The background is a complex, abstract geometric composition. It features a central, multi-faceted structure that resembles a stylized, low-poly object, possibly a piece of machinery or a futuristic architectural element. This structure is rendered in shades of white and light gray, with sharp edges and flat surfaces. From the top and sides, numerous thin, parallel lines of light radiate downwards, creating a sense of depth and perspective. The overall color palette is dark, with deep blacks and grays, which makes the white and light gray elements stand out. A prominent feature is a series of bright red, glowing lines that curve across the lower right portion of the image, adding a dynamic and energetic feel to the composition.

Information

ANDREW McCALLUM, UNIVERSITY OF MASSACHUSETTS, AMHERST

EXtraction

Distilling
Structured
Data from
Unstructured
Text

In 2001 the U.S. Department of Labor was tasked with building a Web site that would help people find continuing education opportunities at community colleges, universities, and organizations across the country. The department wanted its Web site to support fielded Boolean searches over locations, dates, times, prerequisites, instructors, topic areas, and course descriptions. Ultimately it was also interested in mining its new database for patterns and educational trends. This was a major data-integration project, aiming to automatically gather detailed, structured information from tens of thousands of individual institutions every three months.

The first and biggest problem was that much of the data wasn't available even in semi-structured form, much less normalized, structured form. Although some of the larger organizations had internal databases of their course listings, almost none of them had publicly available interfaces to their databases. The only universally available public interfaces were Web pages designed for human browsing. Unfortunately, but as expected, each organization used different text formatting. Some of these Web pages contained two-dimensional text tables; many others used a stylized collection of paragraphs for each course offering; still others had a single paragraph of English prose containing all the information about each course.

The task thus required extracting structured information from English that had been formatted in a mixture of two-dimensional layout and free-running prose—a daunting technical challenge, but one that was ultimately solved successfully. More details about the solution follow, but first, let's place this problem in context.

INFORMATION EXTRACTION TO THE RESCUE

Articles in the October 2005 issue of *ACM Queue* addressed problems with semi-structured data—data that is loosely formatted in XML or CSV (comma separated value) tables, unnormalized, in different schemas, perhaps also noisy with duplicate records. But the majority of the world's information is even less structured than this—it is in so-called “natural language text”—written English and other languages, in Web pages, corporate memos, news articles, research reports, e-mail, blogs, and historical documents.

These text documents can be effectively searched and ranked by modern search engines, but fielded searches, range-based or join-based structured queries, data mining, and decision support typically require much more detailed and fine-grained processing. The information locked in natural language must first be transformed into structured, normalized database form.

Information extraction aims to do just this—it is the process of filling the fields and records of a database from unstructured or loosely formatted text. Thus (as shown in figure 1), it can be seen as a precursor to data mining: Information extraction populates a database from unstructured or loosely structured text; data mining then discovers patterns in that database. Information extraction involves five major subtasks (which are also illustrated in figure 2):

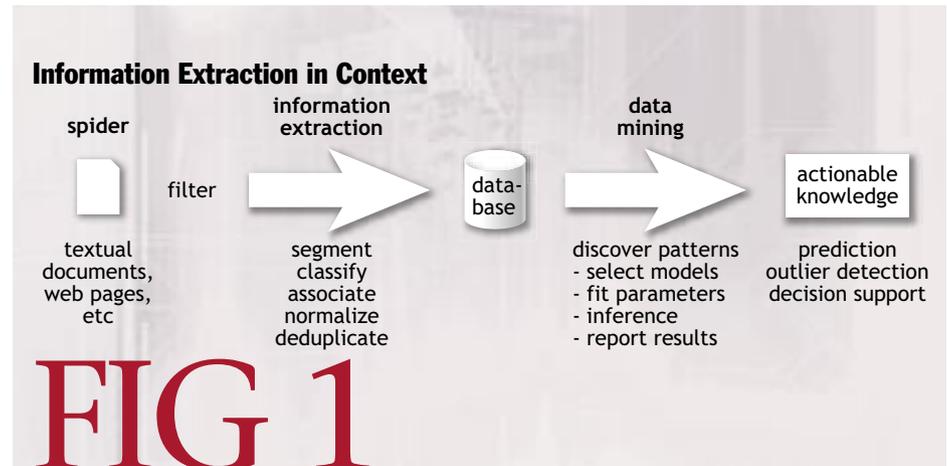
- *Segmentation* finds the starting and ending boundaries of the text snippets that will fill a database field. For example, in the U.S. Department of Labor's continuing education extraction problem, the course title must be

Information EXtraction:

Distilling
Structured
Data from
Unstructured
Text

extracted, and segmentation must find the first and last words of the title, being careful not to include extra words (“Intro to Linguistics is taught”) or to chop off too many words (“Intro to”).

