

Concept Linking for Information Integration in Open Book and Sentinel

Stuart Watt

School of Computing, The Robert Gordon University
St. Andrew Street, Aberdeen, Scotland. AB25 1HG
S.N.K.Watt@rgu.ac.uk

Abstract

Opening up large amounts of loosely structured information for easy access and use is a complex problem. This paper describes two systems that address different aspects of the problem, but which use common technology and similar underlying principles. Open Book is a web-based newsletter, which automatically links stories with each other, and with database information. Sentinel is an email archiving and pro-active search tool, which weaves email dialog and related documents into a web of story content. Both use a story-based approach to information linking, using genre-based information extraction and semantic networks.

Introduction

One way to think about the problem of knowledge management is to consider it as connecting knowledge – connecting one person’s knowledge to that of other people. Yet this knowledge is likely to be widespread: some may be in documents, some in databases, and a very substantial amount in email or other unstructured communication.

For effective knowledge management, information needs to be combined from this variety of different sources. Some may be structured, some unstructured; some may be slow to change, others with a clear temporal context – and above all, there may be a lot of it!

But to make all this information useful it needs to be made available and accessible to people, and to link and combine information from different sources, opening up the information that people need in a way that supports this access. Some of the key requirements are as follows:

- **Engaging interaction.** Knowledge management cannot be imposed from on high. People need to be encouraged to participate, so the system needs to be usable and effective, but more than this, to actually encourage use.
- **Diverse content.** In particular, information will vary in its size and degree of structure, so there is a need for techniques to link information that can interoperate with information extraction, and which can work effectively with only small amounts of data.
- **Extensible to new information.** Organizations are not

static. New documents – and even new kinds of documents – appear all the time. These need to be linked with existing documents without heavy maintenance.

- **Scaleable to large amounts of information.** The value of knowledge management increases as its scope widens to a point where it becomes a useful resource across an organization. This means that solutions need to be scaleable, often to at least hundreds of thousands of texts.

This paper describes two closely related projects, Open Book and Sentinel, each addressing different aspects of the problem of knowledge management, but using a common rationale and set of theoretical principles.

Design Rationale

The rationale behind the requirements on these two projects, and the design issues and decisions that have influenced their direction, are summarized in Table 1 below. For now, it is worth looking through each of the main requirements in more detail, discussing the rationale that led to the design decisions underpinning both systems.

Engaging interaction through linked stories

One of the primary aims of this work was to avoid a kind of ‘oracle’ (Masterton & Watt, 2000) – developing a system that would be perceived as having *the* answer to any particular problem. Usually, there are many issues that touch on a particular topic of concern, and these alternatives need to be accepted and respected.

A story-based approach was central to this. Stories are engaging, and they don’t impose authoritative interpretation. People take away their own lessons from stories. Furthermore, stories have a point-of-view; they are implicitly a person’s story, and other people’s stories may touch on the same issues and concerns. The requirement here, therefore, was to develop a system that would capture these stories, and index them, linking different people’s stories.

ASK systems (Ferguson, Bareiss, Birnbaum, & Osgood, 1992) are a form of case-based instruction system which are based on a systematically organised network of stories. Stories, which are usually anecdotal, are linked so that from one story, you can get to related stories through questions which follow “communicative/associative category

ries”, or CACs. In fact, ASK systems are designed on a theoretical model of conversational interaction (Schank, 1977) that describes how people move from one topic to another in a conversation. ASK systems allow learners to adopt a limited conversation, navigating through a web of stories by asking follow-up questions. One of the key limitations of ASK systems is the amount of effort involved in their construction. Some approaches to automating the process have been proposed (Cleary & Bareiss, 1996) but no practical implementations yet exist.

ASK systems implement an “Aesopic dialog”, using eight CACs. These fall into four classes as follows:

- **Context** (core questions: ‘what is the big picture?’ and ‘what are the details of this situation?’)
- **Comparison** (core questions: ‘where have similar situations occurred?’ and ‘what other approaches exist?’)
- **Causality** (core questions: ‘how did this situation develop?’ and ‘what is the outcome of this situation?’)
- **Advice** (core questions: ‘how can I build on this situation?’ and ‘what might go wrong?’)

In any ASK system, a story is linked to a number of others, but not directly, through one of these questions. Note that this is not the only set of CACs that can be used: ASK systems are specifically for case-based instruction rather than knowledge management. They do, therefore, to some extent downplay the importance of knowledge about people.

