Using natural language processing tools to assist semiotic analysis of information systems

Ken Cosh and Pete Sawyer
Computing Department
Lancaster University
UK LA1 3EU
[k.cosh, sawyer]@comp.lancs.ac.uk

1. Abstract

Semiotic Analysis has been used to aid understanding of information or communication systems, providing information that can be used during requirements engineering. The MEASUR approach begins by analysing short, natural language problem statements and manually extracting the key themes involved. As the process is scaled up and applied to longer problem statements, as found in many real life circumstances, the manual effort required increases. When the starting point for Semiotic Analysis is a large document describing the information system, such as an ethnographic report, assistance in the analytical process is necessary. This paper investigates how statistical Natural Language Processing Tools can aid this analysis.

Natural Language Processing Tools can assist the analyst by directing them to the central themes in the document. Comparing a frequency list of the document with a frequency list from a large corpus of text such as the British National Corpus reveals the key words in the document. Collocation analysis of these keywords enables the creation of a lexical network and then closer investigation of the collocates in context allows the analyst to add semantic information to the model.

2. Keywords

Natural Language Processing, Semiotic Analysis, MEASUR, Requirements Engineering, Organisational Semiotics

3. Introduction to semiotic analysis

Semiotics is the study of 'signs', and how they are used to communicate information between people. Organisational Semiotics is merely the study of how these signs are used within organisations(Stamper 2000). As a sign can be anything that conveys information, understanding the properties and meanings of these signs can be a useful aid to understanding the workings of the organisation. One application of semiotics is the semiotic analysis of information systems. This can be used to aid requirement engineering.

Requirement Engineering is used to analyse a problem domain, prior to designing a solution to the problem. It is a crucial part of any software engineering process, as it is vital to have a thorough understanding of the problem before attempting to solve it. Ambiguity and misunderstandings between the user and the developer are a big cause of costly rework. Analysing a problem using semiotics can reduce ambiguity by creating a common understanding of the semantics involved in a problem domain.

This paper reviews the MEASUR approach to Semantic Analysis, developed by Stamper (Stamper 1994), and looks at the similarities between steps in MEASUR and some statistical Natural Language Processing techniques. It looks at some problems that currently exist with this Semantic Analysis approach and discusses how some statistical Natural Language Processing tools can be used as a solution to these problems.

Several authors have pointed out the apparent links between Organisational Semiotics and Natural Language Processing (NLP). (Charrel 2002)(Connolly 2000). Whenever Semiotic Analysis is used to model any problem or domain, then natural language, whether spoken or documented, is studied. The purpose of this research is to demonstrate how probabilistic NLP techniques can be applied to Semantic Analysis, as described in the next section.

4. The MEASUR approach to semantic analysis

There are 4 major phases to Semantic Analysis as proposed by MEASUR (Lui 2000):

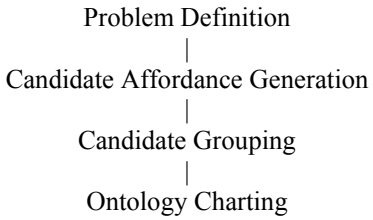

Figure 1, Stages in semantic analysis

The first step, as shown, is problem definition. MEASUR begins by formulating a concise, well-articulated problem statement, which includes all the relevant parts of a problem.

Working from this problem description, semantic analysis takes over, with the goal of creating a model of the problem. This begins with candidate affordance generation, where all the agents, objects, actions, etc. are identified from the text. Once these candidates have been identified, Candidate Grouping takes over as the first step towards creating an ontology chart. The two key types of entity that need to be identified during candidate affordance generation, to be used as construction blocks for the Ontology Chart, are *agents* and *affordances*. An agent is a type of object, a performer or processor, the initiator of an event. An agent can be a human, a device or a program, whatever has responsibility for the action. The behaviour of an agent is directed by its knowledge of, and constrained by the nature of, the environment.