- *Classification* determines which database field is the correct destination for each text segment. For example, “Introduction to Bookkeeping” belongs in the course title field, “Dr. Dallan Quass” in the course instructor field, and “This course covers...” in the course description field. Often segmentation and classification are performed at the same time (using a finite-state machine, as described in a later section).
- *Association* determines which fields belong together in the same record. For example, some courses may be described by multiple paragraphs of text, and other courses by just one; extraction must determine which field values from which paragraphs are referring to the same course. In the course extraction example, association is a fairly coarse-grained operation, but, if you are extracting records about trade negotiation meetings from news articles, then determining which governmental minister met with which other representative to talk about trade between which two countries can involve fairly subtle linguistic cues about relations and associations. This step is sometimes referred to as *relation extraction* for the case in which two entities are being associated. Commercial products that do relation extraction are rarer than those that do only segmentation and classification.
- *Normalization* puts information in a standard format in which it can be reliably compared. For example, the times for one course may be given as “2-3pm”, another as “3pm-4:30pm”, and another as “1500-1630”, but we would like a search to be able to detect any overlap. Obviously, simple string comparisons will not do the job here; the data should be converted to a standard (likely numeric) representation. Normalization is relevant to string values also; for example, given the name



The Components of Information Extraction

Original Material

Computer Science Dept, Dartmouth Course Listings Fall 2006
CS121 Java Programming
MWF 1330-1430, Sutton
An introductory course, covering conditionals, iteration, and I/O. No prerequisites.
CS383 Artificial Intelligence
TuTh 1030-1200, Cash
Logic, search, Bayesian networks, machine learning and robotics. Requires CS245.
CS392 Computational Linguistics
WF 1500-1630, Quass
Covers N-grams, hidden Markov models, parsing, and translation. Prerequisites are CS383 & Stat202.
Cross-listed as Ling380.

Linguistics Dept, Dartmouth College
Course Listings Fall 2006
Ling101 Intro to Linguistics is taught by Dr. Wei Li in Smith Hall, Rm 202. This course introduces phonology, morphology, common grammatical patterns. Meets Mondays and Wednesday from 10:30am to 12:00pm.
Ling380 Computational Linguistics, taught by Dr. Dallan Quass in Sutton Rm 102, this course covers N-grams, hidden Markov models, parsing, and machine translation. Meets Wednesdays and Fridays 3pm to 4:30pm. You must first take CS383 and Stat202. Cross-listed as CS392.

FIG 2

“Wei Li” and “Li, Wei,” a standard ordering of first and last names should be chosen. Issues of normalization may often be intertwined with deduplication, the last subtask, described next.

- *Deduplication* collapses redundant information so you don’t get duplicate records in your database. For example, a course may be cross-listed in more than one department, and thus appear on more than one Web page; it will then be extracted multiple times, but we want only one record for it in our database. In news articles this may also involve determining that “Condoleezza Rice,” “the U.S. Secretary of State,” and “Rice” are all referring to the same person, but that “Secretary of State Powell” and “Rice, Wheat, and Beans” are referring to something else. Usually, commercial products for deduplication are offered separately from segmentation, classification, and association, although later I will argue that they should be integrated. It is somewhat of

a joke in the community that this process of collapsing alternative names itself has so many different names. In the database community it is known as *record linkage* or *record deduplication*; in natural language processing it is known as *co-reference* or *anaphora resolution*; elsewhere it is known as *identity uncertainty* or *object correspondence*. In these different contexts the problem has different subtleties, but fundamentally they are all the same problem.

