An Agent-Based Approach to Mailing List Knowledge Management

Emanuela Moreale[1] and Stuart Watt[2]

[1] Knowledge Media Institute (KMi), The Open University,
Walton Hall, Milton Keynes, England, MK7 6AA
e.moreale@open.ac.uk

[2] School of Computing, The Robert Gordon University,
St. Andrew Street, Aberdeen, Scotland, AB25 1HG
s.n.k.watt@rgu.ac.uk

Abstract

The widespread use of computers and of the internet have brought about human information overload, particularly in the areas of internet searches and email management. This has made Knowledge Management a necessity, particularly in a business context. Agent technology – with its metaphor of agent as assistant – has shown promise in the area of information overload and is therefore a good candidate for Knowledge Management solutions. This paper illustrates a mailing list Knowledge Management tool that is centred around the concept of a mailing list assistant. We envisage this system as the first step towards a comprehensive agent-based Knowledge Management solution.

Introduction

As recently as twenty years ago, it was thought that computers would bring in the age of leisure, with considerable shorter working weeks and overall better quality of life (Murch 1999). On the other hand, nowadays computers are more often associated with frustration and tasks reportedly "taking longer than they should". What is certain is that the increasingly widespread use of computers has meant an exponential increase in the amount of stored documents, making the task of locating and retrieving useful information rather complex and time-consuming. On the other hand, today's economy is knowledge-based, so the main asset of companies, and one on which their competitive advantage rests, is their stock of knowledge (Allee 1997; Barchan 1998; Uit Beijerse 2000). Document management is thus a must in today's organisations. Most work in this area has focused on web pages. These efforts range from information retrieval (IR) to information extraction (IE) and wrapper generation (Eikvil 1999).

One of the most important types of document is email. According to a survey commissioned by BT Cellnet (Sturgeon 2001), UK employees spend up to eight hours per week on email. Most of us feel that there is just too much email to deal with and that better support for this essential working tool is needed. Yet, email is complex: it often contains 'noise' (e.g. parts of earlier emails, signatures) and it displays several different formatting conventions (such as paragraphs and signature layouts).

Within mailing lists, the need for information management for email is even more felt: although the 'noise-to-information' ratio varies across lists, the large number of postings often results in subscribers being unable to keep up and/or follow the flow of argument. Postings prior to one's subscription can often be found through archives: yet, these are rarely used, perhaps because of the low perceived success rate of this operation, the time needed and unhelpful archiving conventions (e.g. by subject line). Thus, often queries are asked again and again on a list. It is also likely that humans prefer 'dialectic' discovery involving interaction with active entities to mechanical and repetitive sifting through "passive material". What is certain is that humans prefer to ask someone a question to doing the searching themselves (Ackerman 1994).

This paper explores the application of the metaphor of the agent assistant to mailing lists. Our Sentinel system works with several lists, giving users archiving and retrieval assistance through an intuitive and dialectic interface: users can email their query directly to the agent and receive a prompt reply day or night. Alternatively, users can post their query publicly to a forum (monitored by the agent) or run a web-like search over the monitored lists. Through the application of IE, IR and a novel information integration (II) technique to the mailing lists, Sentinel automatically links email into a tangled network of stories, and arranges them in a meaningful way (digests, queries asked to date), also providing details of contributors and their postings. Because it allows users to notice relationships between/among pieces of information and people, Sentinel is a useful tool to employ as part of an organisation's knowledge management strategy.

This paper will first introduce the concept of agents as assistants and briefly point to research on email management, then describe our approach to the problem of mailing list management: the Sentinel system. An evaluation of the system and future work conclude the paper.

Agents as User Assistants

Agent systems have been proposed as solutions to the problem of information overload, particularly regarding email (Maes 1994) and internet searches (Caglayan and Harrison 1997). Most of the current implementations aiming to ease the burden of dealing with email are text classifiers (Segal and Kephart 1999; Takkinen and Shahmehri 1998) or keyword extractors (Abu-Hakima et al. 2001), often working as email clients plug-ins (Abu-Hakima et al. 2001; Mock 2001). Unlike our system, these solutions target general email in the users' inbox and not specifically mailing lists.