For instance an agent could swim assuming it has the knowledge of how to swim, and the environment affords it an area of liquid to swim in. A swimming pool could therefore be seen as an affordance within the environment enabling the agent to swim. Affordances can be seen as properties of a situation, not necessarily objects such as swimming pools. Other examples of affordances are budgets, projects, time and even agents, as sometimes agents can become affordances of another agent, depending on context. (Gibson 1979)(Stamper 1996)

Agents are relatively easy to identify since they typically appear as nouns in the problem statement. Nouns may also represent affordances or roles (roles are explained in the conference organisation example below). Unlike agents (and roles), affordances do not map neatly onto a single part of speech. They can be nouns, but equally could be verbs or several other parts of speech. Hence, while a list of candidate agents may be generated mechanistically, the identification of affordances requires analysis of the semantics by considering the relationships between the various syntactic elements of the problem statement. Normally elements will depend on other elements for their existence. For instance, the affordance *swim* depends upon the existence of a *swimming pool*, in which to swim, and also a *swimmer* (the role of a person agent), to actually swim. This is depicted in the fragment of an *ontology chart* in figure 2. Here, agents are shown as ellipses, and affordances are depicted in rectangles.

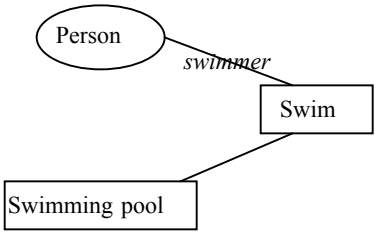

Figure 2. A fragment of an ontology chart

The ontology chart is one of the goals of this process; it is a graphical representation of the relationships between these agents and affordances. The process to create an ontology chart is better demonstrated in the example below. The eventual chart is comprised from grouping the affordances together. Once the ontology chart has been created, the next step is to assemble a set of norms, which govern the standard behaviour of the model. Norms describe how the model is expected to work. They describe the normal behaviour of agents within the problem domain and their use of affordances. When the norms are attached to an ontology chart a thorough understanding of the problem is completed. This paper however concentrates upon the steps involved in creating the ontology chart.

5. Conference organisation example

The following minimal problem statement permits the identification of the principal entities of the problem domain listed in table 1.:

A member of an organisation can organise a conference, which they invite people to attend. Participants can submit papers to be reviewed by the conference organiser.

Member	Role of person in organization
Organisation	Agent
Organise	Affordance of conference and conference organizer
Conference	Affordance of organization
Invite	Affordance of conference organiser and person
People	Agent
Attend	Affordance of participant
Participant	Role of person who accepts invitation
Submit	Affordance of Participant and Paper
Papers	Affordance
Review	Affordance of conference organiser and contributed paper
Conference Organiser	Role of member who organises conference

Table 1. The principal entities in a conference organisation problem

The above list of entities represents the set of candidate nodes in the ontology chart. Once the candidates have been identified, the next task as also illustrated in table 1, is to categorise these as agents, roles and affordances and assemble the ontology chart to explicate the relationships between them. In this example, *Organisation* is an agent (agents aren't necessarily human) while *Member* is a role of a person within the organisation. In order to make the relationship understandable, the affordance *Membership* has to be added. Membership depends upon an Organisation, which can allow Membership, and a Person, to take advantage of Membership, whereupon they become a Member (Figure 3).

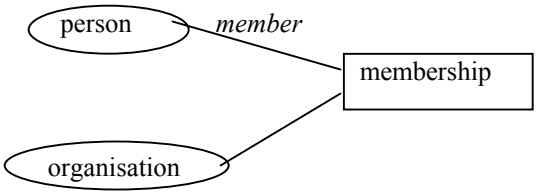

Figure 3. Modelling the membership of an organisation

In the model in figure 3, o*rganisation* and *person* are agents, while *membership* is an affordance. *Member* is a role and so is placed between person and membership. The important entities of the problem domain can be grouped to begin to create an ontology chart. Once candidates have been grouped a structure for the chart appears.