A TOUR OF EXAMPLE APPLICATIONS

Historically, information extraction most often has been studied for news articles from which organizations, locations, and individual names are extracted and related to each other, but more recently information extraction has been applied to many text formats, including Web pages, government reports, scientific articles, e-mail, and legal documents. There are many compelling applications of information extraction, including the U.S. Department of

Segmentation

CS121
Java Programming
MWF 1330-1430
Sutton
CS383
Artificial Intelligence
TuTh 1030-1200
Cash
CS245
CS392
Computational Linguistics
WF 1500-1630
Quass
CS383
Stat202
Ling380
Ling101
Intro to Linguistics
Dr. Wei Li
Mondays and...12:00pm
Ling380
Computational Linguistics
Dr. Dallon Quass
Wednesdays and...4:30pm
CS383
Stat202
CS392

Classification

CS121
Java Programming
MWF 1330-1430
Sutton
CS383
Artificial Intelligence
TuTh 1030-1200
Cash
CS245
CS392
Computational Linguistics
WF 1500-1630
Quass
CS383
Stat202
Ling380
Ling101
Intro to Linguistics
Dr. Wei Li
Mondays and...12:00pm
Ling380
Computational Linguistics
Dr. Dallon Quass
Wednesdays and...4:30pm
CS383
Stat202
CS392

Association

CS121
Java Programming
MWF 1330-1430
Sutton
CS383
Artificial Intelligence
TuTh 1030-1200
Cash
CS245
CS392
Computational Linguistics
WF 1500-1630
Quass
CS383
Stat202
Ling380
Ling101
Intro to Linguistics
Dr. Wei Li
Mondays and...12:00pm
Ling380
Computational Linguistics
Dr. Dallon Quass
Wednesdays and...4:30pm
CS383
Stat202
CS392

Normalization & Deduplication

CS121
Java Programming
MWF 1330-1430
Sutton
CS383
Artificial Intelligence
TuTh 1030-1200
Cash
CS245
CS392, Ling 380
Computational Linguistics
WF 1500-1630
Dallon Quass
CS383
Stat202
Ling380
Ling101
Intro to Linguistics
Wei Li
WF 1030-1200

- = Number
- = Title
- = Times
- = Instructor
- = Prerequisites
- = Crosslisting

Information EXtraction:

Distilling
Structured
Data from
Unstructured
Text

Labor's continuing education course extraction. Here are some others:

- In 2000 FlipDog.com launched as a job search Web site and made quite a splash by having twice as many job openings in its database as Monster.com did. This was possible because, rather than gathering openings from employers who "pay to post" (like a newspaper's classified ads section), FlipDog automatically extracted its job openings directly from more than 60,000 company Web sites (gathering job title, description, location, company name, application contact information, etc., as well as placing the openings in an ontology). As a side project, FlipDog also produced a once-monthly report showing the changing patterns and trends from mining this large database of job openings. Several organizations used this report to help set policy because nowhere else could they get information that was as comprehensive or up to date. Automatic extraction accuracy was usually very high, but in cases of low-confidence extraction, human verifiers were used to improve accuracy. FlipDog was later acquired by Monster.
- ZoomInfo.com extracts information about people from all over the Web, creating cross-referenced records of names, job titles, employment histories, and educational backgrounds for more than 26 million people by processing news articles, press releases, corporate bios, and other sources. The Web site is used for recruiting, sales, and corporate intelligence. On the whole, extraction accuracy is quite good, although there are some errors in segmentation and deduplication.
- CiteSeer.org extracts citation information from academic research papers, including the paper's title, authors, publication venue, year, etc.¹ It also deduplicates citation entries from papers' reference sections, so you can easily find all the papers that cite a certain paper. The resulting "citation graph" can be analyzed to automatically find the seminal papers in a subfield. It also has been used to show that papers available online tend to be cited more often than papers available only from their publishers. Other similar services include scholar.google.com and rexa.info.
- Verity.com's MediClaim can extract various fields from medical insurance claim forms, enabling semi-automated processing and faster throughput. The extraction relies on the regular layout and formatting in a standard set of forms. Other companies with extraction products

include Inxight, ClearForest, Fetch.com, and TeraGram, and specialty companies such as Burning Glass and Molecular Connections.

- William Cohen and some of his students at Carnegie Mellon University have developed several systems for information extraction from a body of e-mail messages. One system extracts signature blocks from e-mail messages,² which could then enable automated extraction of address book information.³ Another system extracts people's names. Dayne Freitag of Fair Isaac Corporation created a system to extract calendar entries from e-mail messages announcing upcoming seminar titles, speakers, locations, and times.⁴

HOW DO THEY DO THAT?

