Technology | DOI:10.1145/1400181.1400187 Alex Wright

Searching the Deep Web

While the Semantic Web may be a long time coming, Deep Web search strategies offer the promise of a semantic Web.

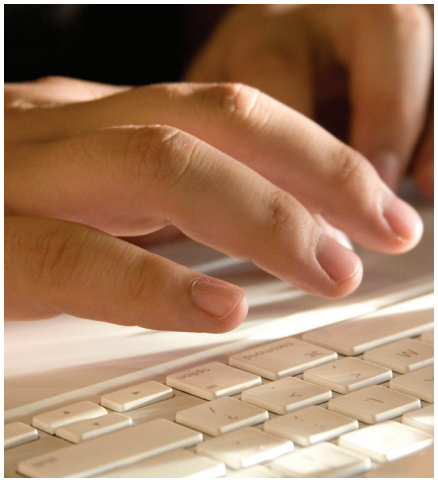
THE WEB IS bigger than it looks. Beyond the billions of pages that populate the major search engines lies an even vaster, hidden Web of data: classified ads, library catalogs, airline reservation systems, phone books, scientific databases, and all kinds of other information that remains largely concealed from view behind a curtain of query forms. Some estimates have pegged the size of the Deep Web at up to 500 times larger than the Surface Web (also known as the Shallow Web) of static HTML pages.

Researchers have been trying to crack the Deep Web for years, but most of those efforts to date have focused on building specialized vertical applications like comparison shopping portals, business intelligence tools, or top-secret national security projects that scour hard-to-crawl overseas data sources. These projects have succeeded largely by targeting narrow domains where a search application can be fine-tuned to query a relatively small number of databases and return highly targeted results.

Bringing Deep Web search techniques to bear on the public Web poses a more difficult challenge. While a few high-profile sites like Amazon or YouTube provide public Web services or custom application programming interfaces that open their databases to search engines, many more sites do not. Multiply that problem by the millions of possible data sources now connected to the Web—all with different form-handling rules, languages, encodings, and an almost infinite array of possible results—and you're have one tough assignment. "This is the most interesting data integration problem imaginable," says Alon Halevy, a former University of Washington computer science professor who is now leading a Google team trying to solve the Deep Web search conundrum.

Deep Web Search 101

There are two basic approaches to searching the Deep Web. To borrow a fishing metaphor, these approaches might be described as trawling and angling. Trawlers cast wide nets and pull them to the surface, dredging up whatever they can find along the way. It's a brute force technique that, while inelegant, often yields plentiful results. Angling, by contrast, requires more skill. Anglers cast their lines with precise techniques in carefully chosen locations. It's a difficult art to master, but



when it works, it can produce more satisfying results.

The trawling strategy—also known as warehousing or surfacing—involves spidering as many Web forms as possible, running queries and stockpiling the results in a searchable index. While this approach allows a search engine to retrieve vast stores of data in advance, it also has its drawbacks. For one thing, this method requires blasting sites with uninvited queries that can tax unsuspecting servers. And the moment data is retrieved, it becomes instantly becomes out of date. "You're force-fitting dynamic data into a static document model," says Anand Rajaraman, a former student of Halevy's and co-founder of search startup Kosmix. As a result, search queries may return incorrect results.

The angling approach—also known as mediating—involves brokering a search query in real time across multiple sites, then federating the results for the end user. While mediating produces more timely results, it also has some drawbacks. Chief among these is determining where to plug a given set of search terms into the range of possible input fields on any given Web form. Traditionally, mediated search engines have relied on developing custom "wrappers" that serve as a kind of Rosetta Stone for each data source. For example, a wrapper might describe how to query an online directory that accepts inputs for first name and last name, and returns a mailing address as a result. At Vertica Systems, engineers create these wrappers by hand, a process that usually takes about 20 minutes per site. The wrappers are then added to a master ontology stored in a database table. When users enter a search query, the engine converts the output into Resource Description Framework (RDF), turning each site into, effectively, a Web service. By looking for subject-verb-object combinations in the data, engineers can create RDF triples out of regular Web search results. Vertica founder Mike Stonebraker freely admits this hands-on method, however, has limitations. "The problem with our approach is that there are millions of Deep Web sites," he says. "It won't scale." Several search engines are now experimenting with approaches for developing automated wrappers that can scale to accommodate the vast number of Web forms available across the public Web.

The other major problem confronting mediated search engines lies in determining which sources to query in the first place. Since it would be impossible to search every possible data source at once, mediated search engines must identify precisely which sites are worth searching for any given query.