The basic idea behind this paradigm is that software agents are to perform tasks similar to those that a human assistant would carry out. Gruen *et al*. (1999) conducted field studies and an analysis of the types of assistance provided by human assistants. They found that these included: pre-processing, filtering/prioritizing, adding relevant information, performing a number of steps in response to a single request and peripheral awareness/pointing out information. Most of these functions, as applicable to mailing lists, have been included in our Sentinel system.

Murch (1999) dedicates a chapter of his book to email agents. He suggests that they may successfully be employed to perform the following eight activities:

a) Controlling any unwanted email or "spam"
b) Alerting users by voice if a certain message arrives
c) Automatic mail forwarding
d) Consolidating mail from numerous sources
e) Searching the internet for new sources of news, stocks and deals and then delivering them by email
f) Distinguishing between private/personal and corporate/business email
g) Automatically answering email and responding according to conditions
h) Carrying out regular administrative tasks such as archiving and indexing for future searching

While b) is not directly applicable to mailing lists and c) and g) are easily achieved through user-end programming, Sentinel can be said to address these issues, although in many respects it is more than a traditional assistant agent.

Email Analysis and Management

While considerable effort has taken place in the area of document management (from company document warehouses and intranets to efficient IR on the web), document analysis and management techniques obviously depend on document characteristics. Web pages and emails might be considered to be fairly similar types of documents, but mailing lists are characterised by smaller volumes of documents (than the Web), more complex item structure and presence of noise.

Although email is sometimes said to be particularly suitable for knowledge management because "it has a fair amount of metadata attached to it" (Weinberger 1999) (e.g. headers and threading information), the latter can give information that is misleading (for instance when people hit 'reply' to send an email on a new subject).

Emails are largely unstructured documents: while headers are structured, the message body – the text written by the sender – is unstructured[1]. This suggests that headers and message body should be treated differently by the text mining operation. Because of these characteristics of email, it is best not to apply IR to the whole document, but instead minimise noise first. This means totally removing irrelevant emails (e.g. 'out-of-office replies') and then carrying out straightforward IE on header fields and a more sophisticated IE on the body text (e.g. remove salutations and signatures). The 'clean' email text can then be stored in a database, optionally undergoing some kind of information integration. In any case, once in a database, the text is easily searchable and IR techniques can then be successfully used. It is argued that a combination of II, IE, and IR represents the best text mining solution for email.

Email is a dynamic type of document: since changes in employees' interests are reflected in their emails, email gives an up-to-date snapshot of a company's activities and current distribution of expertise within it. Email is also "where coworkers trade stories, ask questions, propose new methods, debate techniques" (Weinberger 1999) and where knowledge is created through interaction (Nonaka and Konno 1998). Email thus constitutes an ideal target for knowledge discovery once the unstructured nature problems are overcome. The next section illustrates such a project, in which we applied the techniques discussed above.

The Sentinel System

We were asked to develop an analysis and management system for email. Our data consisted in files containing several mailing lists arranged as Microsoft Outlook public folders, giving a total of several thousands of complex email messages. Our original task was to develop a tool that interacts with mailing lists, extracting information and answering user queries by email. A later requirement was that of arranging information in an easily searchable and semantically meaningful way. Sentinel builds on previous work (Masterton and Watt 2000): the key recommendations it implements are listed in Table 1.

---

[1] E.g. an email often contains parts of previous emails; paragraphs may be separated by blank lines or not, signatures can have different layouts or be absent.

Issue 1: Public vs. Private Interaction

Recommendation: both should be allowed. If all interaction is forced to be public, the total amount of interaction will be reduced. (Masterton and Watt 2000)

Issue 2: Anthropomorphism vs Mechanomorphism

Recommendation: these systems are more acceptable to users when mechamorphised, i.e. presented as an "Active Archive" rather than as an anthropomorphic character ("Uncle Derek") (Masterton and Watt 2000). This was overturned (Issue 4).

Issue 3: Closeness vs. Openness / Visibility

Recommendation: it is best to open up the system to users as a series of threads, thus contextualising content to the current discussion. (Masterton and Watt 2000)

Issue 4: Fitting into the Company Culture / Groupware sold off-the-shelf is doomed

Groupware needs to be customised (Grudin 1994) and must fit into the company culture (Beyer and Holzblatt 1998). Our target company requested an anthropomorphic character (Figure 1), thus recommendations to avoid anthropomorphism (Issue 2) were overturned.