Member has been identified as a role, not just an agent, as the problem statement identifies People, Participants and Members. An experienced analyst will recognise that Participant and Member are (in O-O terms) specialisations of Person. In an O-O notation such as UML class diagrams, the analyst would have the option of modelling these using either sub-classing or, if instances of a single class could play multiple roles, using roles at the termination points of association relationships. In an ontology chart, roles are the modelling mechanism used.

Roles only exist in circumstances where a dependent affordance is taken advantage of, so they are depicted along the arc between the antecedent and the dependent. For instance, in figure 4 which shows the complete ontology chart for the conference organisation problem, a p*erson* only becomes a p*articipant* when they accept the invitation. Roles in an ontology chart are not merely labels, they can form nodes. This is illustrated in the c*onference organiser* role which is itself derived as a special type of the role m*ember* when o*rganising* the c*onference*. Note that roles needn't be restricted to the agents in the model. For

example, c*ontribution* is a specific type of the affordance p*aper* which has been submitted by the role p*articipant*.

In the same way in which *membership* was added to the ontology chart to explain the relationship between a *person* and an *organisation*, other terms can be added. *Authorship* and *accept* are added to the ontology chart despite not being in the original problem statement. *Authorship* explains the relationship between a *person* and a *paper*. *Accept* has to be added to enable the existence of a *participant* as a *person* only becomes a *participant* when they have accepted the invitation to the *conference*.

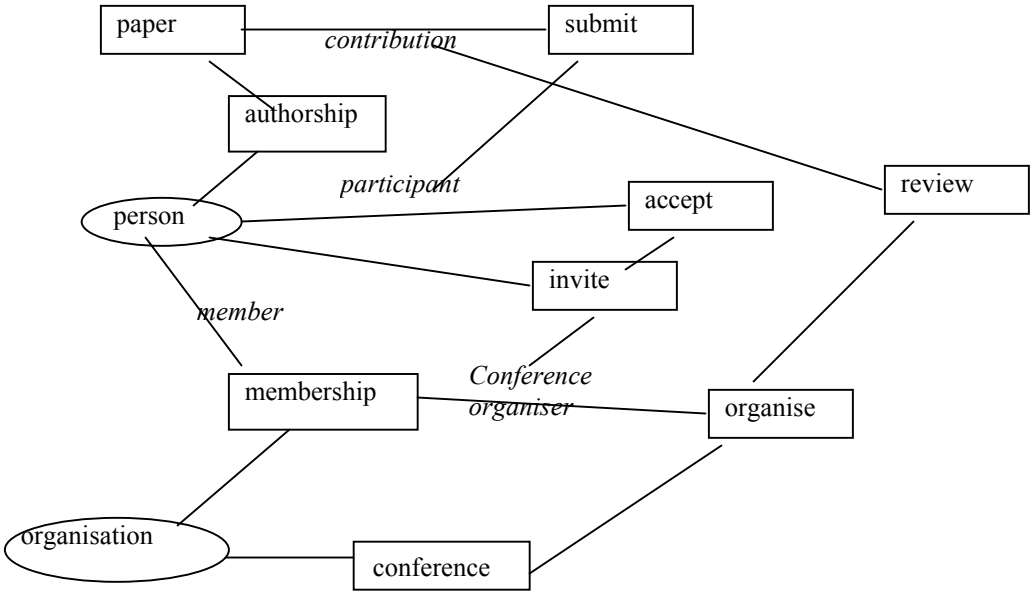

Figure 4. Complete ontology chart of conference organiser example

An ontology chart should be read from left to right, as objects to the right are dependent on the items they are connected to by arcs on their left (Barjis & Chong 2002). For example, the affordance invite can only exist when there is a conference organiser (to perform the inviting) and a person (to be invited). Similarly the affordance accept relies on a person (to perform the accepting) and the existence of the invitation (to be accepted).

The construction of an ontology chart, even from a concise problem description, isn't mechanistic. It requires considerable experience and skill in the application of MEASUR on the part of the analyst and this acts as an inhibitor to the application of MEASUR in systems development.