Linking Material from Different Sources

Given a web of stories and information, different elements need to be linked somehow, so that people can browse them effectively. Also, using conversational/associative

categories means that links need to be categorized somehow – not all links are the same.

There has already been some work in automating linking in ASK systems. After exploring and rejecting a number of object-linking approaches which depend by and large only on the objects named in a piece of text, Cleary and Bareiss (1996) advocated an approach called “point linking”. Each of Cleary and Bareiss’s ‘points’ are 5-tuples: $\langle concept1, mode, sense, relation, concept2 \rangle$. The core of a point is a directed relation between two concepts, but a mode and a sense are added which can be used to “twist” the point. The sense allows opposition between different points, for example, ‘X is evidence for A’ and ‘Y is evidence against A’ can be connected using the same relation but a different sense, allowing X and Y to be linked through A. Mode allows the difference between ‘may’ and ‘is’ to be used.

However, while sound in principle, point linking is unimplemented in practice, and instead, linking within both projects was implemented by using a simpler semantic network instead. Semantic networks are old hat in AI, dating back to Quillian’s work in the late 1960s (Quillian, 1968). A semantic network is simply a directed graph, with objects linked to other objects through relations; they are 3-tuples: $\langle concept1, relation, concept2 \rangle$. Being graphs, semantic networks are easy to grow through machine learning: new objects, and even new types of relations, are relatively simple to add without requiring existing networks to be restructured. Also, they do not depend on the sophistication of linguistic processing required to determine mode and sense in point linking, and mode and sense, while ideal for the CACs in a teaching ASK system, are not always needed for effective knowledge management.

Requirements	Issues	Design decisions
Encouraging browsing rather than searching	Integrated material needs to be presented in an engaging and non-authoritative manner	Story-based approach emphasizing knowledge about people
	Searching focuses on precision and recall on a query, and does not support looking for related material except through similarity measures	Automated hyperlinking, linking objects in a systematic manner
Linking material from different data sources	Web, databases, and email vary in their use of structure and meta-data, yet need to be linked in a way that brings out connections between them	Use of a semantic network to store associations between data
New material can be added without significant manual indexing	Rich and highly structured knowledge representations constrain growth through new knowledge As new objects, concepts, and relationships are discovered, they need to be integrated into existing information structures	
Effective use of information in ‘unstructured’ content	Many data sources contain content that is structured through social consensus rather than through technology. This information can be extremely valuable, not least because of community focus	Use of genre rules and patterns for information extraction
Ensuring scalability	Some data sources may be too large to allow in-memory representations	SQL databases for underlying data storage

Table 1. Requirements and design decisions influencing Open Book and Sentinel

There are limitations in the approach, though. Semantic networks make meta-representation difficult; that is, while ‘Ann is an expert on AI’ is easy to represent, ‘Sally thinks Ann is an expert on AI’ is more complex, as relations aren’t objects in their own right. It can be hard to implement the bigger representational shifts that may accompany the acquisition of new concepts. The flexibility of semantic networks has a price, although as a representation of an individual’s semantic memory, they are effective. For information integration, they have many advantages. Information from multiple data sources is easy to connect using a semantic network, and their graph structure makes them easy to grow through machine learning.

It is worth noting that a semantic network differs significantly from an object-oriented approach to integration. For example, a class-based object-oriented model assumes that objects fall into crisply-defined classes according to a set of defining features. Although semantic networks can be used in an object-like way, for example, using classes as nodes, and a special ‘is-a’ relation type, they have an additional depth to their flexibility. Prototype-based object models can be defined just as easily, and they may provide a more accurate model of an individual’s semantic memory, which was the intent behind semantic networks.

Genre-based Information Extraction

Email is very different from the web and other information sources in several important senses. It is usually described as ‘unstructured’, and treated as a sequence of words which can be accessed through a search tool of some kind.

In practice, email is far from unstructured, and this doesn’t just apply to a message’s headers. In any community, there is a tendency for socially constructed communicative behaviors called ‘genres’ to emerge, to improve the efficiency of communal activities (Yates & Orlikowski, 1992). Good examples of genres include reports, memos,

and letters, all of which have distinctive structures which play an essential role in the interpretation of their content. Genres are distinguished by their form (structural aspects of the text, including socially accepted headings and spacing), as well as by their medium and language (such as the ‘voice’ and formality of the text).