Table 1. Recommendations implemented in Sentinel

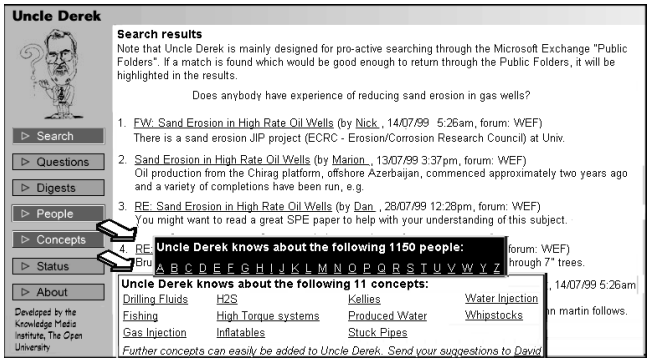

Figure 1. Screenshot illustrating some of the functionality of the Sentinel System, in particular the result of the query: "Does anybody have experience of reducing sand erosion in gas wells?" – some text has been omitted. Uncle Derek is a specific incarnation of Sentinel. Notice the use of anthropomorphism, as requested by the target company.

The Sentinel system has the following characteristics:

- It extracts and stores important information from emails through text mining;
- It links discussions occurring in different lists; the process of feeding knowledge from one forum to another amounts to knowledge discovery for the latter forum;
- It is an information agent system with access to one or more mailing lists. It is able to collate and manipulate information obtained from them in order to answer queries about it such as "Does anybody know about X?" (Wooldridge 2002, p.253). In fact, users can email such queries to this "virtual participant" in the lists and the agent will send a useful reply back to them.
- Alternatively, human users can access a browsable version of the agent's "digested knowledge": the structure of information contained in the list discussions is displayed as a simple semantic network;
- By identifying all contributors to the monitored lists as well as their contributions, it allows people to easily identify each contributor's area of expertise;
- It contains an automated "FAQ-Maker" which extracts discussion digests and identifies previously-asked questions with replies. This low-cost alternative to manually-crafted 'Frequently-Asked Questions' (FAQs) is a partial application of case-based reasoning (CBR) (Watson 1994): problems are identified by initial questions (usually starting a new thread), while replies (often containing 'Best Practices') are the source of case solutions and outcomes. Sentinel can store multiple solution options for each case;
- It uses concepts, a basic ontology for the application domain. These help with knowledge organisation (Mayfield 2002) and provide useful "entry points" for browsing the information network, particularly by people new to the domain and/or list;
- It is presented as an anthropomorphised agent system: although previous work in this area had suggested that CSCW systems are best presented as mechanomorphised systems, this recommendation was overturned to meet the target company's requirements.
- It is a customised system tailored to the target company and fitting into its culture.

The next section will illustrate the steps we followed in developing the system.

Document Analysis and Text Classification Pre-Filtering. The first step consisted in examining the structure of the public folders and several hundred messages. It was evident that some emails should be ignored (e.g. automated 'out-of-office' replies). Text classification was successfully employed here using pairs or triplets of adjacent terms in a manner similar to (Kushmerick *et al.* 2001).

Text Mining. Information Integration and Extraction. First, IE extraction rules were then applied to the 'non-irrelevant' emails to obtain a set of purged email messages. Examples of cleaning rules include: splitting email into simple email chunks and trimming text of main email chunk from both top (to remove salutation) and bottom (to remove signature). The clean text, together with important information about the email (such as threading