"You can't indiscriminately scrub dynamic databases," says former BrightPlanet CEO Mike Bergman. "You would not want to go to a recipe site and ask

PHOTOGRAPH BY AMANCAY MAAHS

about nuclear physics." To determine which sites to target, a mediated search engine has to run some type of textual analysis on the original query, then use that interpretation to select the appropriate sites. "Analyzing the query isn't hard," says Halevy. "The hard part is figuring out which sites to query."

At Kosmix, the team has developed an algorithmic categorization technology that analyzes the contents of users' queries—requiring heavy computation at runtime—and maps it against a taxonomy of millions of topics and the relationships between them, then uses that analysis to determine which sites are best suited to handle a particular query. Similarly, at the University of Utah's School of Computing, assistant professor Juliana Freire is leading a project team working on crawling and indexing the entire universe of Web forms. To determine the subject domain of a particular form, they fire off sample queries to develop a better sense of the content inside. "The naïve way would be to query all the words in the dictionary," says Freire. "Instead we take a heuristic-based approach. We try to reverse-engineer the index, so we can then use that to build up our understanding of the databases and choose which words to search." Freire claims that her team's approach allows the crawler to retrieve better than 90% of the content stored in each targeted site.

Google's Deep Web search strategy has evolved from a mediated search technique that originated in Halevy's work at Transformic (which was acquired by Google in 2005), but has since evolved toward a kind of smart warehousing model that tries to accommodate the sheer scale of the Web as a whole. "The approaches we had taken before [at Transformic] wouldn't work because of all the domain engineering required," says Halevy.

Instead, Google now sends a spider to pull up individual query forms and indexes the contents of the form, analyzing each form for clues about the topic it covers. For example, a page that mentions terms related to fine art would help the algorithm guess a subset of terms to try, such as "Picasso," "Rembrandt," and so on. Once one of those terms returns a hit, the search engine can analyze the results and refine its model of what the database contains.

Rather than relying on Web site owners to mark up their data, couldn't search engines simply do it for them?

"At Google we want to query any form out there," says Halevy, "whether you're interested in buying horses in China, parking tickets in India, or researching museums in France." When Google adds the contents of each data source to its search engine, it effectively publishes them, enabling Google to assign a PageRank to each resource. Adding Deep Web search resources to its index—rather than mediating the results in real time—allows Google to use Deep Web search to augment its existing service. "Our goal is to put as much interesting content as possible into our index," says Halevy. "It's very consistent with Google's core mission."

A Deep Semantic Web?

The first generation of Deep Web search engines were focused on retrieving documents. But as Deep Web search engines continue to penetrate the far reaches of the database-driven Web, they will inevitably begin trafficking in more structured data sets. As they do so, the results may start to yield some of the same benefits of structure and interoperability that are often touted for the Semantic Web. "The manipulation of the Deep Web has historically been at a document level and not at the level of a Web of data," says Bergman. "But the retrieval part is indifferent to whether it's a document or a database."

So far, the Semantic Web community has been slow to embrace the challenges of the Deep Web, focusing primarily on encouraging developers to embrace languages and ontology definitions that can be embedded into documents rather than incorporated at a database level. "The Semantic Web has been focused on the Shallow Web," says Stonebraker, "but I would be thrilled to see the Semantic Web community focus more on the Deep Web."

Some critics have argued that the Semantic Web has been slow to catch on because it hinges on persuading data owners to structure their information manually, often in the absence of a clear economic incentive for doing so. While the Semantic Web approach may work well for targeted vertical applications where there is a built-in economic incentive to support expensive mark-up work (such as biomedical information), such a labor-intensive platform will never scale to the Web as a whole. "I'm not a big believer in ontologies because they require a lot of work," says Freire. "But by clustering the attributes of forms and analyzing them, it's possible to generate something very much like an ontology."

While the Semantic Web may be a long time coming, Deep Web search strategies hold out hope for the possibility of a semantic Web. After all, Deep Web search inherently involves structured data sets. Rather than relying on Web site owners to mark up their data, couldn't search engines simply do it for them?

Google is exploring just this approach, creating a layer of automated metadata based on analysis of the site's contents rather than relying on site owners to take on the cumbersome task of marking up their content. Bergman's startup, Zitgist, is exploring a concept called Linked Data, predicated on the notion that every bit of data available over the Web could potentially be addressed by a Uniform Resource Indicator. If that vision came to fruition, it would effectively turn the entire Web into a giant database. "For more than 30 years, the holy grail of IT has been to eliminate stovepipes and federate data across the enterprise," says Bergman, who thinks the key to joining Deep Web search with the Semantic Web lies in RDF. "Now we have a data model that's universally acceptable," he says. "This will let us convert legacy relational schemas to http."

Will the Deep Web and Semantic Web ever really coalesce in the real world of public-facing Web applications? It's too early to say. But when and if that happens, the Web may just get a whole lot deeper. C

Alex Wright is a writer and information architect who lives and works in New York City.