Intelligent Access to Text: Integrating Information Extraction Technology into Text Browsers

Robert Gaizauskas[a] Patrick Herring[a] Michael Oakes[a]
Michelline Beaulieu[b] Peter Willett[b] Helene Fowkes[b] Anna Jonsson[b]

[a]Department of Computer Science / [b]Department of Information Studies
University of Sheffield
Regent Court, Portobello Road
Sheffield S1 4DP UK
{initial.surname}@sheffield.ac.uk

ABSTRACT

In this paper we show how two standard outputs from information extraction (IE) systems – named entity annotations and scenario templates – can be used to enhance access to text collections via a standard text browser. We describe how this information is used in a prototype system designed to support information workers' access to a pharmaceutical news archive as part of their "industry watch" function. We also report results of a preliminary, qualitative user evaluation of the system, which while broadly positive indicates further work needs to be done on the interface to make users aware of the increased potential of IE-enhanced text browsers.

1. INTRODUCTION

Information extraction (IE) technology, as promoted and defined by the DARPA Message Understanding Conferences [4, 5] and the current ACE component of TIDES [1], has resulted in impressive new abilities to extract structured information from texts, and complements more traditional *information retrieval* (IR) technology which retrieves documents or passages of relevance from text collections and leaves information seekers to browse the retrieved sub-collection (e.g. [2]). However, while IR technology has been readily incorporated into end-user applications (e.g. web search engines), IE technology has not yet been as successfully deployed in end-user systems as its proponents had hoped. There are several reasons for this, including:

1. Porting cost. Moving IE systems to new domains requires considerable expenditure of time and expertise, either to create/modify domain-specific resources and rule bases, or to annotate texts for supervised machine learning approaches.
2. Sensitivity to inaccuracies in extracted data. IE holds out the promise of being able to construct structured databases from text sources automatically, but extraction results are by no means perfect. Thus, the technology is only appropriate for applications where some error is tolerable and readily detectable by end users.
3. Complexity of integration into end-user systems. IE systems produce results (named entity tagged texts, filled templates) which must be incorporated into larger, more sophisticated application systems if end users are to gain benefit from them.

In this paper we present the approach taken in the TRESTLE project (Text Retrieval Extraction and Summarisation Technologies for Large Enterprises) which addresses the second and third of these problems; and also preliminary results from the user testing evaluation of the TRESTLE interface. The goal of the TRESTLE project is to develop an advanced text access facility to support information workers at GlaxoSmithKline (GSK), a large pharmaceutical corporation. Specifically, the project aims to provide enhanced access to *Scrip*[1], the largest circulation pharmaceutical industry newsletter, in order to increase the effectiveness of employees in their "industry watch" function, which involves both broad current awareness and tracking of people, companies and products, particularly the progress of new drugs through the clinical trial and regulatory approval process.

2. IE AND INFORMATION SEEKING IN LARGE ENTERPRISES

While TRESTLE aims to support information workers in the pharmaceutical industry, most of the functionality it embodies is required in any large enterprise. Our analysis of user requirements at GlaxoSmithKline has led us to distinguish various categories of information seeking. At the highest level we must distinguish requirements for current awareness from those for retrospective search. Current awareness requirements can be further split into general updating (what's happened in the industry news today/this week) and entity or event-based tracking (e.g. what's happened concerning a specific drug or what regulatory decisions have been made).

Retrospective search tends to break down into historical tracking of entities or events of interest (e.g. where has a specific person been reported before, what is the clinical trial history of a particular drug) and search for a specific event or a remembered context in which a specific entity played a role.

[1]*Scrip* is the trademark of PJB Publications Ltd. See `http://www.pjbpub.co.uk`.

.