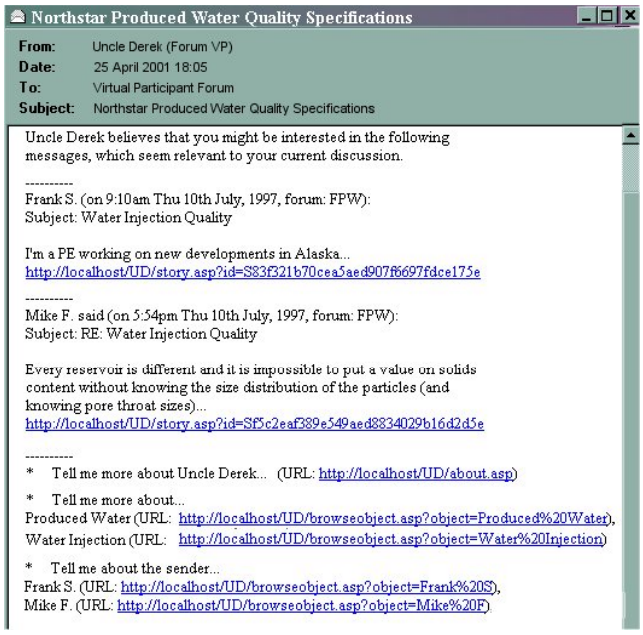

Figure 2. The agent's prompt email reply – some text has been omitted

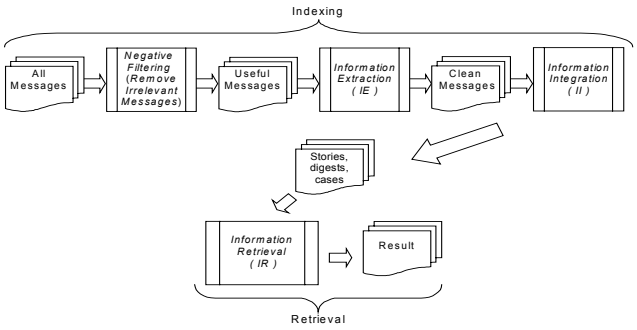

Figure 3. Text and Information Mining Flow in Sentinel (operations/techniques; items)

information), was then stored in a database. Thanks to the hand-crafted rules, the process is remarkably accurate and system recall very high.

An integration step followed: clean email chunks were threaded into 'stories' or coherent sequences of chunks. This story-weaving provides some measure of contextualisation and mimics humans' way of organising information (Schank 2000). We then identified queries asked to date and mapped queries to stories as a means to provide a simple procedure for automated generation of CBR cases.

Technically, Sentinel is implemented in Perl and uses Apache and mod_perl to provide a web interface, and MySQL to store data.

Information integration is central to Sentinel. Stories are linked by author, by subject, by questions, and by domain concepts. Given a story, a user can use the web interface to

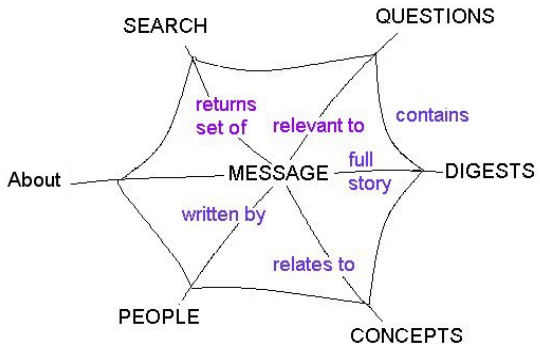

Figure 4. "The Tangled Web of Information": simple semantic network providing a useful way of visualizing the information contained in the monitored mailing lists. The edges represent entry points into the system.

find stories by the same author, addressing a similar question, or touching on similar concepts. Stories are woven together using a semantic network index linking stories to objects through different kinds of relationship ('written by', 'about', and 'asks' are some of the relationships implemented by Sentinel). This low-cost automated hyperlinking technique, simplified from that of Cleary and Bareiss (1996), turns Sentinel's story-base into a tangled web of related stories, which can be browsed through the web.

Sentinel's linking approach depends, through Cleary and Bareiss's work, on Schank's "Conversational/ Associative Category" model (Schank 1977). This is a psychological model of conversational dialogue, and in particular, of topic change in conversations. We are applying this model to explore the extent to which an agent can support the information needs of a community of users in a dialogic style, and as such, Sentinel is a psychological agent as well as a computational one. Although using this is a promising line of work, the current system remains limited in its application of this theory, because the IR techniques it employs are still based on a "query-response" model, and don't yet support the change in topics recommended by this theory. This is a major component of our current work on refining the Sentinel concept.

Information Retrieval and Knowledge Discovery. The last step was to devise an appropriate method to do text mining out of the database as well as to design a suitable query interface. Unlike typical CBR systems, Sentinel does not try to provide definitive answers but, rather, selects the most relevant cases and likely solutions, leaving the final decision to the user. Any story retrieved by Sentinel forms a starting point for further browsing through the web of stories.

