Information extraction tools are improving the way companies find information in a sea of text.

Sarah M. Taylor

Information Extraction Tools: Deciphering Human Language

Are you and your colleagues missing information you need to do your jobs? Are you blindsided by news about important corporate acquisitions, product offerings, or key personnel changes at rival firms? Could your company be more competitive if it had a complete picture of all its competitors and potential customers?

Although anyone would probably answer yes to all these questions, just the thought of trying to consume even more information induces a huge headache. Most readily available tools—basic search engines, possibly a news or information service, and perhaps agents and Web crawlers—are inadequate for many information retrieval tasks and downright dangerous for others. These tools either return too much useless material or miss important material. Even when such tools find useful information, the data is still in a text form that makes it difficult to build displays or diagrams. Employing the data in data mining or standard database operations, such as sorting and counting, can also be difficult.

Today, an emerging technology called *information extraction* (IE) is beginning to change all that, and you might already be using some very basic IE tools without even knowing it. Companies are increasingly applying IE behind the scenes to improve information and knowledge management applications such as text search, text categorization, data mining, and visualization ("From Unstructured Data to Actionable Intelligence," Ramana Rao, *IT Professional*, Nov.-Dec. 2003). IE has also begun playing a key role in fields such as national security, law enforcement, insurance, and biomedical research, which have highly critical information and knowledge needs. In these fields, IE's powerful capabilities are necessary to save lives or substantial investments of time and money.

Figure 1 shows the organizations, people, locations, and time-date expressions found in one newspaper article. In addition, the IE tool identifies some verbs for further processing. Once the tool identifies these entities, they can be useful for many tasks. Having a simple list of entities, for example, can make summarizing a document's content easier. Entities can also help improve the accuracy of search tools like Google. Users can record extracted information in databases and apply the information to aggregated analyses, answering questions such as, How many export sales of aircraft did the news media report in 2003, and what was the total value for all sales?

IE: HELPING COMPUTERS DECIPHER HUMAN LANGUAGE

Human language records much of the information and knowledge that people produce and must manage. But computers cannot help users much with language in its natural form. The characteristics that make language valuable to us as humans—great variety and flexibility—make it difficult for the software to process in a way that accounts for meaning. For example, is *Rose* a flower, color, person's name, or the past-tense

Inside

Published by the IEEE Computer Society
1520-9202/04/$20.00 © 2004 IEEE

form of the verb to rise? Is "his heart swelled within his breast" the symptom of a disease or only poor writing? Is the phrase, "what a turkey" praise or disapproval?

The possible examples are endless and amusing, but using computer tools to help people find, understand, and manage information in text is a very real challenge.

IE is one technology to help solve these problems. It grew from the need to pull specific information from large volumes of text and store this information in structured databases, where users could quickly query, aggregate, and otherwise analyze it. Thus, IE views language up close, considering grammar and vocabulary, and tries to determine the details of "who did what to whom" from a piece of text. In its most in-depth applications, IE is domain focused; it does not try to define all the events or relationships present in a piece of text, but focuses only on items of particular interest to the user organization.

Most systems extract named people, organizations, and locations, as well as date expressions. Systems also commonly extract numbers, such as telephone numbers and currency amounts, which are fairly easy to identify. It is possible to closely tie the extraction of other entities to the particular subject area and customize it for a subset of users, such as those in the pharmaceutical industry. It is not uncommon for entity identification to be broad, but shallow—that is, a tool might be able to identify a few of many categories of entities, but leaves many within each category undiscovered. Accuracy on a few well-understood entities, like named people, does not guarantee similar accuracy on all entities. Lockheed Martin's AeroText software can extract the list of entity types in Table 1.

At a practical level, the input to an IE system is a continuous stream (or batch) of text documents straight from the word processor, newswire, Internet, or a scanning or OCR (optical character recognition) system. The output is a list of tags (such as first

Figure 1. Information extraction technology finds entities, relationships, and events in text.