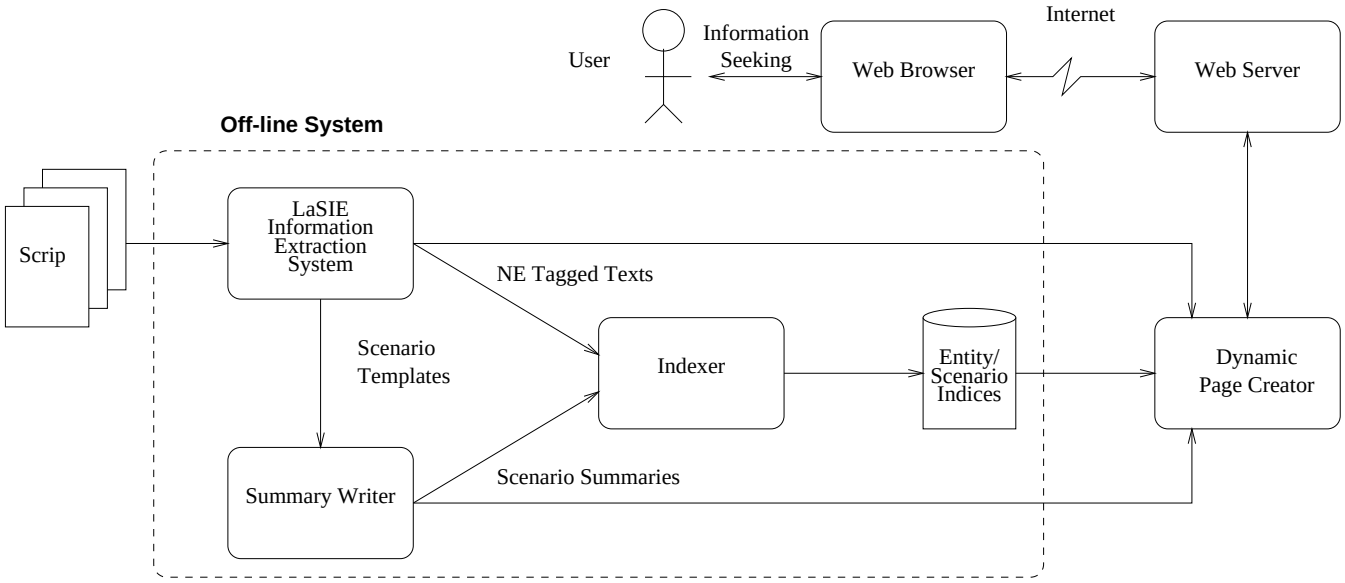

Figure 1: TRESTLE Architecture

Notice that both types of information seeking require the identification of entities and events in the news – precisely the functionality that IE systems are intended to deliver.

3. THE TRESTLE SYSTEM

The overall archictecture of the TRESTLE system is shown in Figure 1. The system comprises an on-line and an off-line component. The off-line component runs automatically whenever a new electronic delivery of *Scrip* takes place. It runs an IE System (the LaSIE system, developed for participation in the MUC evaluations [6]), which yields as output Named Entity (NE) tagged texts and Scenario Templates. To address the domain of interest, the MUC-7 NE categories of person, location and organisation have been retained and the categories of drug names and diseases have been added. The system generates three scenario templates: person tracking (a minor modification of the MUC-6 management succession scenario), clinical trials experimental results (drug, phase of trial, experimental parameters/outcomes) and regulatory announcements (drugs approved, rejected by various agencies).

After the IE system outputs the NE tagged texts and scenario templates, an indexing process is run to update indices which are keyed by entity type (person, drug, disease, etc.) and date, and by scenario type and date.

The on-line component of TRESTLE is a dynamic web page creation process which responds to the users' information seeking behaviour, expressed as clicks on hypertext links in a browser-based interface, by generating web pages from the information held in the indexed IE results and the original *Scrip* texts. A basic Information Retrieval component has also been plugged in to TRESTLE to provide users with seamless access to query *Scrip* texts, i.e., not confined to the pre-defined named entities in the index.

3.1 Interface Overview

The interface allows four ways of accessing *Scrip*: by headline, by named entity category, by scenario summary, and by freetext search. For the three first access routes the date range of *Scrip* articles accessible may be set to the current day, previous day, last week, last four weeks or full archive.

The interface is a browser whose main window is divided into three independently scrollable frames (see Figure 2). An additional frame (the "head frame") is located at the top displaying the date range options, as well as information about where the user currently is in the system. Down the full length of the left side of the window is the "access frame", in which text access options are specified. The remainder of the main window is split horizonally into two frames, the upper of which is used to display the automatically generated index information (the "index frame") and the lower of which is used to present the *Scrip* articles themselves (the "text frame").