6. Statistical natural language processing and organisational semiotics.

In all the documented case-studies and examples used in the organisational semiotics literature, the problem scope is small, and the problem statement is a concise description with no redundant description or ambiguity. For many real life cases it isn't possible to neatly and perfectly summarise the problem in a brief problem statement, so the starting point for semantic analysis may not be a carefully worded problem statement. Instead it could be a set of long descriptive reports. Examples could include ethnographic study reports, legal documents, codes of practice, meeting transcripts, and any other documentation which could add knowledge to the problem definition. Many problems are just too complex and subtle to be neatly encapsulated by a short problem statement. To be practical it should be possible to apply MEASUR to problems where the sources of information are less clearly bounded, diffuse, scattered and poorly structured.

The problem with this is that to read through and analyse long documents, using the approach discussed above, becomes increasingly difficult, particularly if we intend to "generate candidate affordances", by selecting every noun, noun-phrase, verb etc. While in a short precisely written problem statement the analyst will only find the necessary semantic units in order to create the ontology chart, in a longer larger document, the analyst might find many nouns which only occur once throughout the document and don't actually add anything to the model. This irrelevant information needs to be filtered out.

Many of the terms will be used several times, in different contexts, so they become easier to define accurately. Having selected the key candidates using the frequency tests, we can then look at these words in context, singling out the word and the passage in which the word occurs. Looking at the key word in context (KWIC), as well as aiding definition of terms, can also be used for grouping the keywords together, as instances where keywords are used in conjunction with each other can be isolated.

As large amounts of natural language has to be analysed, it is logical to use some NLP tools to assist with the automation of the approach. The following section discusses how some NLP techniques can be used to aid with Candidate Affordance Generation and Candidate Grouping.

7. Applying natural language processing techniques

For the purposes of this example, information collected in an ethnographic report into the Air Traffic Control (ATC) domain is used. This has previously been looked at in the REVERE project (Rayson, Garside & Sawyer 2000). The document is 66 pages long with over 40,000 words in it, so it clearly isn't the concise carefully written problem statement alluded to previously.

Using this as the 'problem statement', the next step is to generate a list of candidate affordances. Using the statistical NLP tools this can be done by firstly compiling a frequency list – counting the number of occurrences for each word. This frequency list can then be compared to a Corpus frequency list. A Corpus is a large body of text. The example used here is the British National Corpus (BNC), which is a giant frequency list consisting of many words from the English language, and how frequently they can be expected to occur.

The first step in this process, is to create a frequency list for each word in our document. Initially the most frequently occurring words are likely to be words such as 'the', 'of', 'and' and 'to'. These words aren't a particularly helpful indication of what the document is about, as they are the words which occur most commonly in written natural language. The most interesting and helpful words are those which occur most significantly more often than we would expect them to within the document, and we can detect these by comparing the document with the BNC.

The first step in calculating the most significantly overused words is to calculate how frequently we would expect a word to occur, given the size of the document. This can be done by first creating the following contingency table;

	Text to be analysed	BNC	Total
Frequency of word	a	b	A + b
Frequency of other words	c-a	d-b	C + d – a - b
Total	c	d	C + d

Figure 5. Contingency table. (Rayson & Garside 2000)

With the information in this table, the expected frequency for any word in the text to be analysed can be calculated using the following formula;

$$E1 = c * (a + b) / (c + d)$$

And the expected frequency, given out text to be analysed, for any word in the BNC can be calculated;

$$E2 = d * (a + b) / (c + d)$$

Once the two expected frequencies have been calculated, we can calculate the significance in the difference between these two scores, using a Log Likelihood test. The following formula will give a significance score showing how significant it is that the word occurs as frequently as it does;

$$Sig = 2 * ((a * \ln (a / E1)) + (b * \ln (b / E2)))$$

(Rayson & Garside 2000)

The higher the result of this test, the more significant it is that the word has occurred more often than it should have. After this comparison using a log likelihood test between the BNC frequency list, and

the ATC frequency list, a new frequency list is created with the most significantly overused words prominent. These words become the keywords of the problem domain. Using the example of the air traffic control ethnographic report, the words that are most significantly overused within the document are, as could be expected, words like, controller, radar, flight etc. (fig 6). Further information about this technique can be found in (Rayson, Garside & Sawyer 1999).

