the tense of "was captured" is the linguistic marker of a semantic value that can be NEW-STATE (he was captured) or UNREALIZED (he was not captured). However, the tense itself is not enough to decide which one of the two values must be assigned, so the context has to be analysed in order to get more clues. In this case "in spite" and "without" are the clues that determine the semantic values of these sentences.

Linguistic markers correspond to plausible hypothesis that must be confirmed by the presence of certain clues. The heuristic mechanism is based on rules. A rule R is defined as follows:

$$R_k = [K, \{I_p, C_p\}, D_k]$$

Where K is a class of linguistic marker, $\{I_p, C_p\}$ a finite set of couples wich associates a linguistic clue I_p with the a textual context C_p, in order to take a decision D_k. In other words, if the linguistic marker K is found in the context C_p sourrounded by a set of clues I_p, the EC system must take the decision D_k [25]. Rules are declared in a language intended to separate their linguistic definition from their computer-based implementation. As an example, the following rule is meant to assign the semantic value INGREDIENCE to a sentence like "The hard drive is generally an element of the central unit"[3]:

```
Head: Ringrconst01
Task : Static Relations
Trigger: &ILIngNomConst
Body
       E1 := CreateSpace(Previous (I))
       E2 := CreateSpace (Afterwards (I))
       L1 := &line
       L2 := &2-points
       L3 := &to-be
       L4 := &LIngPrep2
Conditions
       There_is x belonging_to E1 that_class_of x belongs_to (L1 OR L2 OR L3)
       There_is y belonging_to E2 that_class_of y belongs_to (L4)
Actions
       SemanticLabel(ParentSentence(I), "is_part_of")
       SearchArgument2(ParentSentence(I), E1, E2 , x1->x2)
End
```

[2] This example is from G. Guillaume, quoted by Desclés. [5]
[3] This rule was written by F. Le Priol [26]

The class of markers triggering this rule is &ILIngNomConst. In the linguistic repository, this class contains 54 nouns, like "member", "piece", "part", "organ", etc. The occurrence of any of these markers triggers the rule, which looks in the context (afterwards and previous) for a flexional form of the verb to-be (clue-class &to-be) , the prepositions *de* (of), *par* (by) or *avec* (with) from the clue-class &LIngPrep2, and punctuation sign ":" (&2-points). Therefore, a sentence like "Hard drive : part of the central unit" would get the INGREDIENCE label as well. Linguistic clues may not only be lexical entries. Punctuation signs, text structural elements, or even discourse acts, like a definition or a conclusion, are allowed in a clue class. The context's scope is not the same for all clues. Desclés et al. [5] characterise four kinds of context:

- Context C1 is limited to the clause or sentence S where the semantic value is to be assigned.
- Context C2, necessary to solve anaphora, is limited to sentences belonging to the same paragraph, located just before the sentence S containing the anaphoric clue.
- Context C3 includes several sentences before and after the sentence S. It is limited, either by the beginning of a textual section or by textual organisation clue.
- The context C4 must be explored to identify textual segments formatted with special cues, such as bulleted or enumerated lists.

Rules are organised in tasks. The rule from the above example is part of the task *Static Relations*, collected by Le Priol [26]. This task specifies a set of 238 rules, 6149 markers and 1777 clues in order to assign a set of 14 semantic labels, corresponding to the semantic values defined as static relations in the Cognitive and Applicative Grammar. So far, the CE method has been applied to domain knowledge extraction, thematic announcements extraction, automatic summarisation [27], filtering of causality relations [28], relations between concepts [26] and aspecto-temporal relations [29]. The CE linguistic repository has also been used for Information Extraction under different semantic viewpoints in [30].