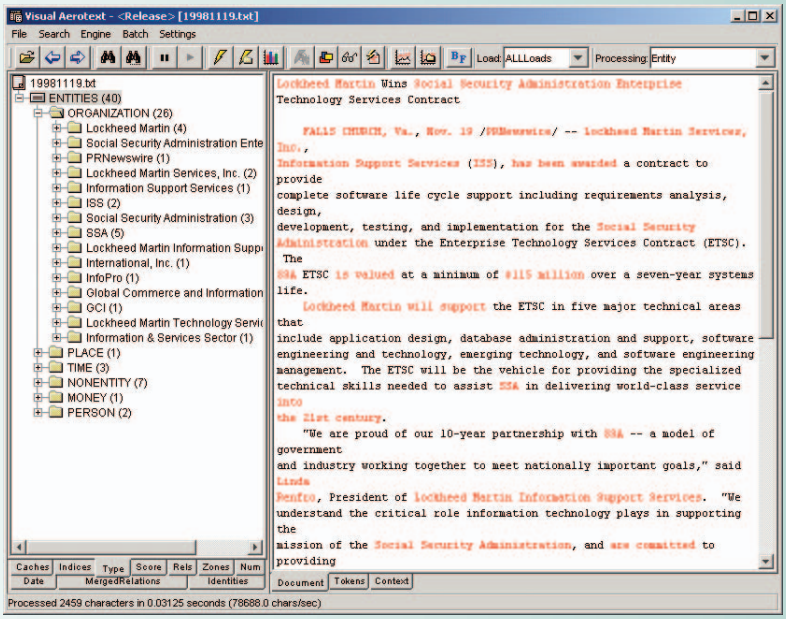
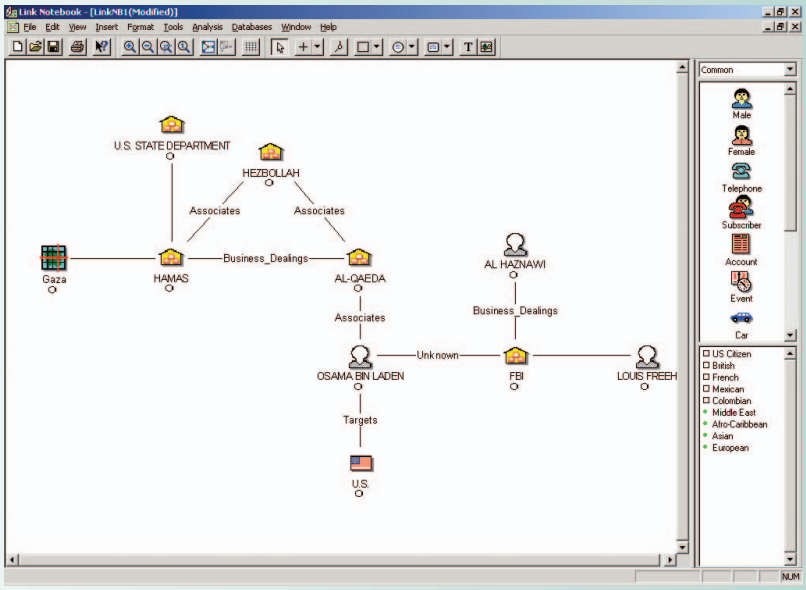


Table 1. Types and examples of entities that an IE system identifies.

Entity type	Examples
Aircraft	B-52 bomber, F-22
Drug	Cocaine
Facility	JFK Airport
Latitude and longitude	322° N-116° W
Measure	50 kilograms
Money	$500,000
Organization	IBM
Percentage	3.5
Person	John P. Smith
Phone number	(609) 522-4086
Place	West Virginia
Time	5 March 1989, 3:40 p.m., or "a week ago"
URL	http://www.lmco.com
Vehicle	Toyota Camry
Vessel	USS Cole
Weapon	AK-47

Figure 2. Extracted information from natural-language text.



name or last name) and the locations in the documents to which they refer. In some cases, the IE system physically embeds the tags into documents, bracketing the information items to which they refer. The tags, often collectively called metadata, normally identify one or more of three types of information:

- *Entities.* Entity types include person, place, or organization names; dates; and times. Other entities can be addresses, identification numbers, chemical compounds, equipment names, manufacturing processes, manufactured items, weapons, or vehicles.
- *Relationships.* A relationship defines a link between two entities. Relationships can be quite specific, such as "employed by," or general, such as "work." Examples of relationships include familial (such as parent, spouse, children, or sibling); company-level (such as owner, member of, or employee of); or positional (such as located at, born in, or lives in).
- *Events.* Events identify occurrences; examples are purchases and sales of companies, conclusions of agreements, military actions, negotiations, terrorist attacks, and meetings of world leaders.