There is good evidence that genre can be recognized using rules and patterns, and that an awareness of genre can help with interpretation of the text itself (Collins, Mulholland, & Watt, 2001). This is often omitted in information retrieval and information extraction, which have historically tended to focus on a text as a sequence of words, and has only recently begun to use formatting and other genre evidence to improve effectiveness of processing.

Open Book and Sentinel

Open Book

Open Book was originally envisaged as a way of integrating news stories with a web site. It was first piloted for an open day, where it was set up on a stand with a web cam. Visitors were encouraged to write short stories about their visit, and these stories would be automatically linked to relevant projects and researchers.

The system built on earlier work from KMi Planet (Domingue & Scott, 1998), which was a web-based newsletter program using in an academic department. Open Book different from Planet by being ‘seeded’ with information extracted from two databases, one of people, and the second of projects. Each of these databases set out a number of core properties for each person and each project. Significantly, people and projects were closely linked: projects involved a group of people. News stories in Open Book do not stand in isolation, they are automatically linked to relevant people, projects, and through these, to related stories. The actual interface to Open Book, with some automatic links magnified, is shown in Figure 1.

Taking a hard line on the ‘browsing not searching’ issue, Open Book did not, and never has, implemented a searching tool, although the related work on genre (Collins et al., 2001) was used to develop a search tool for KMi Planet itself. This search tool showed high precision with reasonable recall, and allowed more structured questions (for example, searching within a particular type of story) than a traditional textual information retrieval tool.

News stories show an intriguing property: the news genre often follows an ‘inverted pyramid’ dating from the days of metal type. Put simply, the first sentence summarizes the story, and the first paragraph ‘lead’ completes the summary in a little more detail. The breadth of overview increases as the story continues, and more and more background is drawn in. The story can, therefore, be cut off at almost any point without losing its essence: necessary when cutting a story to make the layout fit meant precisely that! Incidentally, in new stories, the headline is often more or less useless as a guide to the actual content, instead, it is

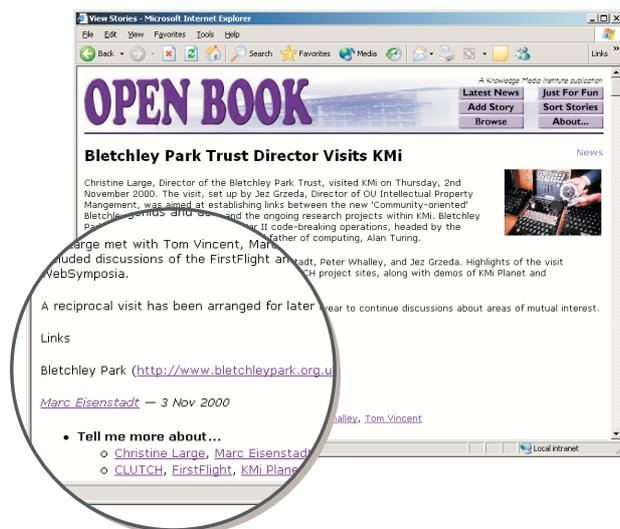


Figure 1. Screen snapshot for Open Book, showing (enlarged) the automatically generated links

usually an ‘attention grabber’, designed to draw people in to reading the rest of the story.

This means that the first sentence of a story is usually an excellent summary of it. Certainly, journalists are trained to get over all the important parts, the ‘who’, ‘what’, ‘where’, ‘when’, ‘why’, and ‘how’, as early as possible. The story below is a good example: all of these are clearly communicated in the first paragraph.

Bletchley Park Trust Director Visits KMi

Christine Large, Director of the Bletchley Park Trust, visited KMi on Thursday, 2nd November 2000. The visit, set up by Jez Grzeda, Director of OU Intellectual Property Management, was aimed at establishing links between the new ‘Community-oriented’ Bletchley Park Trust activities and the ongoing research projects within KMi.

Bletchley Park is the home of the UK World War II code-breaking operations, headed by the mathematical genius and acknowledged father of computing, Alan Turing. Ms. Large met with Tom Vincent, Marc Eisenstadt, Peter Whalley, and Jez Grzeda.

Highlights of the visit included discussions of the FirstFlight and CLUTCH project sites, along with demos of KMi Planet and WebSymposia. A reciprocal visit has been arranged for later in the year to continue discussions about areas of mutual interest.