Sentinel uses an important additional technique to enhance information retrieval. Communities converge on communicative genres, and we observed several distinct classes of message in the communal lists. Sentinel is

principally intended to respond to one of these – the open question inviting response from the community. Sentinel indexes and stores other messages, including answers, follow-up questions, and forwarding recommendations. Patterns which can be used to recognise these genres (Collins *et al*., 2001) are programmed into Sentinel, and are used as an additional source of evidence when choosing appropriate actions. This hints at the possibility of structuring the system more explicitly and directly as a team of experts with different skills.

Sentinel offers an interactive way to search the archives: users can email a question to the agent privately or post their query to the whole list. The agent monitors the list and, after an appropriate interval of time (to allow human participants to contribute) since a query is posted, can post a reply. In doing so, it will prefer less recent material and postings from other lists.

Alternatively, users can browse the system through a web interface. As well as providing a standard search box, this lists all contributors to the mailing lists, with links to their postings, and allows users to discover knowledge about employees' skills that would otherwise be difficult to gather from the postings.

Evaluation

The system described in this paper has been in use for approximately a year in the target company. Informal feedback suggests that the software has proved useful and has been successfully received. While a formal qualitative evaluation of the system is planned within the next few months, it is significant that an operationalised version of the system has been requested and is currently under development.

It is worth noting that evaluation of large-scale groupware systems like Sentinel is problematic (it is one of Grudin's, 1994, challenges for developers of systems such as this), and most evaluations inevitably resort to surveys. We feel that a more 'utilization-focused' approach (Patton, 1997) is more appropriate in this context, directing evaluation at improving use and uptake of the system, and even assisting key stakeholders in conducting their own evaluations. Our criteria for success, therefore, can be summed up "do people use it?", and although some do, there is definite room for improvement.

In keeping with this approach, we have conducted informal evaluations of the relevance of postings from Sentinel with domain experts, with mixed results. Some matches are highly relevance, others much less so. One important result from our evaluation is that assessing relevance for posting messages is a novel challenge for IR. IR techniques traditionally rank retrieved matches, but less well developed in the area of providing a confidence measure associated with these relevance measures. A separate evaluation of these IR aspects of Sentinel is planned for the near future, although the methodological issues it raises are different to most IR evaluation work (e.g., that in the TREC series of conferences).

The planned formal evaluation will explore the wider presentation and use of the Sentinel system. Some aspects of the implemented system (e.g., management of information about people) evolved as the system was introduced to the users, and were not part of the original concept.

Future Work

Sentinel has been applied with success to internal mailing lists that are part of a corporate intranet. Consideration is being given to expanding its knowledge capture net: being built in a modular fashion, Sentinel can be easily extended to integrate with other document management initiatives to perform more extensive knowledge discovery.

Work is also taking place on a separate Sentinel implementation in which both the filtering and relevance matching stages are carried out by an agent team (Tambe 1997): this should provide easier implementation of certain features such as customisation and personalisation. More work on the IR side is being conducted to support this, developing techniques based on the probabilistic model (Sparck Jones et al. 1998) to provide the kinds of confidence measures that will allow multiple agents to reason about judgements in an effective manner.

Further agents supporting querying through different channels (e.g. WAP, Instant Messaging) are also planned. Other likely upgrades include extending the concept network to integrate with existing ontologies. Future versions might also make more use of CBR and learning.

Conclusion

This paper has illustrated how a combination of shallow text processing techniques (II, IE, and IR) can be successfully used to build an agent software system that alleviates the problem of information overload and management in mailing lists and helps with a company's overall knowledge management strategy.

The case is made for Sentinel, a software tool that uses II, IE and IR techniques, agent technology and CBR to manage information in email within several lists. Sentinel is not just an email-mining agent tool offering a convenient way to archive emails and search over them, but also allows knowledge discovery within mailing lists and organisations.

Acknowledgements

We are grateful to BP for data and support, and to Trevor Collins for valuable input.

References

Abu-Hakima, S., McFarland, C. and Meech J.F. 2001. *Proceedings of the 5th International Conference on Autonomous Agents, AGENTS'01*. Montreal, Quebec, Canada.