Starting with person names and moving down the preceding list, automated extraction becomes increasingly difficult. Out-of-the-box, IE applications focus on tagging entities plus a few standard relationships; they might tackle some events of common interest. An IE component included within a larger application, such as text search, typically tags only basic entities. If a user organization wants more types of extraction—such as additional entities, more events, or better accuracy—it must tune the IE component or application to meet these requirements.

When an entity is tagged in a piece of text, the IE system normally tries to tag all references to that entity. For example, whenever "George Washington" or a variant (such as "G. Washington," "Washington," "the president," or "General Washington") appears, the tool will tag it as a person's name. Of course, within any one document, many of these references point to the same person. The IE system needs rules or patterns to figure this out, and must also separate the instances of Washington's name from references to the state or city. This coreference capability within a document is at the leading edge of IE today.

Within documents, IE systems use linguistic cues and conventions to track most coreferences.

Figure 2 shows a link diagram in i2's Analyst's Notebook, a product for displaying entities and relationships. It shows information that Lockheed Martin's AeroText extracted from natural-language text. This illustrates the true power of IE tools. Once the tool has done its job and the data is in the database, a user can display and manipulate it using database queries, spreadsheets, geographic information systems, link analysis, data mining, and online analytical processing tools. In a well-designed system, the user can move easily from one analysis tool to another, looking at the same data from multiple viewpoints. This flexibility, as well as access to larger volumes of data, provides more complete understanding and more thorough analysis than working without IE support. This is true whether the user is conducting marketing, scientific, or national security research.

Until recently, relating the information extracted from one document to similar information from another document was a task left entirely to the user. To provide automated support for deciding whether the "John Doe" of document A is the same person as the "J. Doe" of document B, someone must write special rules that reflect the user organization's needs and business rules. Unlike the rules and patterns for the initial extractions, these rules are based on business logic (such as "accept two names as the same if they differ only in one character") and operate on the database of extracted entities, not directly on the natural language in the text. Today, some IE tools incorporate such cross-document coreference rules as part

Table 2. Sample IE products.

Company	Product name	Description
Attensity	Attensity Server	Extracts relational facts, entities, and events from text-based information.
Inxight	ThingFinder	Identifies and extracts key entities such as people, dates, places, or companies from text data sources.
Lockheed Martin	AeroText	Provides an information extraction system for developing database generation, routing, browsing, summarizing, and searching applications.
SRA International	NetOwl	Extracts, summarizes, and categorizes unstructured text.

of their solutions, but these rules can be as simple as a string match.

Another type of coreference is the identification and resolution of relative time expressions, such as "yesterday," "a few months ago," "the next day," or "Monday." For these expressions to be useful, the IE system must translate them into standardized date-time expressions that account for the writer's geographic location to determine the time zone. Because of these complexities, IE products often ignore relative-time issues.

DISCRIMINATING BETWEEN IE PRODUCTS

About 15 to 20 independent IE products are on the market today, all claiming major capabilities. Table 2 lists a few of the major products. Other knowledge and information management tools might incorporate simplified IE routines built specifically for each tool. How should you choose between these capabilities? Many products, of course, are relatively new offerings from small companies that might not be stable enough for you. However, several main features distinguish one product from another.