	Word	Frequency	Log Likelihood
1	controller	217	1386.84
2	strips	227	1372.39
3	radar	168	1060.39
4	strip	173	966.805
5	flight	113	576.605
6	controllers	73	458.849
7	chief	114	451.441
8	sector	82	353.931
9	pole	75	334.929
10	traffic	74	307.131
11	of	671	294.06
12	planes	56	293.969
13	pending	48	281.001
14	aircraft	58	279.465
15	hill	75	273.275
16	level	106	255.248
17	the	1848	254.816
18	airspace	41	246.31
19	ph	40	239.917
20	im	38	239.149

Figure 6. Significantly overused words in the ATC example.

This list still includes words like “the” and “of” which are significantly overused within the document, but don’t add to the model. These words can be removed by refining the list so we only have words that we are interested in.

Clearly with a large document to attempt to draw out the candidate affordances manually would be very time consuming. Using NLP tools, the first steps of the semiotic analysis method can be completed more speedily, and with less manual input. The third stage of the process is to group candidate affordances. Once a list of the terms that should occur within our ontology chart has been generated, they can be grouped. Once again there are NLP tools to assist during this stage.

By looking at each keyword in more detail, it is possible to discover what it means and how it relates to the other keywords more precisely. The overused keyword ‘controller’ is chosen, it is a role, played by a person agent.

Collocation analysis is a statistical test, which produces a Z score that tells us how likely it is for two words to have “co-occurred”. This works by first predicting the number of times that the second word should occur within a specified range, bearing in mind the frequency of the word within the entire document. Given the expected co-occurrence frequency and the actual co-occurrence frequency, the probability can be calculated. With this probability, the significance of the co-occurrence can be tested using Berry Rogghe’s z-score calculation (Oakes 1998).

To calculate the significance of a collocation, the following information is needed;

Z – *total words in text*
A – *number of times keyword occurs in text*
B – *number of times potential collocate occurs in text*
K – *number of times the keyword and the collocate co-occur within span*
S – *Span – number of words on either side of the keyword to be considered*

The first step is to calculate the number of times the collocate should co-occur near the keyword if the two words were randomly distributed and then compare this with the actual number of co-occurrences. To calculate the expected number of co-occurrences, we first need the probability of the collocate occurring where the keyword does not occur;

$$P = B / (Z - A)$$

Then, the expected number of co-occurrences is given by;

$$E = P * A * S$$

The statistical test to see how significant the collocation is, is determined by calculating a z score, as follows;

$$z = (K - E) / \sqrt{(E * (1 - p))}$$

Once again, the higher the z score, the more significant the collocation.

Setting the span to 10 words either side of our keyword, 'controller', the collocation significance of each other word in the document can be calculated. Once we have the z score, as given by Berry Rogghe's calculation, we can compare this to a percentage significance level. Here, words have been split into 1%, 5% and 10% significance,1% being the most significant. Words are significant at the 1% level with a z score greater than 2.33, at the 5% level when greater than 1.65 and at 10% level when greater than 1.3. The most significant co-occurring words (collocates) for 'controller' are;

Word	Frequency	# Co-occurances	Expected Co-occurances	Z Score
Roger	14	14	1.54	10.03
midland	17	14	1.87	8.86
Ph	40	19	4.41	6.96
Upper	4	5	0.44	6.87
131	2	3	0.22	5.92
Fault	2	3	0.22	5.92
Asks	2	3	0.22	5.92
Wit	1	2	0.11	5.69
32	1	2	0.11	5.69
looming	1	2	0.11	5.69
105	1	2	0.11	5.69
arrived	1	2	0.11	5.69
searching	1	2	0.11	5.69
charters	1	2	0.11	5.69
Fallen	1	2	0.11	5.69
Leans	4	4	0.44	5.36
10	17	9	1.87	5.21
Who	48	17	5.29	5.1
assistant	9	6	0.99	5.03
Pilot	35	13	3.85	4.66