Headline access is the traditional way GSK *Scrip* users access text, and is retained as the initial default presentation in TRESTLE. In the index frame a list of *Scrip* headlines is presented in reverse chronological order. Each headline is a clickable link to full text of the article; clicking on one displays the full text in the text frame (like Figure 2, only without the second column in the index frame).

Named entity and scenario access are the novel IE-based techniques TRESTLE supports.

3.2 NEAT: Named Entity Access to Text

From the access frame a user selects a category, for example, drugs. The index frame then displays an alphabetically ordered list of drug names extracted from the *Scrip* texts by the IE engine (Figure 2). To the right of each drug name is the title of the article from which the name was extracted (if a name occurs in multiple texts, there are multiple lines in the index frame). Once again the title is a hyperlink to the text and if followed the full text is displayed in the text frame.

When a text is displayed in the text frame, every occurrence of every name which has been identified as a named entity of any category is displayed as a clickable link; furthermore, each name category is displayed in a different colour. Clicking on a name, say a company name (e.g. Warner-Lambert in Figure 2) occurring in a text which was accessed initially via the drug index, updates the index frame with the subset of entries from the index for that name only – in our example, all entries for the selected company.

In addition to listing the full drug index alphabetically, the user may also enter a specific drug name in the `Index Look-up` box