Accuracy

The process of evaluating IE accuracy is labor-intensive, so users typically do not invest adequate time in evaluating the accuracy of IE products. Information about the US government's method for evaluating IE research systems is available at the National Institute of Standards and Technology (NIST, http://www.itl.nist.gov/iaui/894.02/related_projects/muc/). Basically, the method measures how well IE software reproduces the results that humans would provide for the same task. The method includes testing applications for missing and excessive extractions. Some tools are available to help; for example, the AeroText integrated development environment (http://www.aerotext.com) lets the evaluator create *ground truth* answer keys (which are texts that one or more human evaluators have marked with the answers that the extraction system should find) and evaluate a system's output against them within a single environment. The evaluator can also use Mitre's Alembic (http://www.mitre.org/tech/alembic-workbench/) for assistance in preparing answer keys. NIST lists other tools to automatically score the results of an IE system against a ground truth (http://www.nist.gov/speech/tools/index.htm).

Machine learning versus human analysis

IE systems rely on patterns in text to find information that the user wants. These patterns can come from an analysis by knowledge engineers or from machine learning. Many vendors and researchers are now working with hybrids of these two approaches; each has its strengths and disadvantages. Although machine-learned patterns might become the long-term solution, this method has not yet proven itself conclusively faster or better than human analysis. Also, when a machine makes a mistake in learning a pattern, you cannot correct the pattern directly. You must instead adjust the learning method to adjust the pattern. Therefore, machine learning operates best on stable, well-understood data sets.

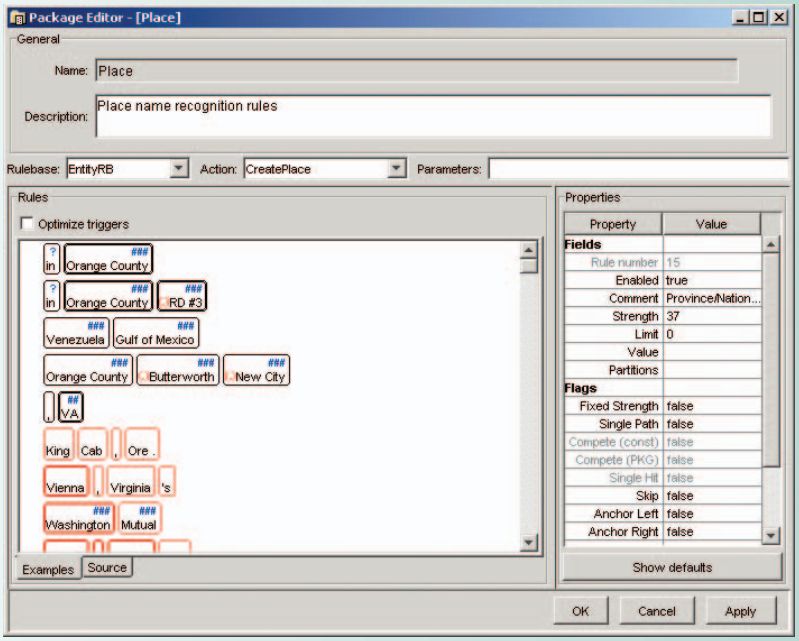
On the other hand, products based on patterns and rules derived via human analysis can achieve impressively high levels of accuracy on complex extractions. But this method requires experts to identify the patterns and develop rules based on them. As the applications for IE become more complex, the rule sets become more difficult to organize and manage effectively. Eventually, these applications could outstrip an individual rule developer's ability to easily comprehend, so that machine-learned pattern and rules development might become a necessity.

Speed of pattern development

When a general-purpose IE system is not enough, and the system requires tuning to meet user requirements, the pattern development process' efficiency can be a major factor in setting up the system. IE products differ considerably in the level of support they provide for building and changing rules. An environment that promotes extensive regression testing and eases a developer's work can speed up the process significantly, whether the vendor or your own organization develops rules.

Most IE tools incorporate at least some manually devel-

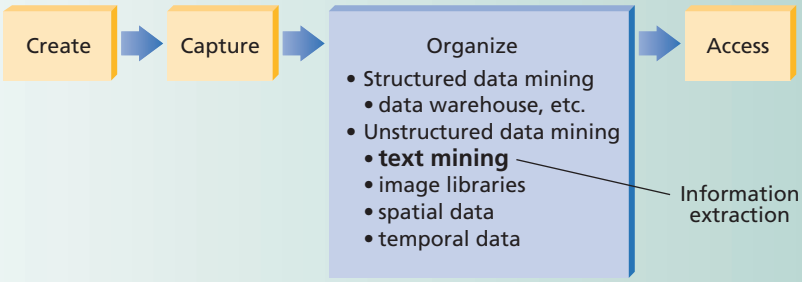
Figure 3. AeroText's graphic interface for rule building.