Figure 7. Significant collocates at 1% level

As can be seen several of the keywords only occur once within the entire document, so it isn't that interesting that they co-occur with 'controller'. As they only occur once within the document, they add little to the eventual model, so we can filter the list and investigate further the interesting co-occurrences.

Word	Frequency	# Co-occurances	Expected Co-occurances	Z Score
Roger	14	14	1.54	10.03
midland	17	14	1.87	8.86
Ph	40	19	4.41	6.96
Upper	4	5	0.44	6.87
Leans	4	4	0.44	5.36
Who	48	17	5.29	5.1
assistant	9	6	0.99	5.03
Pilot	35	13	3.85	4.66
Red	5	4	0.55	4.65
marks	3	3	0.33	4.64
express	3	3	0.33	4.64
Him	78	22	8.59	4.58
climbing	24	10	2.64	4.53

Figure 8. Significant collocates of 'Controller'

Given the list of significant collocates for each keyword a lexical network could be created automatically. This involves simply connecting every word to those which occur near them. However, as this network contains little semantic content it isn't very useful in understanding the problem domain. Further investigation into each collocate produces much more valuable insight into the nature of the problem.

Each of these significant collocates can be looked at in context, firstly so the meaning of the second word can be understood, and secondly so the relationship between the two words can be analysed. In this example, "roger" is the most significant collocate, and further investigation of the pair in context reveals that the controller communicates to a pilot via radio, and regularly uses the radio term "roger". From this information it is possible to begin to group possible candidate affordances, such as that radio is dependent upon pilot and controller for it's existence.

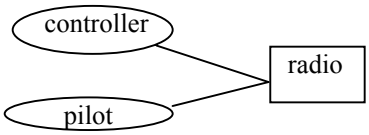

Figure 9. Grouping of controller, radio and pilot.

Looking at keywords in context with other keywords assists the analyst in understanding the semantics of them. By isolating the controller keyword it is possible to find sentences such as;

It is the job of the radar or sector	**controller**	To coordinate this traffic through his/her sectors.

Which amongst others provides the information that the controller is a role performed by a human agent. This information can be added to the growing ontology chart;

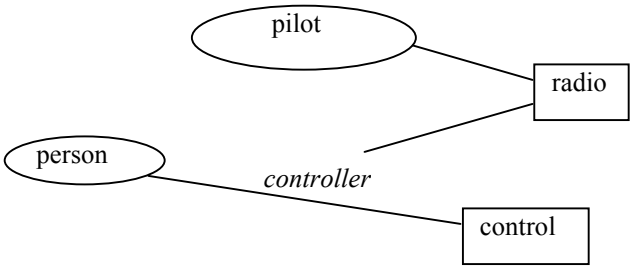

Figure 10. Fragment of ontology chart demonstrating the role controller.

It is also possible to learn further information about the controller role. Firstly from the above sentence alone, a function of the controller role is to co-ordinate traffic through his or her sector. There is clearly a relationship between the controller and 'sector'. Using NLP to look at keywords in context with each other, it is possible to isolate every piece of text which contains both 'controller' and 'sector'. This provides the information needed to discover that it is a sector which the controllers control, or in other words, the affordance control can only exist when there is a controller (to perform the controlling) and a sector (to be controlled). This information can be added to the ontology chart.

