

Novel User Interfaces via Model-Mediated Information Retrieval

Earl J. Wagner, Jiahui Liu and Larry Birnbaum
Intelligent Information Laboratory
Northwestern University
Evanston IL USA
{ewagner, j-liu, birnbaum}@cs.northwestern.edu

ABSTRACT

Using content-specific models to guide information retrieval can provide richer interfaces to end-users in both navigating news articles and learning the context of news events. We present *Brussell*, a system that uses semantic models of news event situations to perform anticipatory information retrieval, organize extraction results and present a novel interface for navigating among the milestone events of a situation.

1. INTRODUCTION

People browse the web not only to search for specific facts, but also in "'building a picture' of an organization, topic or person." [11] However, the nature and specific *kinds* of "big picture" views that might benefit information gatherers, and how software might be constructed to support their elaboration, has not received nearly as much attention as search more narrowly construed.

The need for a "big picture" view is particularly acute when reading news. An article may cover a new event involving organizations and individuals previously unknown to the reader. Or the reader may be familiar with the event participants, but not with the overall situation involving the event—where by *situation* we mean a limited sequence of causally-related events, such as all of the newsworthy actions in a lawsuit. For example, the dismissal of a lawsuit follows the filing of the lawsuit and both are part of a particular lawsuit situation.

In establishing the context of a new event, news articles reference previous events. Often these events are related to the topic of the current article by being part of the same overall situation - perhaps an earlier event in the situation, such as the filing of the suit. Or it may reference other similar or related situations. A similar lawsuit may be taking place in another locale. Related lawsuits include a suit acting as a case precedent, or other suits involving some of the same participants, such as other suits against the defendant.

All of these relationships are part of the situational context that the user draws upon in making sense of the events the article describes. This context gives rise to specific questions, such as:

- What happened in this situation?
- What happened in the other situations referenced in this article?
- What other similar and related situations have these participants been involved in?

Neither conventional news web pages nor current browser software provides content-specific support for answering these questions, however.

Some online news sources offer links to related pages, but these are frequently irrelevant or out of date. An article web page about the filing of a lawsuit isn't typically updated to link to coverage of the lawsuit's dismissal. Some articles link previous-event textual references to earlier articles, though these links must be added manually.

Without an in-page link, to answer her natural questions, the user must find related articles manually. She must identify relevant terms such as entity names and situation keywords. Then she must cut-and-paste them into a news search engine. Finally she must sort through lists of results to find relevant articles. These steps make for an inconvenient process familiar to anyone who reads news on the web. Even news timelines provided by advanced search engines are unable to provide content-specific overviews of a situation in accordance with the user's expectations of how it begins and continues.

Existing automated approaches typically offer support through domain-independent methods, such as by clustering articles based on term frequencies, or summarizing multiple articles about the event. These approaches don't leverage a user's expectations, however, for how the situation has unfolded *causally* and how it will proceed. For example, a lawsuit that begins with a high-profile filing may end with a low-profile settlement. Although a user expects the lawsuit to end in one of several ways, domain-independent systems do not and may miss these more obscure events. A domain-specific approach is necessary to support users' expectations for how events relate in a situation and thus enable new kinds of user interaction.

We present *Brussell*, a system that performs anticipatory information retrieval and model-based information extraction to support the user in exploring the situational context of the news. *Brussell* retrieves news articles and creates and extracts situation models from templates. When a user selects a situation, it presents a storyline with the major milestone events. Clicking on the event label loads an article that either immediately covers the event or is the earliest mention of the event. Evidence that an event took place, for its date and location, or for important attributes of participating entities can also be viewed in the form of collected textual snippets and links to source pages.

2. EXAMPLE

Consider the case of a user reading about the history of the terrorist group Hamas. The article references the kidnapping of a BBC journalist, and although the user was vaguely aware of this incident, he would like to find out more. With standard search technology, he would enter terms into a search engine and peruse the results in order to develop an overall sense of how the kidnapping situation transpired. Through Brussell, he can interact with the textual reference directly, by first clicking on a button in the Brussell toolbar to show its situation reference "matches", then right-clicking on the highlighted text in the page (see Figure 1).

The context menu presents options for viewing the history of the situation and finding out more about its participants (see Figure 2). The user wants to see a summary of what happened, so he selects the first option, which updates the toolbar to show a storyline for the kidnapping with its major events and their dates (see Figure 3).

Next, he wants to know more about how the journalist was released, so he selects the "release" event button that loads the most relevant page describing the event in detail (see Figure 4).

3. ARCHITECTURE

Brussell consists of a Firefox browser plugin and server software, which may both run on the same computer. When the user wants to inspect a situation reference the browser plugin sends the current page title and URL to the server, which responds with the (possibly cached) page situation references. A user can view situation references in news pages, as in the example, or can request the analysis of arbitrary web pages, such as blog posts.