Some simple extraction tasks can be solved by writing regular expressions. Extraction from moderately more complex text sources, yet that have sufficient formatting regularity, can be addressed accurately with hand-tuned, programmed rules. For example, if you wanted to extract book titles and author names from the Web pages at Amazon.com, you could rely on the fact that they appear with exactly consistent formatting (title just under the blue bar, in bold; author hyperlinked underneath, preceded by the word *by*), and write a fairly straightforward Perl script in about 30 to 60 minutes that would do the job using these formatting regularities. But if you also wanted to gather information from BarnesandNoble.com, and thousands of other booksellers, you would have to write new rules for each one—and then rewrite them every time one of the sellers changed its Web layout.

When the human resources for this level of rule-writing (and ongoing rewriting) aren't available, or when the formatting clues are unreliable or not present, information extraction must rely on the language itself—the words, word order, grammar—perhaps also combined with whatever weak, irregular formatting clues are present. For example, the U.S. Department of Labor task previously described falls in this category. The hand-tuned rule-writing approach is sometimes used in this situation (and has been successfully used by companies such as ClearForest Corporation and SRA International); however, as the language patterns get subtler, as the exceptions of English usage pile up, and as the rules interact with each other more and more, the rule writing can get extremely complex. It is not unusual for such systems to include lit-

erally thousands of written rules, with subtle interconnections that make editing the rules extremely error-prone.

Over the past decade there has been a revolution in the use of statistical and machine-learning methods for information extraction.^{5,6,7,8,9,10} These are methods that automatically tune their own rules or parameters to maximize performance on a set of example texts that have been correctly labeled by hand. In other words, instead of trying to tune the complex extraction rules manually, you show the machine what to do on specific example texts by performing the extraction task yourself. The machine then generalizes from these examples, appropriately tuning its own rules and parameters. For complex extraction tasks, many examples may be required (on the order of hundreds or thousands), but labeling data is often still significantly easier than hand-tuning rules, and it can be done by less-skilled, part-time labor. In many cases, machine learning obtains significantly higher accuracy than human-tuned methods.

Some of these machine-learning methods use decision trees¹¹ or if-then-else rules.^{12,13} Such an approach is often followed in systems that use machine learning to create formatting-based extractors (called wrappers), as described in the Amazon.com example. Increasingly popular are machine-learning methods that use large numbers of relatively simple features of the input but assign subtly interacting, real-value weights to these features. For example, the word *said* is a weak indicator that a person's name may be coming next. These methods assign some weight with which the preceding-word-is-*said* feature votes for the next word being a person's name; then, by combining evidence from many such appropriately weighted, weakly indicative features, very accurate extraction decisions can be made.

One such statistical model with simple features is the HMM (hidden Markov model)—a finite-state machine with probabilities on the state transitions and probabilities on the per-state word emissions. HMMs became widely used in the 1990s for extraction from English prose.^{14,15} States of the machine are assigned to different database fields, and the highest-probability state path associated with a sequence of words indicates which subsequences of the words belong to those database fields.

More recently there has been interest in combining the advantages of finite-state machines with more complex features—a prospect that is enabled by conditional-probability models, including maximum entropy Markov models¹⁶ and conditional random fields.¹⁷ These models have ranked highly in information extraction competitions (for example, the BioCreative competition to extract

protein names from bioinformatics research papers). Conditional random fields have been used to integrate more of the stages of information extraction, including not only segmentation and classification, but also normalization and deduplication, using models beyond just finite-state machines.

The U.S. Department of Labor course extraction problem was solved by a company called WhizBang Labs using a combination of several machine-learning components. To find the Web pages likely to contain course listings, text classification was used in conjunction with a spider. Statistical language modeling methods hypothesized segmentations and classifications of the different fields, which also were put into a classifier responsible for coarse-scale segmentation of one course from another. A method called scoped learning was then used to learn formatting (wrapper-like) regularities on the fly from each page, without human intervention. Logistic-regression classifiers were used to complete the association and deduplication phases. (Conditional random fields were not used only because they had not yet been developed.) In the end, the project was deemed a success—data was extracted with sufficient accuracy so that it could be deposited directly into the Web site's structured database.