Example:

Street Address = StreetAddress , City , State ZipCode

Figure 4. IE's role in a complete enterprise system.



oped rules. Rule development requires knowledge of the application subject area as well as linguistic skills. Rule construction for some IE tools might also require programming expertise. A well-designed rule development environment can speed rule development and also improve rule accuracy. The ability to repeatedly test rules and evaluate their effectiveness is key. Figure 3 shows the AeroText integrated development environment's graphic interface for rule building. Using this interface, the rule developer can focus on understanding the content and structure of the text he is using, rather than making an effort to work with a specialized computer language.

Speed and ease of integration

IE products differ considerably in their speed, but speed will also differ depending upon the data and the number and complexity of the extractions. Events, for example, typically take more processing time than entities. A large number of entities typically take more time than a few entities. The output formats are a key issue for integration, as are document formats that the IE product can read. Additionally, you might need to access the offsets of the items tagged in the text; offsets record the position of the extracted item in the original text. Offsets are important because many follow-on types of analysis, such as visualization or data mining, use them to provide highlighting of extracted items in the original text, calculate proximities, look for contextual elements not defined as part of the extraction task, or to perform accuracy evaluation tasks. Thus, these types of analyses need offsets to realize their full potential.

DOES YOUR ENTERPRISE NEED IE?

To improve the accuracy of retrievals from the enterprise document store, almost any organization with an enterprise-wide text-searching capability needs IE. As I noted earlier, IE's role in improving search and other related tools is becoming more common. For the most benefit, you should look carefully at how any application you purchase incorporates IE.

As Figure 4 shows, IE plays an important role in a complete enterprise information system. IE tools, for example, are a precursor step to effective data mining of natural-language text. Extracted information can provide metadata to aid users in retrieving the documents they need. It can also help users enhance the indexing process of text search tools such as Convera's RetrievalWare or Verity's K2 Enterprise

or strengthen the text categorization of tools such as Stratify's Classification Server or Entrieva's SemioTagger. IE tools can capture extracted information in structured databases, and can more easily combine the information with the other structured information that an organization holds. For example, an organization might maintain a database of companies that have purchased its products or services. The responsible salesperson could complete records for this database. However, IE tools can track the news media for additional useful information about these same companies, such as changes in their location, ownership, officers, or investment status. The tools can automatically notify responsible salespeople and supplement database holdings with event details.

IE is a specialized capability, and the IE component of a search product should preferably come from another vendor that specializes in IE. A search product's homegrown IE tools are far less capable. You might also investigate the product's robustness as a stand-alone tool, comparing it to the integrated version. Typically, an integrated IE tool implements only a subset of the rules available in the stand-alone version. Vendors limit the integrated tool to increase speed and save storage in the overall system, resulting in fewer extractions and, in some cases lower accuracy. Additionally, you should determine whether and how you can tune the IE component to your organization's needs and data. Can you access the IE component's output independently of the search system? Some integrations never store the IE output data; it only appears as highlights or notations when a user views the text. This might be sufficient. But, if users must manipulate the material—keep lists of people and organizations, display links in a link tool, or maintain spreadsheets of sales—they need access to the IE results to automatically populate tools for these downstream tasks.

Some organizations might need a high-end, powerful, and specialized extraction capability to fill complex event and relationship databases. This remains a moderately expensive proposition, whether the organization uses machine learning or builds manually developed rules. If you are considering IE for this type of task, a cost-benefit analysis to determine its utility to your enterprise is a must. First, you should ensure that sufficient text data is available to answer your users' questions—if you only need to analyze a few pages a day, a person might do the task as efficiently as an automated application.

On the other hand, even though a high volume of material might be available, someone should ensure that the desired information is actually present in the data. But once your organization determines that it can use the application, IE can make good information workers 4 to 6 times faster. It can also help them be at least 1.5 to 2 times more thorough than when they have only basic search-and-retrieval functionality.