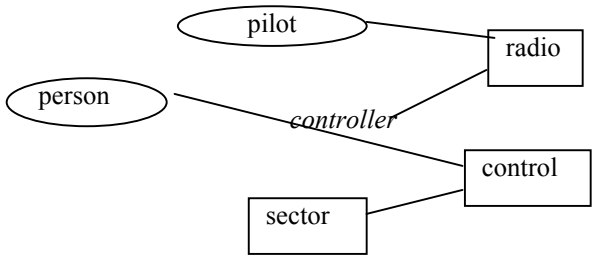

Figure 11. Fragment of ontology chart demonstrating addition of sector.

Gradually by grouping more information, as various parts of the text are investigated using the keywords in context, the ontology chart can be added to until it provides a thorough analysis of the Air Traffic Control domain.

8. Conclusions

Semiotic analysis of an information system can aid requirement engineers and other analysts to fully understand it. When conducting semiotic analysis of an information system a rich source of information could be an ethnographic report, or another natural language document describing the domain. Current approaches to semiotic analysis are designed for use with concise, carefully constructed problem statements. This paper has investigated how NLP can be used to scale up the approach so larger, more information rich documents can be analysed.

The NLP tools that have been included in this research are statistical frequency tests and collocation analysis, aimed at guiding an analyst to the important areas of the document being analysed. Further work in this area could be useful to further aid the analyst in identifying agents and affordances automatically, either by refining Part of Speech tagging or semantic tagging.

Using collocation analysis it is possible to automate the creation of a lexical network, connecting related words based on them occurring near one another in a document. Further human input is necessary to add semantic information to the lexical network. Whilst viewing the keyword in context alongside other keywords is an aid for the analyst, further tool support her could aid the process further.

9. References

Barjis & Chong 2002 Integrating Organisational Semiotic Approach with the Temporal Aspects of Petri Nets for Business Process Modeling, in J.Filipe, Sharp, B. and Miranda, P. (Eds.), *Enterprise Information Systems III*. Kluwer Academic Publishers, Dordrecht, The Netherlands

Charrel 2002 Viewpoints for knowledge management in system design. *In Proceedings 5th Annual International Workshop on Organisational Semiotics*

Connolly 2000 Accomodating Natural Language Within The Organisational Semiotic Framework, *Third International Workshop in Organisational Semiotics WOS3*

Gibson 1979 The Ecological Approach to Visual Perception. *Boston, Houghton Mifflin company.*

Liu 2000 Semiotics in Information Systems Engineering, *Cambridge University Press*

Oakes 1998 Statistics for Corpus Linguistics, Michael P. Oakes. *Edinburgh textbooks in empirical linguistics.*

Rayson, Garside & Sawyer 1999 Language Engineering for the recovery of requirements from legacy documents. *REVERE project report, Lancaster University, May 1999*

Rayson & Garside 2000 Comparing corpora using frequency profiling. In proceedings of the *workshop on Comparing Corpora, held in conjunction with the 38th annual meeting of the Association for Computational Linguistics (ACL 2000)*. 1-8 October 2000, Hong Kong, pp. 1 - 6.

Rayson, Garside & Sawyer 2000 Assisting requirements engineering with semantic document analysis. In Proceedings of Content-based multimedia information access *RIAO 2000 (Recherche d'Informations Assistie par Ordinateur, Computer-Assisted Information Retrieval) International Conference, College de France, Paris, France, April 12-14, 2000.* C.I.D., Paris, pp. 1363 - 1371. ISBN 2-905450-07-X

Stamper 1994 Signs. Information, Norms and Systems *In: B. Holmqvist, P. B. Andersen, H. Klein and R. Posner (Eds), Signs of Work: Semiosis and Information Processing in Organisations*, Walter de Gruyter & Co, Berlin, pp.349-397.

Stamper 1996 Ontological Dependency In *Proceedings of the workshop on "Ontological Engineering" Aug 1996*

Stamper 2000 – Organisational Semiotics Informatics without the Computer? In *Information, Organisation and Technology - Studies in Organisational Semiotics* By Kecheng Liu, Rodney J. Clarke, Peter Bøgh Andersen, Ronald K. Stamper (eds), Kluwer Academic Publishers, Boston, 2001.