Open Book uses this deliberately: it pays special attention to the first paragraph, and particularly to the first sentence. By using the genre to focus attention on small parts of the story, rather than trying to laboriously and exhaustively analyze all the background as well as the essence, it captures the important content within the story and leaves the background content where it belongs.

The story object is then linked to all these different objects in the semantic network, using relations including ‘by’, ‘about’ (a person), ‘project’ (about the project), and ‘is’ (a visit story, in this case). In many cases, these correspond to the journalist’s ‘who’, ‘what’, ‘where’, ‘when’, ‘why’, and ‘how’, but the semantic network representation allows this to be easily extended.

A second parallel project (Kalfoglou, Domingue, Motta, Vargas-Vera, & Buckingham Shum, 2001) developed Planet in a different direction, using an ontology to map knowledge in news stories more deeply. When set up, this is highly effective, but full use of an ontology requires significant manual effort. Kalfoglou et al.’s MyPlanet used templates for information extraction, but its templates were at the sentence pattern level and did not build on genres. Although the semantic networks might seem directly opposed to the use of ontologies, this is not actually the case: the network was seeded with a simple ontology. The concept linking approach just allows this to be smoothly extended through the semantic network.

Open Book’s approach was more flexible: it used a very simple predictor-substantiator approach, based on (but much more limited than) that of FRUMP (DeJong, 1982).

This approach allows text to be skimmed rather than deeply processed – ideal for focusing on certain parts of the text with genre recognition, and for ensuring that information extracted is relevant to the point, improving precision substantially, possibly with a small cost in recall. This approach, combining textual and semantic-structural evidence has been evaluated in information retrieval, and seems to work well (Mauldin, 1991).

Sentinel

Where Open Book integrated databases with a web-based newsletter, and dynamically generated hyperlinks, Sentinel addresses the problems of email. Sentinel was a spin-off from the Virtual Participant (Masterton, 1998; Masterton & Watt, 2000). Like the Virtual Participant, Sentinel is a proactive, email-based search and archiving tool; it tracks discussions, and when an issue arises that has been touched on earlier, or which has been stored in its knowledge base, it automatically posts a public message on the matter.

The Virtual Participant had a relatively small, manually constructed case base, and followed the ASK system pattern fairly closely. Sentinel adopted a different approach, driven by the need to scale to tens of thousands of messages. A sample screen display is shown in Figure 2.

Sentinel was developed for a client organization that uses Microsoft Exchange ‘Public Folders’ for extensive networking and knowledge sharing between engineers on a worldwide basis. These forums are used through Outlook, and can store both emails and documents, both directly and as attachments and enclosures in emails. Outlook, however, has an exceedingly primitive (and exceedingly slow) search tool, and this made the valuable experience communicated through these folders very hard to use.

Indeed, it is common for someone to ask a question that was answered only a few months previously, but which they had not been willing or able to find for in the forum archives. This is a common phenomenon, Ackerman (1994) also noted a tendency for people ask questions first and look for answers afterwards – an issue built into the design of Answer Garden. The client traditionally overcame this through forum coordinators, who would prepare summary digests, and store these in a separate part of the Public Folder structure. Sentinel’s web interface significantly improves access to the archive, and using links like Open Book, affords browsing as well as searching.

In effect, Sentinel reads email and uses a set of heuristics and genre patterns to focus on the core parts of each message or document in the public folders. These texts are stored in a database, and indexed using genre rules and a semantic network in just the same way as Open Book.

It was a deliberate design decision to store the messages in the database, rather than to store pointers to the messages in the public forums, for several reasons. First of all, a significant proportion of contributors were not Outlook users – their contributions were mediated through forum moderators or through forum email addresses. Providing access to these users required messages to be copied rather than linked. Secondly, and more importantly, the ‘signal to

noise' ratio in the original messages can be quite low. Sentinel includes a set of rule-based filters that remove extraneous content within messages, such as introductory text, quotes, and signatures. The resulting chronologically ordered threads of cleaned messages improve the readability and usability of messages. Incidentally, Outlook usually appends the original message text at the end of a reply, making each message a duplicate partial digest in reverse chronological order, and very hard to read.

Sentinel has been in operation for a little over two years now, although it has recently undergone a significant overhaul, improving the quality of the information retrieval and indexing techniques it uses – the BM25 probabilistic information retrieval algorithm (Robertson, Walker, Jones, Hancock-Beaulieu, & Gatford, 1995) now forms much of the heart of the system, and the probabilistic conceptual structure it provides is laying a solid foundation for future development. It is currently being rolled out further to additional public email forums within the client organization.