LIFE IS GOOD, BUT RARELY PERFECT

The accuracy of automated extraction methods varies drastically depending on the regularity of the text input and the strength of the extraction method used. Extraction from formatted, highly regular database-generated Web pages (such as those on Amazon.com) can be done with perfect accuracy. Extraction from other somewhat regular text, such as postal address blocks or research paper citations, usually has a percentage accuracy in the mid- to high-90s. Accuracies in the mid-90s are now standard for extracting names of people, companies, and locations from standard news articles. (Extracting these entity names from Web pages, however, is more difficult, yielding accuracies in the 80s.) Extraction of protein names is more difficult, since their naming scheme is more irregular; the accuracies in a recent competition were in the 80s.

Success in the association stage of extraction is generally more difficult because a correct final answer also requires correct segmentation and classification of all of the fields that should be associated. Furthermore, in many domains, such as news articles, the evidence for certain associations or relations may require understanding complex subtleties of English usage and meaning.

Information EXtraction:

Distilling
Structured
Data from
Unstructured
Text

Accuracy for relation extraction in news stories is typically in the 60s.¹⁸

Deduplication, on the other hand, often performs more accurately when there are more fields to process. Deduplication of entity names in news articles often has accuracy in the 70s or 80s.¹⁹ Deduplication of multi-field research paper citations is in the 90s.²⁰

Perfect accuracy in anything but super-regular or trivial applications will never be attained—the subtleties and exceptions of human language are too deep. Information-extraction customers who demand perfect accuracy should be reminded that existing manual practices are also full of errors. For example, typical hospital patient contact records and corporate databases are filled with typographic errors and duplicates. Even when paid specialists (linguists) are tasked with labeling text (to be used in training a machine-learning extractor), there is a surprising amount of disparity between their results on the same documents; agreement among multiple humans each manually labeling news articles for entity relations is typically only in the 70s at best. In many cases, a synergistic combination of automated methods and human processing can yield the best accuracy and throughput.²¹

SHOP AROUND BEFORE YOU BUY

Increasingly, information-extraction solutions are being made available commercially. If you are thinking about using one of these, here are some questions to ask yourself and the supplier:

- Is the product an unchangeable black box? How much can you tune the extractor to your own purposes? It may be advertised as a person-name extractor, but for what type of data did the supplier develop it? Perhaps it will work well on news articles, but very poorly on legal documents.
- If you can tune it yourself, how? By writing rules? How flexible is this rules language? What subtleties will it let you capture? Does it let you express weights or “votes” on certain outcomes? How does it capture dependencies and conflicts among the rules?
- Can you train it using machine learning? That is, if you can tune it yourself, can you do so by providing examples of data with correct answers (and have the extractor self-tune with machine learning)? What machine-learning methods are employed, and how flexible are the features it uses? Does the supplier also provide tools to

help you label data and perform error analysis?

- Is it designed mostly for leveraging HTML formatting regularities? Does this paradigm match your needs? If you need to extract fields from the middle of paragraphs, this paradigm is unlikely to work well. What is the interface for creating these extractors? Does it let you see the Web page and the results of the rule matches on the fly? Does it also use learning, simply letting you highlight different regions, and automatically learn the formatting rules from your examples? Such a tool may enable less-skilled labor to build the extractors.

You should also consider open source solutions and in-house development. Free information-extraction systems include GATE (<http://gate.ac.uk/ie/annie.html>), MALLET (<http://mallet.cs.umass.edu>), MinorThird (<http://minor-third.sourceforge.net>), and Road Runner (<http://www.dia.uniroma3.it/db/roadRunner>). There are also other open source solutions for document classification (<http://www.cs.cmu.edu/~mccallum/bow/>) and document retrieval and matching (<http://lucene.apache.org>, <http://www.lemurproject.org/indri>).