The back-end system requires manually-created situation model types (scripts) and currently supports *kidnappings*, *legal trials* and *corporate acquisitions* each of which has multiple possible outcomes and on the order of 8-12 possible events. The system runs daily to retrieve news articles from several news web sites via RSS feeds and store them in a Lucene index. [7] It then queries the database for new articles with keywords associated with the situation types it supports and reads through the returned articles to instantiate and extend situation models of these types. Situations include information from a few articles, up to several hundred if they are well-publicized.

Brussell uses GATE [4], a standard open-source information extraction system to extract situation information including event references, dates and locations, and entity information such as person names and



Figure 1. Viewing a situation reference within an article.

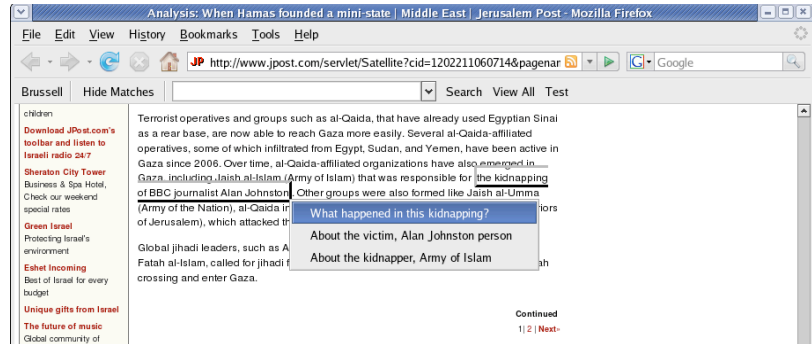


Figure 2. Asking about the situation.

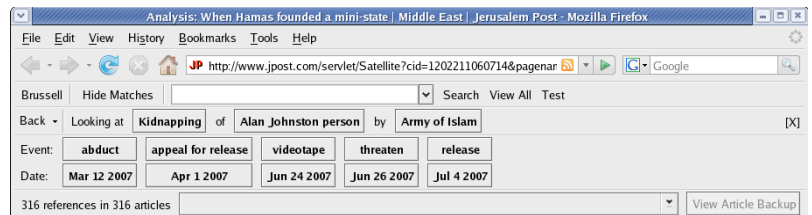


Figure 3. Viewing milestone events for the selected situation

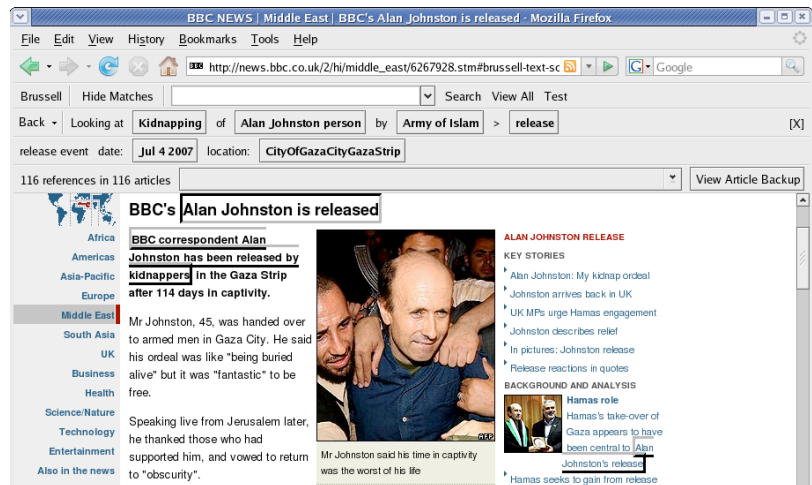


Figure 4. Viewing the article for the selected situation event.

occupations or organization names and nationalities. Extracting this information allows references such as "the British journalist abducted last year" to be resolved to a particular kidnapping. In fact, the same mechanism used for extracting information is used to identify situation references in page text, and in analyzing news articles, the system caches the textual references for all of articles it processes. Saving textual supports for extracted information serves an additional purpose: to justify how conflicting information has been reconciled.

3.1 Resolving Conflicting Article Information and Extraction Results

A well-known problem with building and manipulating explicitly represented models is that of resolving conflicting information. Often a breaking news article features incorrect information that is later amended. Or information in an article may be correct, but presented idiosyncratically and, as a result, extracted incorrectly. Based on the expectation that correct information will be stated more often than incorrect information, Brussell implements a voting algorithm to resolve error due either to incorrect article information or faulty extraction.

Voting is used to resolve conflicts at multiple levels:

- At the top-most level, to select which actual events occur within a situation
- Around event information including dates, locations and monetary amounts
- Concerning biographical information about situation participants such as person names and occupations or organization names and nationalities

A preliminary evaluation of this voting approach shows that the performance of relatively shallow extraction technologies integrated across multiple documents is comparable to more sophisticated extraction from single document, as found in, e.g., the MUC competitions.

4. BACKGROUND

Previous research has produced query-free information retrieval systems for end users such as Letizia [6] and Watson [3]. These systems search the web to find documents relevant to a user: Letizia by following the links of the currently open web page, and Watson by modeling her current task in the browser or an open Microsoft Office document.