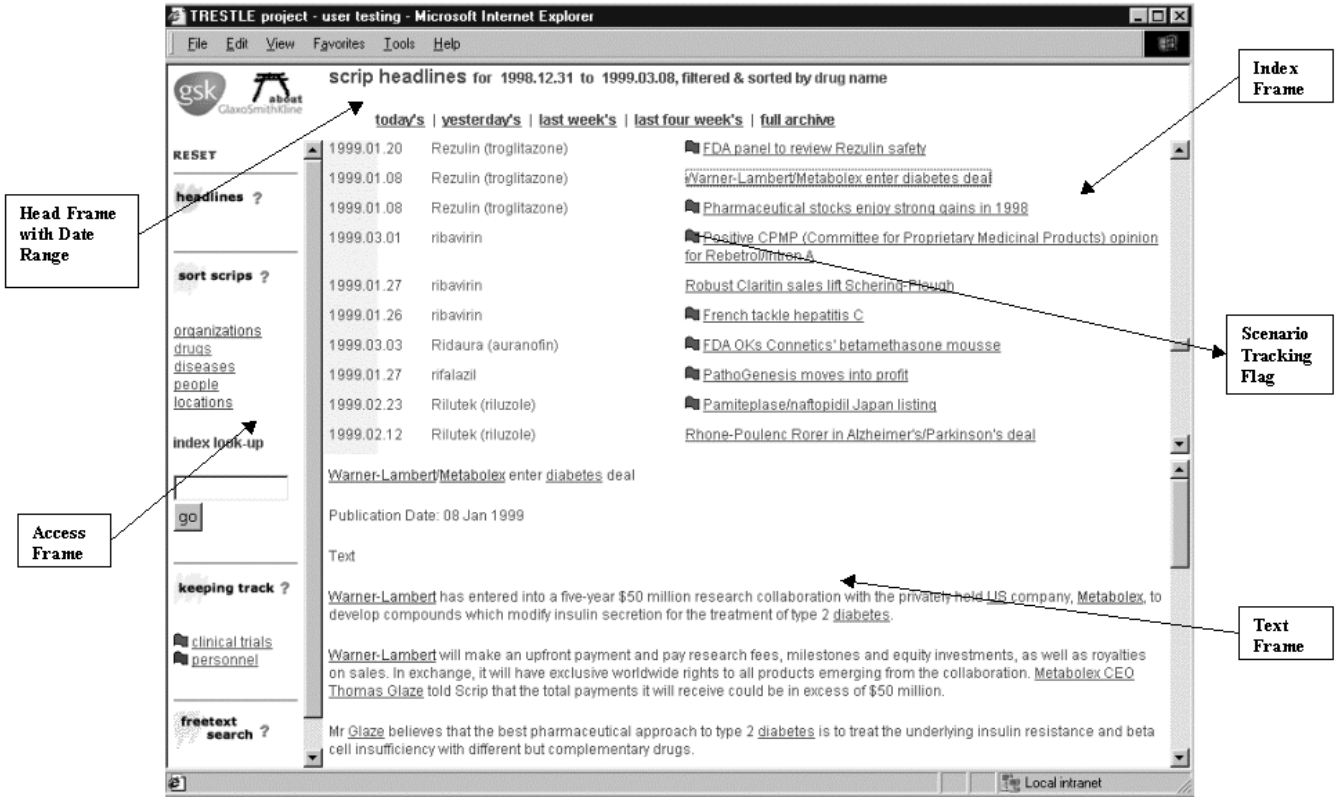

Figure 2: TRESTLE Interface: NEAT

in the access frame, and the index frame will then list the titles of all articles containing that drug name.

NEAT allows rapid text navigation by named entity. A user with a watching brief on, say diabetes, can start by reviewing recent articles mentioning diabetes, but then follow up all recent references to companies or drugs mentioned in these articles, extending the search back in time as necessary, and at any point branching off to pursue related entities.

3.3 SCAT: Scenario Access to Text

While NEAT allows navigation by named entity, the user still derives information by reading the original *Scrip* texts. Scenario access to text (SCAT) utilises summaries generated from templates extracted by the scenario template filling component of the IE system to provide access to the source texts. It is based on the observation that many scenarios of interest can be expressed via single sentence summaries. For example, regulatory announcements in the pharmaceutical industry can be captured in a template and summarised via one or more simple sentence schemas such as "Agency approves/rejects/considers Company's Drug for Disease in Jurisdiction".

To use SCAT a user selects one of the tracking options (`keeping track`) from the access frame of the interface. A list of one line summaries, one per extracted scenario, is then presented in the index frame. Along with each summary is a link to the source text, which allows the user to confirm the correctness of the summary, or to follow up for more detail/context. Clicking on this link causes the source text to appear in the text frame (see Figure 3). The presence of a summary in a *Scrip* article is also presented to the user through coloured tracking flags next to the article headline (see Figure 2). This feature can be viewed as a shortcut to the summary facility; clicking the flag gives the generated summary in the text frame together with the link to the source. Of course sufficient information may have been gleaned from the summary alone, obviating the need to read the full text.

4. PRELIMINARY USER EVALUATION

Although input from users has informed each stage of the design process from the conceptual non-interactive mock-ups to the development of the web-based prototype, this section reports on a preliminary evaluation of user testing of the first fully functional prototype. The aim was to elicit feedback on the presentation and usability of NEAT and SCAT and the overall interface design. The objectives were two-fold. Firstly, and more broadly, to assess to what extent the interface conformed to principles of good usability design such as simplicity, consistency, predictability, and flexibility [7]. Secondly, and more importantly, to focus on the interaction issues presented by NEAT and SCAT:

- procedurally, in terms of users' ability to move between different search options in a logical and seamless fashion; and
- conceptually, in terms of users' awareness and understanding of the respective functions for exploiting current and retrospective Scrip headlines and full text.

4.1 Evaluation Methodology

A group of eight users consisting of postgraduate students and research staff were recruited from the Department of Information Studies at the University of Sheffield. The subjects had different subject backgrounds and all had experience of using web based