Ackerman, M. 1994. Augmenting the Organizational Memory: A Field Study of Answer Garden, *Conference on Computer-Supported Cooperative Work*, pp. 243-252.

Allee, V. 1997. *The Knowledge Evolution: Expanding Organizational Intelligence*, Butterworth-Heinemann.

Barchan, M. 1998. How Celemi Ensure Strategic Gains by Measuring Intangible Assets, *Knowledge Management Review*, September–October 1998.

Beyer, H. and Holzblatt, K. 1998. *Contextual Design*, Morgan Kaufmann.

Caglayan, A.K. and Harrison, C.G. 1997. *Agent Sourcebook: A complete Guide to Desktop, Internet, and Intranet Agents*.

Cleary, C. and Bareiss, R. 1996. Practical Methods for Automatically Generating Typed Links, *Seventh ACM Conference on Hypertext (Hypertext '96)*.

Collins, T. D., Mulholland, P., and Watt, S. N. K. 2001. Using Genre to Support Active Participation in Learning Communities. In the Proceedings of Euro-CSCL 2001, Maastricht, NL.

Eikvil, L. 1999. *Information Extraction from World Wide Web – A Survey*, July 1999.

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

Gruen, D, Sidner, C. and Boettner C. 1999. A Collaborative Assistant for Email, *Proceedings of Human Factors in Computing Systems*, Extended Abstract.

Kushmerick, N., Johnston, E. and McGuinness, S. 2001. Information Extraction by Text Classification, *IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*.

Maes, P. 1994. Agents that Reduce Work and Information Overload. *Communications of the ACM*, 37(7), July 1994.

Masterton, S. and Watt, S.N.K. 2000. Oracles, Bards, and Village Gossips, or Social Roles and Meta Knowledge Management, *Information Systems Frontiers* 2(3/4):299-315.

Mayfield, J. 2002. Ontologies and Text Retrieval. *The Knowledge Engineering Review*, 17(1):71-75, Cambridge University Press.

Mock, K. 2001. An Experimental Framework for Email Categorization and Management. *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'01*, New Orleans, Louisiana, USA.

Murch, R. 1999. *Intelligent Software Agents*. Prentice-Hall.

Nonaka, I. and Konno, N. 1998. The Concept of 'Ba': Building a Foundation for Knowledge Creation, *California Management Review*, 40(3).

Patton, M. Q. 1997. *Utilization-Focused Evaluation: The New Century Text*. Sage Publications.

Schank, R. C. 1977. Rules and Topics in Conversation. Cognitive Science, 1:421-441.

Schank, R.C. 2000. *Tell Me a Story: Narrative and Intelligence, Rethinking Theory*. Third edition. Evanston, Illinois: Northwestern University Press.

Segal, R.B. and Kephart J.O. 1999. MailCat: An Intelligent Assistant for Organizing E-Mail. *Proceedings of the 3rd International Conference on Autonomous Agents, Autonomous Agents '99*, Seattle, WA, USA.

Sparck Jones, K., Walker, S., and Robertson, S. 1998. A Probabilistic Model of Information Retrieval: Development and Status. *Information Processing and Management*, 36(3):809-840.

Sturgeon, W. 2001. *Eight hours per week lost to email*, Dec 2001, www.silicon.com.

Takkinen, J. and Shahmehri, N. 1998. CAFÉ: A Conceptual Model for Managing Information in Electronic Mail. Copyright 1998 IEEE. Published in the *Proceedings of the Hawaii International Conference on System Sciences, HICSS98*, January 6-9, 1998, Kona, Hawaii.

Tambe, M. 1997. Agent Architectures for Flexible, Practical Teamwork. *National Conference on Artificial Intelligence (AAAI-97)*.

Uit Beijerse, R.P. 2000. Questions in Knowledge Management: defining and conceptualising a phenomenon, *Journal of Knowledge Management*, 3(2):94–109.

Watson, I. and Marir F. 1994. Case-Based Reasoning: A Review, *Knowledge Engineering Review*, 9(4):355–381 (1994).

Weinberger, D. 1999. Tacit Knowledge, *KMWorld*, 22nd Nov 1999.

Wooldridge, M. 2002. *An Introduction to Multi-Agent Systems*. John Wiley & Sons.