Information workers need training to effectively employ these new capabilities. The best policy is to start lean and mean, and plan for incremental growth over one year. As users become familiar with the new technologies, expect to make changes in the following areas: selection of material entering the application, extraction rules, database design, user interface, and analysis tools. Of these application elements, the correct database design is the most important to achieve as early as possible.

Plan for a trial period during which users perform the application task manually, using the planned database. This permits developers to identify and resolve database design issues while they can still easily change the database structure.

For Further Information

Tutorials on information extraction are sometimes available in conjunction with major conferences from professional organizations that address information retrieval and information extraction, such as the Association for Computational Linguistics' International Joint Conference on Natural Language Processing; the Association for Computing Machinery's SIGIR (Special Interest Group for Information Retrieval), and ACM's International Conference on Information and Knowledge Management. Other resources include the following:

- **"Introduction to Information Extraction Technology," Douglas E. Appelt and David J. Israel, *Proc. Int'l Joint Conf. Artificial Intelligence* (IJCAI 99), Morgan Kaufman, 1999; http://ranger.uta.edu/~alp/cse6331/ixtutorial.pdf: A tutorial introducing fundamental concepts of IE technology.**
- **Proceedings of the Message Understanding Conference (MUC), numbered 3 through 6: These are available in academic libraries. Most of these are now out of print, although MUC 5 is still available for purchase. MUC 7 proceedings are available on the NIST Web site (http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html). Morgan Kaufmann still offers Tipster Program Proceedings for Phases I and II (http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/pip.htm). Tipster was a Defense Advanced Research Projects Agency (DARPA) research program devoted to development of information retrieval and information extraction technology and sponsored the MUC for most of the 1990s.**

IE: ONE OF MANY TOOLS FOR TEXT HANDLING

Regarding their approach to language, text-handling technologies lie on a continuum. At one end are capabilities like keyword matching, which only superficially determine meaning. At the other end are capabilities like machine translation (MT), which attempt a highly detailed understanding of text. IE lies closer to the MT end of this continuum. Similar to MT, IE uses structure (syntax or grammar) and the meaning (semantics) of individual sentences and paragraphs. Unlike MT, it does not attempt to address every word of every sentence. Instead, IE focuses on phrases and sentences of importance to the user's subject area and application. Additionally, most IE applications today do not have any component that generates a natural-language output, in the way that MT must.

In the 1950s and 1960s, distinct theoretical approaches emerged to address what were then the main tasks for language-handling software—retrieval, dissemination, and MT. Early approaches to IE in the 1980s employed a syntactic parsing step, followed by a semantic analysis step that was similar to the first steps in the MT processes of the time.

However, many more target applications for language processing exist today, and users employ a wider array of techniques to accomplish them. Especially in practical applications, fewer rigid theoretical barriers exist between the different applications. For example, you can find elements of IE technologies in most of today's advanced enterprise search engines, in many categorization systems and summarization technologies, and in some MT tools. Conversely, some IE systems use approaches developed initially for search applications and for MT. This trend toward blending language-processing technologies will continue, although the mother of all language processing tools—one that can do everything users need with one integrated approach—might not exist for at least another 10 years.

For IE, the immediate future looks bright. With the continued availability of cheaper processing and storage, integration of increasingly thorough IE techniques with other text-processing applications will continue. As a result, even for general applications, enterprises will have access to precise text search and retrieval capabilities without needing to integrate additional tools into their systems. At the same time, the marriage of growing computational capability with constantly improving IE techniques will also permit highly complex IE applications that look for the greatest depths of meaning in language. These applications will still require careful tuning to extract the specified data from specific text sources. However, for the most cherished causes, such as health, safety, and security, these powerful IE applications will be critical. ■

Sarah M. Taylor *is a principal systems engineer and chief technologist for knowledge management at Lockheed Martin. Contact her at sarah.m.taylor@lmco.com.*

For further information on this or any other computing topic, visit our Digital Library at http://computer.org/publications/dlib.