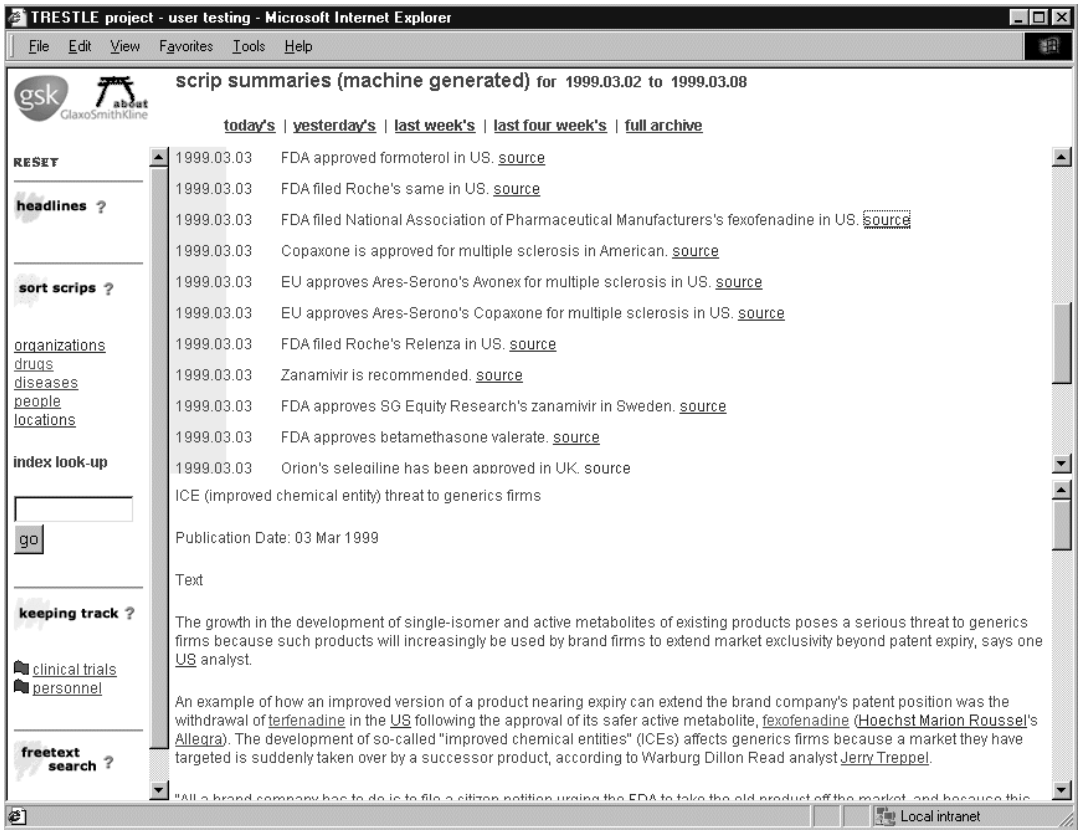

Figure 3: TRESTLE Interface: SCAT

interfaces, searching for information online and some knowledge of alerting/current awareness bulletins.

The focus of the exercise was to observe user-system interactions 'real time' to gain insight into:

- ease of use and learnability of the system;
- preferred strategies for accessing text;
- problems in interpreting the interface.

In addition, user perceptions of the interface were also elicited to provide further explanations on searcher behaviour. A combination of instruments was thus used including a usability questionnaire, verbal protocols and observational notes. Note that this evaluation was a purely quantitative exercise aimed at gaining an understanding of how the users responded to the novel functions offerred by the interface. A further evaluation will take place in an operational setting with real end users from GSK.

After a brief introduction to the purpose of the evaluation and a brief overview of the system, users were asked to explore the system in an undirected manner, asking questions and providing comments as they proceeded. Following this, they were asked to carry out a number of tasks from a set of tasks that simulated typical information needs characteristic of real end-users at GSK and were instructed to identify a 'relevant' article for each task. The tasks were designed to exploit the full functionality of the prototype; an example of the task is given below:

> *You've heard that one of your colleagues, Mr Garcia, has recently accepted an appointment at another pharmaceutical company. You want to find out which company he will be moving to and what post he has taken up.*

The number of tasks completed by each subject varied according to the subject's thoroughness in exploring the system. The order in which the tasks were assigned was random.

4.2 Access Strategies

Access to named entities was made available in three ways:

1. by clicking directly on a list of four categories;
2. through the index look up query box;
3. through the free-text keyword search option.

The optimal strategy differed for the different assigned tasks. Most subjects tended to use the index look-up as a first attempt irrespective of its appropriateness for the task in hand. Preference for the use of the index look-up as opposed to selecting more general entity categories may be explained by the fact that users knew what they were looking for (i.e. an artefact of the assigned task). Moreover the query box for the index look-up option may have been a more familiar feature which encouraged searchers to adopt a searching strategy as opposed to browsing named entities. The preference for using the index look-up option over free text searching may have been influenced by the order of presentation as well as the prominence of the text entry box in the access frame. In addition for assigned tasks where the choice was between any of the three entity access strategies, or using the tracking options, the majority of users opted for the entity access via the index look-up. The novelty of the tracking options appeared to be a contributory factor.