Underlying Technology

Both Open Book and Sentinel were implemented in Perl using a web server supporting Active Server Pages (both Apache and Microsoft's Internet Information Services have been used) and an SQL server (both MySQL and SQL Server have been used). Using a database to represent the semantic network raised intriguing challenges – it is not as easy to trace paths through tables as it is through graph structures in list processing languages. In the end, it was found that all the necessary access involved fairly short paths, of two or three links, and these could all be implemented fairly efficiently using SQL joins of the semantic network table onto itself.

Neither Open Book nor Sentinel use standardized representations for meta-data, such as XML, RDF, or ontologies. However, the knowledge stored in the semantic net-

work can be rendered in these forms, and this is not an inherent limitation of the approach. Indeed, the semantic network approach is also extensible to newer technologies for information integration, such as extended XLinks.

Discussion and Future Work

Although Open Book and Sentinel set out as wholly separate projects, with very different designs, there has been a surprising convergence between them. In retrospect, given the similarity of the requirements, this should not have been so great a surprise as it was. The systems are now close to converging, and it is expected that they will shortly become different aspects of a single system.

Having said that, each system's mode of use provided different insights; Sentinel addressed larger amounts of information than Open Book, and with less structure. These have given some very different reflections on the primary design aims.

Engaging interaction

Open Book in particular seemed to be highly engaging. Although the system has not been formally evaluated, it certainly seemed to afford and encourage browsing. Features that worked especially well were the heuristics which identified who worked with whom, and which projects were related. Neither of these were explicit in the databases, which listed people, and who worked on which projects, but they could be inferred from the projects people had in common, and the people projects had in common.

Sentinel did not relate projects in the same way, but a similar finding was that information about people became more significant as the project developed. In both systems, presenting and integrating information in structures which associated it with named people promoted engagement.

One area where future work is planned is in the form of interaction. Although the web interface has proved highly successful, and email ideal for notifying people and keeping them informed, we are interested in exploring interaction through instant messaging and presence. In particular, Chatterbox (Thomas & Watt, 2002) also looks at an automatically managed ASK system, but will extend the concept linking approach into a chat interface. This will stretch the technology, as chat messages depend on conversational context more than email: issues such as resolving anaphoric references become more important.

Concept linking

The concept linking approach used in both Open Book and Sentinel has proved a practical step towards Cleary and Bareiss's point linking. It seems to deliver much of the benefits of point linking, and to be a significant step beyond simple object linking, yet it is straightforward to implement using fairly conventional technologies. Perl has proved itself well up to the challenges of high-performance symbolic processing, and to embed complex artificial intelligence techniques within web applications.

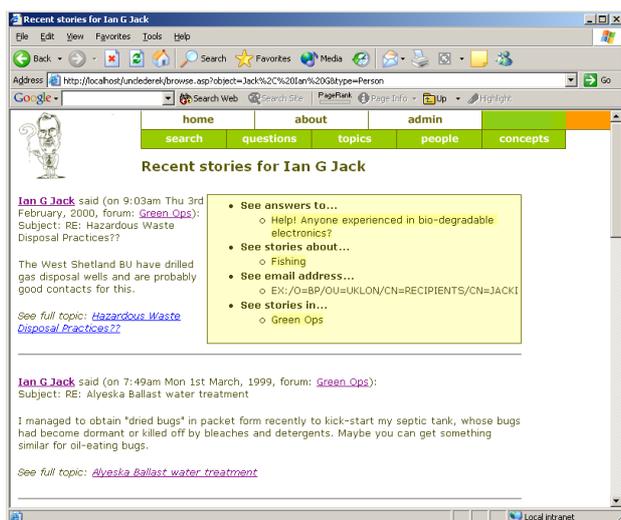


Figure 2. Screen snapshot from Sentinel, showing the 'cleaned' email and some of the automatic links

Despite this, both Open Book and Sentinel are limited in their ability to skim text for concept links, and this ability is hard-coded to a certain extent in both systems. It is planned to develop a more open and more flexible predictor-substantiator system, and to use a more easily specified set of genre templates and higher level dynamic memory structures (influenced by Schank, 1999) to implement a more precise concept linking system with a higher recall.