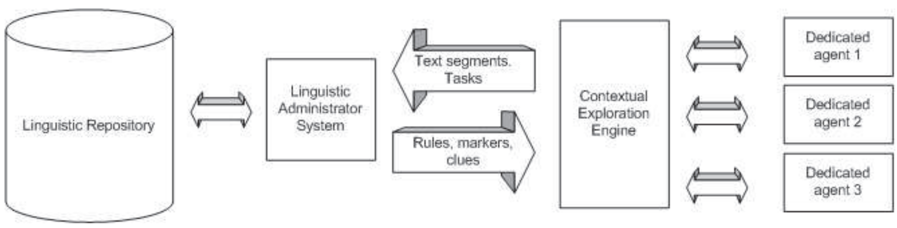

Fig. 1. A Contextual Exploration System [31]

NL processing in CE systems (see Fig. 1) differs from the typical grammar + lexicon approach in that no syntactic or lexical analysis is performed. However, morphological variations are coded as part of CE classes in the linguistic repository, which stores only domain independent linguistic knowledge. The CE engine executes rules associated to a certain task on a text collection. The results are processed and presented to the user by dedicated agents. Domain knowledge extraction, automatic summarisation and filtering of causal relations are examples of a dedicated agents. So far, two CE systems have been developed: ContextO [32] and Semantext [33]. Both of them support the CE cycle. Yet, Semantext differs from ContextO in that, in sake of performance, it analyses ext as a flat chain of characters, while ContextO processes it in a hierarchical way (title, section, subsection, paragraph, etc).

4 CE for requirements analysis and validation

4.1 Semantic Viewpoints

The purpose of applying semantic viewpoints to large TRD is, on the one hand, to improve readability of large TRD by drawing together a small number of statements that convey certain semantic notions. On the other hand, this approach aims at extracting relevant statements from the TRD to shape a new abridged text unit that, because of its size and relevance, would facilitate further processing, like automatic generation of conceptual schema or semantic validation. This approach is intended to extract requirements that are far away from each other in the TRD document, but share common semantic properties. This approach is similar to Natt och Dag's et al. requirements clusters [15] because selected statements could be considered a "semantic cluster", but it differs in that the "clustering" criteria are not lexical, but semantic.

However, extraction is made using relevance criteria, and relevance, as it has been pointed out by Minel [34], depend for the most part on the reader's point of view. For instance, a reader interested in data modelling would certainly pay particular attention to "part-of" relations between domain entities. A relevance criterion for this kind of reader could be to filter only those statements where there are linguistic markers of a "part-of" relation, like the following (from a TRD of an insurance system):

(1) *A joint policy* ***includes*** *the joiners age, the joiners gender and his smoker condition.*

(2) *The agent displays billing mode, effective date, due date, bill number and total premium amount, which* ***are part of*** *the policy's billing detail.*

(3) ***For each*** *mandatory rider, the agent should specify the following values:*
- *Face amount*
- *Due period*
- *Increased Periods*

According to CE rules, this sentences would have the INGREDIENCE semantic label, even if the linguistic markers (in bold) from (1) and (2) look stronger than those of (3), where the part-of relation is not explicitly indicated, but its plausibility may be confirmed by non-lexical clues, like the typographical sign ":" or the presence of a list.

Therefore, from the requirements engineering point of view, the following viewpoints are considered as semantically relevant:

I Relations between concepts
II Aspecto-temporal organization
III Control
IV Causality

Each one of this viewpoints conveys a semantic concepts that, according to the GAC linguistic model, may structure and organise meaning [24], and each one represents an important aspect in requirements analysis as well. Much effort have been devoted to build conceptual schema from TRD's relations between concepts for requirements analysis ([7, 19, 11]). The value of extracting events and processes (II) temporal organisation (the "dynamic" aspect) for NL requirements validation has been remarked by Burg [13]. Control issues (III), i.e. specifying if actions are "environment controlled" or "machine controlled", are of primary importance in requirements engineering [35], and a precise understanding of the causal organisation (IV) of actions is necessary to specify the rules that a system must obey [36]. The following example shows different views of one TRD paragraph according to these semantic viewpoints. Relevant sentences are marked in bold. Under-braces indicate semantic values assigned by CE rules according to linguistic markers and clues (underlined)[4]:

- Relations between concepts viewpoint:

  When the start button is pressed, if there is an original in the feed slot, the photocopier makes N copies of it, and places them in the output tray. ***N*** $\underbrace{\text{\underline{\textit{is the}}}}_{\text{(EQUALITY)}}$ ***number currently registering in the count display.*** *If the start button is pressed while photocopying is in progress, it has no effect. The number N in the count display updates in response to button pressed according to the state table.*