UPCOMING TRENDS AND CAPABILITIES

Information extraction has made much progress in the past decade, and further research and industrial creativity continue to push this progress. Extraction is being applied to increasingly complex problems and is being designed for more sophisticated yet easy use by nontechnical end users. Active research trends include the following:

Estimating uncertainty, managing multiple hypotheses. As already discussed, extraction will never be perfectly accurate, and some of the most problematic consequences of this occur when the final answer is the result of a cascade of processing steps, through which errors accumulate to a high, unusable level. There is increasing interest in methods that maintain multiple extraction hypotheses from one step to the other, and use probabilistic information to combine hypotheses. Rather than having errors accumulate, this approach can actually allow later processing steps to correct errors made in earlier steps.^{22,23} These methods will increase the accuracy of association and relation extraction, and enable new applications providing deeper analysis and integrating extraction directly with data mining.

Easier training, semi-supervised learning, interactive extraction. Machine-learning methods often provide the

Suggestions for Further Reading

- Appelt, D. E., Hobbs, J. R., Bear, J., Israel, D., Kameyama, M., Kehler, A., Martin, D., Myers, K., and Tyson, M. 1995. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann.
- Blei, D., Bagnell, D., and McCallum, A. 2002. Learning with scope, with application to information extraction and classification. *Uncertainty in Artificial Intelligence (UAI)*.
- Blei, D., Ng, A., and Jordan, M. 2003. Latent Dirichlet allocation. *JMLR* 3: 993–1022.
- Bruninghaus, S., and Ashley, K. D. 2001. Improving the representation of legal case texts with information extraction methods. *Proceedings of the 8th International Conference on Artificial Intelligence and Law*.
- Carreras, X., Marques, L., and Padro, L. 2002. Named entity extraction using adaboost. *Proceedings of CoNLL-2002*: 167–170.
- Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., and Slattery, S. 1998. Learning to extract symbolic knowledge from the World Wide Web. *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98)*: 509–516.
- Culotta A., and McCallum, A. 2005. Reducing labeling effort for structured prediction tasks. *AAAI*.
- Freitag, D. 1998. Information extraction from HTML: Application of a general learning approach. *Proceedings of the 15th Conference on Artificial Intelligence (AAAI-98)*: 517–523.
- Freitag, D. 1998. Machine Learning for Information Extraction in Informal Domains. Ph.D. thesis, Carnegie Mellon University.
- Haghighi, A., Toutanova, K., and Manning, C. D. 2005. A joint model for semantic role labeling. *Ninth Conference on Computational Natural Language Learning*: 173–176.
- Kushmerick, N., Weld, D., and Doorenbos, R. 1997. Wrapper induction for information extraction. *Proceedings of the 15th International Conference on Artificial Intelligence*: 729–735.
- Lawrence, S. 2001. Online or invisible? *Nature* 411(6837): 521.
- McCallum, A., Nigam, K., Rennie, J., and Seymore, K. 2000. Automating the construction of Internet portals with machine learning. *Information Retrieval Journal* 3: 127–163.
- McCallum, A., and Wellner, B. 2004. Conditional models of identity uncertainty with application to noun co-reference. *Neural Information Processing Systems (NIPS)*.
- McDonald, R., and Pereira, F. 2005. Identifying gene and protein mentions in text using conditional random fields. *BMC Bioinformatics* 6 (Supplement 1).
- Miller, S., Guinness, J., and Zamanian, A. 2004. Name tagging with word clusters and discriminative training. *Proceedings of HLT-NAACL*.
- Minkov, E., Wang, R., and Cohen, W. 2004. Extracting personal names from e-mails: Applying named entity recognition to informal text. In preparation; <http://www.cs.cmu.edu/wcohen/pubs-x.html>.
- Muslea, I., Minton, S., and Knoblock, C. 1998. Stalker: Learning extraction rules for semi-structured, Web-based information sources. *Proceedings of the AAAI Workshop on AI and Information Integration*.
- Ng, V., and Cardie, C. 2002. Improving machine learning approaches to co-reference resolution. *Proceedings of the Fortieth Annual Meeting of the Association for Computational Linguistics*.
- Pasula, H., Marthi, B., Milch, B., Russell, S., and Shpitser, I. 2002. Identity uncertainty and citation matching. *Advances in Neural Information Processing*.
- Pinto, D., McCallum, A., Lee, X., and Croft, W. B. 2003. Combining classifiers in text categorization. *Submitted to SIGIR '03: Proceedings of the Twenty-sixth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.
- Ray, S., and Craven, M. 2001. Representing sentence structure in hidden Markov models for information extraction. *Proceedings of the 17th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann.
- Soderland, S. 1997. Learning to extract text-based information from the World Wide Web. *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*.
- Sutton, C., and McCallum, A. 2005. Composition of conditional random fields for transfer learning. *Empirical Methods in Natural Language Processing*.
- Yeh, A., Morgan, A., Colosimo, M., and Hirschman, L. 2005. BioCreative task 1a: Gene mention finding evaluation. *BMC Bioinformatics* 6 (Supplement 1).
- Zelenko, D., Aone, C., and Richardella, A. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research* 3: 1083–1106.