Several areas of research have focused on distilling information from multiple news articles. Techniques in text summarization merge and reduce the information in multiple documents presenting the user with a natural language summary. [8] Research in topic-detection and tracking has focused on representing events, typically by term-vectors, and classifying and clustering documents using these event representations. [1] These domain-independent approaches do not model types of events and situations and the associated semantic constraints and thus cannot support users' expectations for the milestones of these situations and how they proceed. Our approach of modeling user expectations for situations is based on the script conceptual formalism for story understanding. [10].

Extracting event information using templates from single news articles was the focus of work in the Message Understanding Conferences [5]

One notable site that uses a model to extract and integrate information from multiple web pages is ZoomInfo.com, which automatically generates an individual's CV based on text references in web pages. [12]

5. FUTURE WORK

Two challenges remain for the system to scale not just on many articles, but many situation types. First, there is the problem of generating situation type models that consist of semantic constraints, document retrieval keywords and extraction patterns. Authoring the patterns is the most time-consuming component by far, though this could be automated through unsupervised learning techniques such as [9] or [13]

As more types of situations are modeled, support for richer knowledge representation will be required. For example, tracking an individual's employment at an organization would require representing an individual's occupation as multiple job records not just strings. Although trivial, it is expected that supporting more situation types will introduce many new representation requirements such as this one, each of which must be accommodated within the voting system.

6. CONCLUSION

Many researchers have put forward the goal of integrating the web with high-level semantic models to provide more goal-oriented interfaces. Some, including those working as part of the Semantic Web effort, expect to provide this user-level functionality by requiring authors to annotate their web pages using standardized domain-specific logical annotations. [2] In other words, this effort is aimed at providing smarter interactions with web content by constructing the web out of explicit logical representations.

We are taking the opposite approach to semantically-informed user interaction with web content. Rather than dragging the web to semantics, kicking and screaming, we are bringing semantics to the web. With Brussell, we have presented a system that enables users to interact directly with entities and situations mentioned in web pages in order to navigate the context of the content they are viewing. Brussell uses standard IR and IE technologies integrated with situation model templates to anticipate user questions, and provide links to - and summaries of - the answers resulting in high-level overviews of situations that match user expectations.

7. REFERENCES

- [1] Allan, J., Carbonell, J., Doddington, G., Yamron, J., and Yang, Y.: 1998, 'Topic Detection and Tracking Pilot Study: Final Report'. In: Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop. San Francisco, CA, pp. 194-218, Morgan Kaufmann Publishers, Inc.
- [2] Berners-Lee, T., Hendler, J. & Lassila, O. "The Semantic Web", Scientific American 284(5):34-43 (May 2001)
- [3] Budzik, J. and Hammond, K. J. 2000. User interactions with everyday applications as context for just-in-time information access. In *Proceedings of the 5th international Conference on intelligent User interfaces* (New Orleans, Louisiana, United States, January 09 - 12, 2000). IUI '00. ACM, New York, NY, 44-51. DOI=<http://doi.acm.org/10.1145/325737.325776>
- [4] Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V. GATE: A Framework and Graphical Development

Environment for Robust NLP Tools and Applications. *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*. Philadelphia, July 2002

- [5] Grishman, R. 1997. Information Extraction: Techniques and Challenges. In *International Summer School on information Extraction: A Multidisciplinary Approach To An Emerging information Technology* M. T. Paziienza, Ed. Lecture Notes In Computer Science, vol. 1299. Springer-Verlag, London, 10-27.
- [6] Lieberman, H., Letizia: 1995. An Agent That Assists Web Browsing, *Proceedings of the 1995 International Joint Conference on Artificial Intelligent*, Montreal, Canada, August 1995.
- [7] <http://lucene.apache.org/java/docs/>
- [8] McKeown, K. and Radev, D. R. 1995. Generating summaries of multiple news articles. In *Proceedings of the 18th Annual international ACM SIGIR Conference on Research and Development in information Retrieval* (Seattle, Washington, United States, July 09 - 13, 1995). E. A. Fox, P. Ingwersen, and R. Fidel, Eds. SIGIR '95. ACM, New York, NY, 74-82. DOI= <http://doi.acm.org/10.1145/215206.215334>
- [9] Riloff, E. (1996) "Automatically Generating Extraction Patterns from Untagged Text", *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)* , 1996, pp. 1044-1049
- [10] Schank, R. C. and Abelson, R. P. 1977. *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.
- [11] Sellen, A. J., Murphy, R., and Shaw, K. L. 2002. How knowledge workers use the web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves* (Minneapolis, Minnesota, USA, April 20 - 25, 2002). CHI '02. ACM, New York, NY, 227-234. DOI= <http://doi.acm.org/10.1145/503376.503418>
- [12] <http://www.zoominfo.com/>
- [13] Yangarber, R. 2003. Counter-training in discovery of semantic patterns. In *Proceedings of the 41st Annual Meeting on Association For Computational Linguistics - Volume 1*