- Aspecto-temporal organisation viewpoint:

  <u>When</u> the start button $\underbrace{\text{\underline{\textit{is pressed}}}}_{\text{(EVENT)}}$***, <u>if</u> there <u>is</u> an original in the feed slot, the photocopier <u>makes</u> N copies of it, and <u>places</u> them in the output tray.*** *N is the number currently registering in the count display.* ***<u>If</u> the start button <u>is pressed while</u> photocopying <u>is in progress</u>, it <u>has</u> no effect. The number N in the count display <u>updates</u> in response to button pressed according to the state table.***

- Control viewpoint:

[4] This example is taken from a requirements document from Kovitz [36]

When the start button *is pressed* (ENVIRONMENT CONTROLLED), if there is an original in the feed slot, the photocopier makes (MACHINE CONTROLLED) **N copies of it, and places them in the output tray**. N is the number currently registering in the count display. If **the start button is pressed** while photocopying is in progress, it has no effect. **The number N in the count display updates in response to button pressed according to the state table.**

- Causality viewpoint:

  When the start button is pressed, ***if there is an original in the feed slot, the photocopier makes N copies of it****, and places them in the output tray. N is the number currently registering in the count display.* ***If the start button is pressed while photocopying is in progress, it has no effect****. The number N in the count display updates in response to button pressed according to the state table.*

4.2 Supporting RE analysis and validation with semantic viewpoints

Every semantic viewpoint produce a two-fold output: the extracted statements and their associated semantic values. Our first hypothesis is that the extracted TRD statements, organised by semantic viewpoints, can improve readability of the overall TRD documents. Our second hypothesis is that semantic values can support requirements validation, especially of large TRD, by means of CE-based rules. The following are examples of how CE rules could be applied on requirements validation:

- To apply analytic rules based on "Relation between concepts" semantic values in order to verify, for instance, that in INGREDIENCE (part-of) statements, the relationship between an element and its parts does never gets reversed.
- To look for conflicts between viewpoints. For instance, suppose that the following statements, taken from an insurance system TRD, are filtered by control and cause viewpoints:

  (page 60) *When the product's life-cycle is over, the system should trigger a premium-collection event.*

  (page 234) *The system can prevent a premium-collection event but only an agent can cause it.*

  In the statement of page 234, filtered by the causality viewpoint, establishes that only an agent can *cause* a premium-collection event, while in page 60's statement (control viewpoint) there are traces of a machin-controlled situation over the premium collection event, where the system *triggers* the event.
- To look for semantic patterns. For instance, a high proportion of cause and control statements may be sign of a Jackson's control problem frame [4].

The activities that would allow to incorporate the CE method to RE processes are shown in Fig. 2. First, the configuration of linguistic resources is

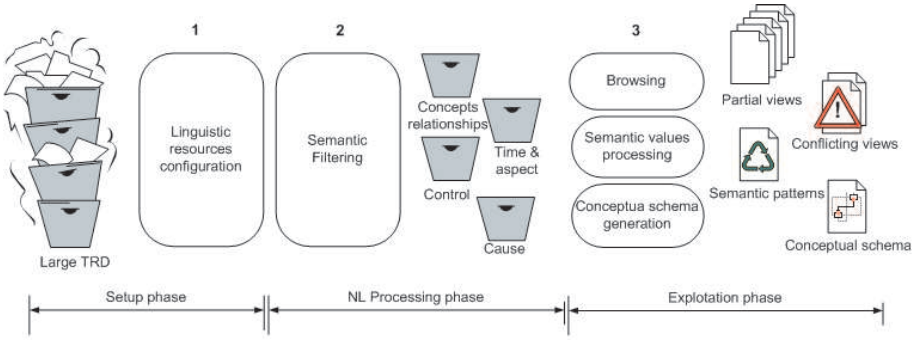

Fig. 2. TRD filtering cycle

needed. This setup phase is intended assimilate the application's domain glossary into the linguistic resources and to do fine-tuning on CE rules in order to adapt them to the style and the language of TRD. Then, TRD documents would be filtered according to semantic viewpoints. An interactive presentation of the partial TRD views is required in order to browse the TRD and make a first evaluation on the filtering quality. Conflicting views and semantic values analysis need a set of CE-based rules, that would allow inter-viewpoint analysis based on semantic values.