Information EXtraction:

Distilling
Structured
Data from
Unstructured
Text

most accurate extractors, but gathering the necessary training data can be time-consuming and tedious—especially if impatient end users are the ones doing the data labeling. New research in semi-supervised machine learning reduces the amount of required labeled data by cleverly (some would say nearly magically) leveraging large quantities of unlabeled data to improve learning efficiency.^{24,25} Also, methods called “interactive information extraction” begin with imperfect automatically labeled data, then make manual corrections faster by highlighting low-confidence fields, and furthermore, use soft constraints to automatically correct additional fields after the human corrects just one.²⁶

AN ALTERNATIVE VARIATION: MINE THE TEXT DIRECTLY

This article has mostly discussed traditional information extraction that builds a structured database. When the goal, however, is topical trend analysis, or a rough summary of a large collection of documents, an interesting alternative is to use a loose mixture of text extraction and data mining. These are methods that leverage whatever limited structured information is available (such as the dates, senders, and recipients of e-mail messages, or simply document boundaries) and then use data-mining techniques that are robust enough to operate directly on the raw text associated with this limited structure.

For example, the latent Dirichlet allocation is a document-clustering method that gives a bird’s-eye view of the topics discussed in a document collection (the topics are represented by collections of automatically discovered, prominent keywords). The Author-Recipient-Topic model²⁷ gives a topical summary of a large collection of e-mail, identifying prominent senders and recipients associated with different topics, and identifying people in this e-mail social network who have similar roles. For example, given the text in the body of the e-mail messages, as well as the people in the To, From, and CC headers, the Author-Recipient-Topic model automatically discovers a set of topics that summarizes the communications within a social network, as well as who talked about which topics to whom. Among the applications for this model would be expert-finding in large corporations.

The Group-Topic model²⁸ discovers groups rather than roles, and has been used to identify associations between topics and like-minded legislators by mining 16 years of voting records in the U.S. Senate, along

with the text of the corresponding bills. Kleinberg’s Word Burst algorithm automatically detects trends over time;²⁹ for example, when applied to the text of U.S. Presidential State of the Union addresses, it shows the word *slaves* bursting from 1859-1863, the word *atomic* bursting from 1947-1959, and *inflation* from 1971-1980. Intelliseek.com has applied similar methods to trend analysis from blogs.

INFORMATION EXTRACTION, THE WEB, AND THE FUTURE

The World Wide Web is the world’s largest repository of knowledge, and it is being constantly augmented and maintained by millions of people. However, it is in a form intended for human reading, not in a database form with records and fields that can be easily manipulated and understood by computers. In spite of the promise of the Semantic Web, the use of English and other natural language text will continue to be a major medium for communication and knowledge accumulation on the Web, in e-mail, news articles, and elsewhere.

Eventually we will reach the point at which the answer to almost any question will be available online somewhere, but we will have to wade through more and more material to find it. The next step in improved search tools will be a transition from keyword search on documents to higher-level queries: queries where the search hits will be objects, such as people or companies instead of simply documents; queries that are structured and return information that has been integrated and synthesized from multiple pages; and queries that are stated as natural language questions (“Who were the first three female U.S. Senators?”) and answered with succinct responses.