Evaluation

Neither system has been formally evaluated, and this is a priority for future work. Having said that, evaluation is a complex issue for systems like these, which involve many users, and which involve a disparity in benefit to some users (Grudin, 1994). Sentinel in particular is designed to support a minority of users through its pro-active email interaction, and evaluating learning in a self-selecting subset of (often silent) users is a significant challenge. Evaluating the information integration is a rather different issue. Initial results showed that Open Book was correctly categorizing over 90% of messages, while its linking to free text referenced objects had a rather lower accuracy (somewhat over 70%, although very dependent on the kind of object referred to).

Although there is much research and development work yet to be done, the basic principles of concept linking and a story-based interface have proven themselves a viable and effective approach to information integration in practical knowledge management. Perhaps most interestingly, though, people have turned out to be more pivotal than we expected, and making knowledge about people usable and accessible in an engaging and effective manner is the most active topic for future work.

Acknowledgements

I am grateful to Paul Mulholland and Trevor Collins for collaboration on genre; to Simon Masterton for collaboration on the Virtual Participant; and to Marc Eisenstadt and David Stevens for supporting and guiding work on Open Book and Sentinel respectively.

References

Ackerman, M. S. 1994. Augmenting the Organizational Memory: A Field Study of Answer Garden. In Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'94), Chapel Hill, North Carolina.

Cleary, C., and Bareiss, R. 1996. Practical methods for automatically generating typed links. In Proceedings of the Seventh ACM Conference on Hypertext (Hypertext '96), Bethesda, MD.

Collins, T. D.; Mulholland, P.; and Watt, S. N. K. 2001. Using genre to support active participation in learning communities. In Proceedings of the European Conference

on Computer Supported Collaborative Learning (EuroCSCL'2001), Maastricht, The Netherlands.

DeJong, G. 1982. An Overview of the FRUMP System. In W. G. Lehnert & M. H. Ringle (Eds.), *Strategies for Natural Language Processing* (pp. 149-176): Lawrence Erlbaum Associates.

Domingue, J., and Scott, P. J. 1998. KMi Planet: A Web Based News Server. In Proceedings of the Asia Pacific Computer Human Interaction Conference (APCHI'98), Shonan Village Center, Hayama-machi, Kanagawa, Japan.

Ferguson, W.; Bareiss, R.; Birnbaum, L.; and Osgood, R. 1992. ASK Systems: An Approach to the Realisation of Story-Based Teachers. *Journal of the Learning Sciences*, 2(1): 95-134.

Grudin, J. 1994. Groupware and Social Dynamics: Eight Challenges for Developers. *Communications of the ACM*, 37(1): 92-105.

Kalfoglou, Y.; Domingue, J. B.; Motta, E.; Vargas-Vera, M.; and Buckingham Shum, S. 2001. MyPlanet: an ontology-driven Web-based personalised news service. In Proceedings of the IJCAI'01 Workshop on Ontologies and Information Sharing, Seattle, WA.

Masterton, S. J. 1998. Computer support for learners using intelligent educational agents: the way forward. In Proceedings of the Sixth International Conference on Computers in Education (ICCE'98), Beijing, China.

Masterton, S. J., and Watt, S. N. K. 2000. Oracles, bards, and village gossips, or, social roles and meta knowledge management. *Journal of Information Systems Frontiers*, 2(3/4).

Mauldin, M. L. 1991. Retrieval performance in FERRET: a conceptual information retrieval system. In Proceedings of the 14th International Conference on Research and Development in Information Retrieval, Chicago.

Quillian, M. R. 1968. Semantic Memory. In M. Minsky (Ed.), *Semantic Information Processing* (pp. 216-260). Cambridge, MA: MIT Press.

Robertson, S.; Walker, S.; Jones, S.; Hancock-Beaulieu, M.; and Gatford, M. 1995. Okapi at TREC-3. In Proceedings of the Text REtrieval Conference (TREC-3).

Schank, R. C. 1977. Rules and topics in conversation. *Cognitive Science*, 1(4): 421-442.

Schank, R. C. 1999. *Dynamic memory revisited*: Cambridge University Press.

Thomas, M., and Watt, S. N. K. 2002. Intelligent instant messaging agents to support collaborative learning. In Proceedings of the 16th British HCI Group Annual Conference (HCI2002), London.

Yates, J., and Orlikowski, W. J. 1992. Genres of Organizational Communication: A Structural Approach to Studying Communication and Media. *Academy of Management Review*, 17(2): 299-326.