5 Conclusion and further work

This paper has proposed a semantic-based approach for NLRE, which extracts semantically relevant sentences from large TRD making use of linguistic-based rules. It has exposed the CE method, which organises rules, linguistic markers and clues, assigning semantic values according to four major viewpoints, considered as relevant from a requirements engineering perspective: relations between concepts, aspecto-temporal organisation, control and causality. Furthermore, this paper has outlined how semantic viewpoints could improve requirements analysis and validation in light of other NLRE approaches.

Currently, work is being done to evaluate the precision of CE rules, markers and clues (most of them issued from linguistic studies on scientific corpus) on industrial requirements documents, as well as on the implementation of a declarative language for semantic-value based rules in a way that could allow inter-operability between viewpoints. An evaluation will be necessary, in order to know to which extent CE improve large TRD readability and, on the other hand, if the proposed semantic viewpoints facilitate requirements validation. Based on evaluation methods of the CREWS/L'écritoire project [21] and automatic summaries evaluation experiences [34], we can conclude that empirical evaluation would be necessary, where system analyst could use the proposed method on real TRD.

Acknowledgements

Thanks to Camille Salinesi for his feedback about this work.

References

1. Mich, L., Franch, M., Novi Inveradi, P.: Requirements analysis using linguistic tools: Results od an on-line survey. Technical report 66, Department of Computer Management Sciences. Università de Trento, http://eprints.biblio.unitn.it/view/department/informaticas.html (2003)
2. Jacobson, I.: Object Oriented Software Engineering: A use case approach. Addison-Wesley, États-Unis (1992)
3. Goguen, J.: Requirements engineering as the reconciliation of social and technical issues. In Goguen, J., Jirotka, M., eds.: Requirements Engineering: Social and technical issues. Academic Press, London (1994)
4. Jackson, M.: Software Requirements and Specifications : A Lexicon of Practices, Principles and Prejudices. ACM Press, New York (1995)
5. Desclés, J.P.: Systèmes d'exploration contextuelle. In Guimier, C., ed.: Co-texte et calcul de sens. Presses Universitaires de Caen, France (1997) 215–232
6. Chen, P.: English sentence structure and entity-relationship. Information Systems (1983) 127–149
7. Rolland, C., Proix, C.: A natural language approach for requirements engineering. In: Advanced Information System Engineering (Lecture Notes in Computer Science). Springer Verlag, Paris (1992)
8. Macias, B., Pulman, S.: Natural language processing for requirements specifications. In: Safety Critical Systems, Current Issues, Techniaues and Standards. Chapman and Hall (1993) pp. 67–89
9. Goldin, L., Berry, D.: Abstfinder, a prototype natural language text abstraction finder for use in requirements elicitation. Automated Software Engineering (1998) 375–412
10. Osborne, M., MacNish, C.: Processing natural language software requirement specifications. In: 2nd IEEE International Conference on Requirements Engineering, IEEE Press (1996)
11. Ambriola, V., Gervasi, V.: Processing natural language requirements. In: 12th International Conference on Automated Software Engineering, Lake Tahoe, Etats Unis, IEEE Press (1997) 36–45
12. Ben Achour, C., Rolland, C.: Introducing genericity and modularity of textual scenario interpretation in the context of requirements engineering. Report 97-15 (http://sunsite.informatik.rwth-aachen.de/crews/), CREWS (1997)
13. Burg, J.: Linguistic Instruments in Requirements Engineering. IOS Press, Amsterdam (1997)
14. Rayson, P., Garside, R., Sawyer, P.: Assisting requirements engineering with semantic document analysis. in proceedings of content-based multimedia information access. In: Proceedings of Content-based multimedia information access RIAO 2000, Paris (2000)
15. Natt och Dag, J., Regnell, B., Carlshamre, P., Andersson, M., Karlsson, J.: Evaluating automated support for requirements similirity analysis in market-driven development. In: Seventh International Workshop on Requirements Engineering: Foundation for Software Quality, Switzerland (2001)