The first half of the Internet revolution consisted of the creation of a wide area network for easy data sharing, enabling human access to an immense store of knowledge and services. The second half of the Internet revolution has yet to come. It will happen when there is machine access to this immense knowledge base, and we are thus able to perform pattern analysis, knowledge discovery, reasoning, and semi-automated decision-making on top of it. Information extraction will be a key part of the solution making this possible. ☐

REFERENCES

1. McCallum, A., Corrada-Emanuel, A., and Wang, X. 2005. Topic and role discovery in social networks.

- International Joint Conferences on Artificial Intelligence.*
2. Collins, M., and Singer, Y. 1999. Unsupervised models for named entity classification.
 3. Lafferty, J., McCallum, A., and Pereira, F. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the ICML*: 282–289.
 4. Klein, D., Smarr, J., Nguyen, H., and Manning, C. 2003. Named entity recognition with character-level models. *Proceedings of the Seventh Conference on Natural Language Learning.*
 5. Wang, X., Mohanty, N., and McCallum, A. 2005. Group and topic discovery from relations and text. In *Workshop on Link Discovery (LinkKDD)*, Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
 6. Bikel, D. M., Miller, S., Schwartz, R., and Weischedel, R. 1997. Nymble: A high-performance learning name-finder. *Proceedings of ANLP*: 194–201.
 7. McCallum, A., and Jensen, D. 2003. A note on the unification of information extraction and data mining using conditional-probability, relational models. *IJCAI Workshop on Learning Statistical Models from Relational Data.*
 8. Lawrence, S., Giles, C. L., and Bollacker, K. 1999. Digital libraries and autonomous citation indexing. *IEEE Computer* 32(6): 67–71.
 9. Soderland, S., and Lehnert, W. G. 1994. Corpus-driven knowledge acquisition for discourse analysis. *AAAI*.
 10. Kleinberg, J. 2002. Bursty and hierarchical structure in streams. *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*.
 11. See reference 5.
 12. Carvalho, V. R., and Cohen, W. W. 2004. Learning to extract signature and reply lines from e-mail. *Conference on E-mail and Spam (CEAS)*.
 13. Califf, M. E., and Mooney, R. 1999. Relational learning of pattern-match rules for information extraction. *Proceedings of the National Conference on Artificial Intelligence.*
 14. See reference 6.
 15. See reference 4.
 16. See reference 7.
 17. See reference 8.
 18. Freitag, D., and McCallum, A. K. 1999. Information extraction with HMMs and shrinkage. *Proceedings of the AAAI Workshop on Machine Learning for Information Extraction.*
 19. Roth, D., and Yih, W. 2002. Probabilistic reasoning for entity and relation recognition. *COLING*.
 20. See reference 1.
 21. See reference 3.
 22. Nahm, U. Y., and Mooney, R. J. 2000. A mutually beneficial integration of data mining and information extraction. *AAAI/IAAI*: 627–632.
 23. See reference 9.
 24. Culotta, A., and Sorensen, J. 2004. Dependency tree kernels for relation extraction. *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
 25. Ando, R. K., and Zhang, T. 2005. A high-performance semi-supervised learning method for text chunking. *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
 26. See reference 3.
 27. McCallum, A., Freitag, D., and Pereira, F. 2000. Maximum entropy Markov models for information extraction and segmentation. *Proceedings of ICML*: 591–598.
 28. Wellner, B., McCallum, A., Peng, F., and Hay, M. 2004. An integrated, conditional model of information extraction and co-reference with application to citation matching. *Conference on Uncertainty in Artificial Intelligence (UAI)*.
 29. Kristjansson, T., Culotta, A., Viola, P., and McCallum, A. 2004. Interactive information extraction with conditional random fields. *Nineteenth National Conference on Artificial Intelligence.*

LOVE IT, HATE IT? LET US KNOW

feedback@acmqueue.com or www.acmqueue.com/forums

ANDREW McCALLUM is an associate professor at the University of Massachusetts, Amherst, and director of the Information Extraction and Synthesis Laboratory. He was previously vice president of research and development at WhizBang Labs, a company that used machine learning for information extraction from the Web. In the late 1990s he was a research scientist at Justsystem Pittsburgh Research Center, where he spearheaded the creation of CORA, an early research paper search engine that used machine learning for spidering, extraction, classification, and citation analysis. He was a post-doctoral fellow at Carnegie Mellon University after receiving his Ph.D. from the University of Rochester in 1995. He is an action editor for the *Journal of Machine Learning Research*. For the past 10 years, McCallum has been active in research on statistical machine learning applied to text, especially information extraction, document classification, finite state models, semi-supervised learning, and social network analysis. (www.cs.umass.edu/mccallum).
© 2005 ACM 1542-7730/05/1100 \$5.00