4.3 User Perceptions

4.3.0.1 Colour Coding.

The colour coding of the named entities was highly noticeable, although there was some disagreement on its usefulness. Of those subjects that found the colour coding unhelpful, it was the choice of colours that they objected to rather than the function of the colour *per se*. Although subjects claimed that coloured entity links were distracting when reading full news items, the majority indicated that the linking to previous Scrip items was very useful. The distraction often had a positive effect in leading to useful and related articles. The overall integration of the current awareness and retrospective searching functions through named entities was thus widely appreciated.

4.3.0.2 Index Look-up.

All subjects except one found the index look-up function useful, once they discovered that it was a quick way of accessing predefined named entity categories. The fact that the approach only provided exact string matching was judged to be limiting.

4.3.0.3 Scenario Tracking.

The `keeping track` option was not as easily understood as the named entity options. The label "keeping track" was misinterpreted by some subjects as a search history function or an alerting service based on user profiles. After having used the tracking facility half of the subjects did, however, correctly understand the function. One problem that arose was the differentiation between summaries presented in SCAT and the actual Scrip headlines. Although the header informed searchers that they were viewing Scrip summaries, the display of the summaries in the same frame where the headlines were normally presented as well as the similarity in content led to confusion.

The coloured flags next to the headlines, which were meant to serve as a tracking label to allow users to move seamlessly from headlines to scenario summaries, raised another problem. Not only was the meaning of the flag symbol poorly understood, but also subjects did not realise that they could click on it. Moreover when they clicked on the flag they expected to see a full news item rather than a summary. Hence, the scenario access was both procedurally and conceptually confusing.

5. CONCLUSIONS

To date IE has largely been a "technology push" activity, with language engineers working to develop core technologies. For the technology to become usable, and for its further development to be influenced by end user requirements ("user pull"), prototype end-user application systems must be built which exploit the significant achievement of the technology to date, while acknowledging its limitations. In this paper we have described such a prototype, the TRESTLE system, which exploits named entity and scenario template IE technology to offer users novel ways to access textual information.

Our preliminary evaluation has revealed that although search options initially selected from the access frame were not always optimal for undertaking set tasks, the colour coded textual and iconic cues embedded in the headline index and full text frames on the whole enabled users to exploit the different functions seamlessly. Whilst the TRESTLE interface appeared to support interaction at a procedural level, at the conceptual level however, searchers did not necessarily gain sufficient understanding of the underlying functionality, particularly in respect to the scenario access. For example the inability to distinguish between the original headlines and the system generated summaries for SCAT was problematic and requires further investigation. Other studies have reported similar issues in introducing more complex interactive search functions [3, 8]. More meaningful labelling may in part address some of the difficulties encountered. A more extensive evaluation in a work setting will follow to assess to what extent the integration of new and established conventions can support users with domain knowledge and greater familiarity with alerting systems to adopt new searching and awareness approaches effectively.

6. ACKNOWLEDGEMENTS

The authors would like the acknowledge the financial support of GlaxoSmithKline which has made this work possible, and in addition the helpful comments and insights of many staff at GSK, in particular Peter McMeekin, Charlie Hodgman, David Pearson and Derek Black.

7. REFERENCES

[1] ACE: Automatic Content Extraction. http://www.itl.nist.gov/iaui/894.01/tests/ace/. Site visited 08/01/01.

[2] R. Baeza-Yates and B. Ribiero-Neto. *Modern Information Retrieval*. ACM Press Books, 1999.

[3] M. Beaulieu and S. Jones. Interactive searching and interface issues in the Okapi best match probabilistic retrieval system. *Interacting with Computers*, 10:237–248, 1998.

[4] Defense Advanced Research Projects Agency. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995.

[5] Defense Advanced Research Projects Agency. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998. Available at `http://www.saic.com`.

[6] K. Humphreys, R. Gaizauskas, S. Azzam, C Huyck, B. Mitchell, H. Cunningham, and Y. Wilks. Description of the LaSIE-II system as used for MUC-7. In MUC-7 [5]. Available at `http://www.saic.com`.

[7] J. Nielson. *Designing Web Usability: The Practice of Simplicity*. New Riders, 2000.

[8] A. Sutcliffe. Evaluating the effectiveness of visual user interfaces for information retrieval. *International Journal Human-Computer Studies*, 53:741–763, 1982.