16. Fabbrini, F., Fusani, M., Gnesi, S., Lami, G.: An automatic quality evaluation for natural language requirements. In: Seventh International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'2001). (2001)
17. Gervasi, V., Nuseibeh, B.: Lightweight validation of natural language requirements. Software Practice and Experience (2002) 113–133
18. Fantechi, A., Gnesi, S., Lami, G., Maccari, A.: Applications of linguistic techniques for use case analysis. Requirements Engineering Journal **8** (2003) pp. 161 – 170
19. Fliedl, G., Kop, C., Mayr, H.: From scenarios to kcpm dynamic schemas: Aspects of automatic mapping. In: Proceedings of the 8th International Conference on Applications of Natural Language and Information Systems (NLDB'2003), Germany, GI Edition (2003) 91–105
20. Nuseibeh, B., Easterbrook, S.: Requirements engineering: A roadmap. In Finkelstein, A., ed.: The Future of Software Engineering. ICSE, Londres (2000)
21. Ben-Achour, C.: Extraction des besoins par analyse de scénarios textuels. PhD thesis, Université de Paris 6 (1999)
22. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K., Tengi, R.: Five papers on wordnet. Technical report, Cognitive Science Laboratory, Princeton University, New York (1993)
23. Fillmore, C.: The case for case. In Harms, B., ed.: Universals in linguistic theory. Holt and Rinehart and Winston, Chicago (1968)
24. Desclés, J.P.: Langages applicatifs, langues naturelles et cognition. Hermes, Paris (1990)
25. Minel, J.L.: Filtrage Sémantique. Hermes, Paris (2002)
26. LePriol, F.: Extraction et capitalisation automatiques de connaissances a partir de documents textuels. PhD thesis, Université Paris-Sorbonne (2000)
27. Berri, J.: Contribution à la méthode d'exploration contextuelle. Applications au résumée automatique et aux représentations temporelles. Réalisation du Système SERAPHIN. PhD thesis, Université Paris-Sorbonne (1996)
28. Jackiewicz, A.: L'expression de la causalité dans les textes. Contribution au filtrage sémantique par une méthode informatique d'exploration contextuelle. PhD thesis, Université Paris-Sorbonne (1998)
29. Chagnoux, M., Ben Hazez, S., Desclés, J.P.: Identification automatique des valeurs temporelles dans les textes. In: 10e conférence sur le traitement automatique des langues (TALN' 2003). (2003)
30. Laublet, P., Nait-Baha, L., Jackiewicz, A., Djioua, B.: Interaction homme-machine et recherche d'information. In: Collecte d'information textuelles sur le Web selon différents points de vue. Paganelli, c. edn. Hermes, Paris (2002)
31. Minel, J.L., Cartier, E., Crispino, G., Desclés, J.P., Ben Hazez, S., Jackiewizc, A.: Résumé automatique par filtrage sémantique d'informations dans des textes. présentation de la plate-forma filtext. Technique et Sciences Informatiques (2002)
32. Crispino, G.: Une plate-forme informatique de l'Exploration Contextuelle : modélisation, architecture et réalisation (ContextO). PhD thesis, Université Paris-Sorbonne (2004)
33. Ben Hazez, S.: Un modèle d'exploration contextuelle des textes: filtrage et structuration d'informations texttuelles, modélisation et réalisation informatique (système SEMANTEXT). PhD thesis, Université de Paris-Sorbonne (2002)
34. Minel, J.L., Nugier, S., Piat, G.: How to appreciate the quality of automatic text summarization? examples of fan and mluce protocols and their results on seraphin. In: Workshop Intelligent Scalable Text Summarization EACL 97. (1997) 25–30
35. Zave, P., Jackson, M.: Four dark corners in requirements engineering. ACM Computing Surveys **6** (1997) 1–30

36. Kovitz, B.: Practical Software Requirements. Manning, London (1998)