The Deep Web

Definition (Deep Web, Hidden Web)
All the content of the Web that is not directly accessible through hyperlinks. In particular: HTML forms, Web services.

Size estimate

- [Bri00] 500 times more content than on the surface Web!

Dozens of thousands of databases.

- [HPWC07] ~ 400 000 deep Web databases.

Sources of the Deep Web

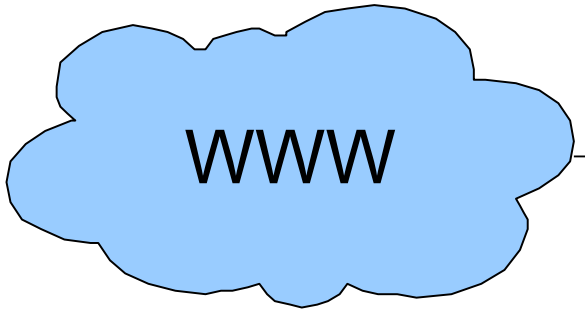
Examples

- *Yellow Pages* and other directories;
- Library catalogs;
- Publication databases;
- Weather services;
- Geolocalization services;
- US Census Bureau data;
- etc.

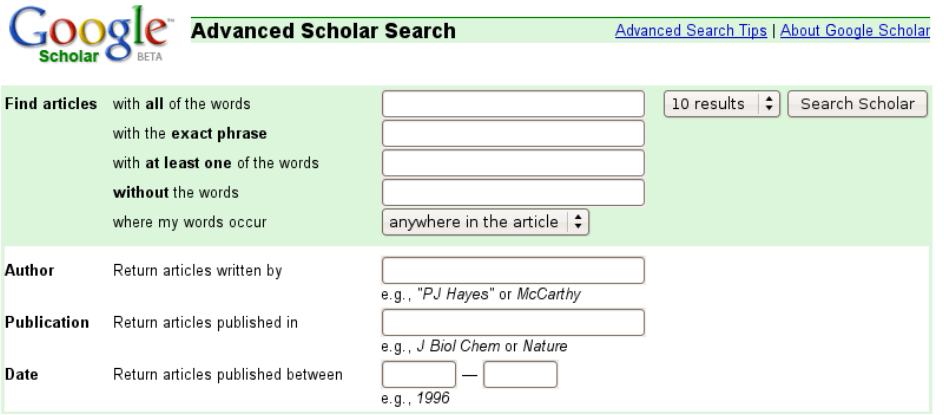
Discovering Knowledge from the Deep Web

- Content of the deep Web hidden to classical Web search engines (they just follow links)
- But very valuable and high quality!
- Even services allowing access through the surface Web (e.g., e-commerce) have more semantics when accessed from the deep Web
- How to benefit from this information?
- How to do it automatically, in an unsupervised way?

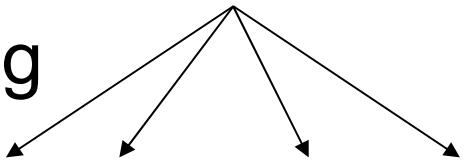
Extensional Approach



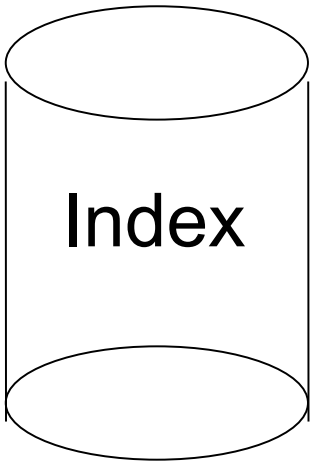
discovery



siphoning



bootstrap

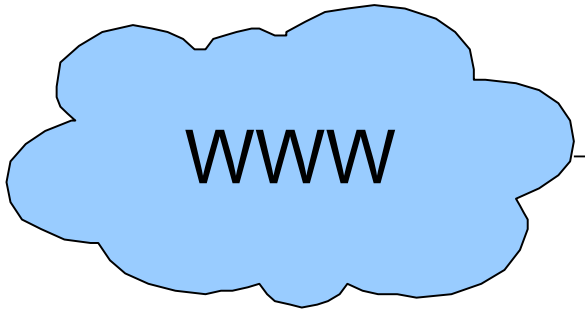


indexing

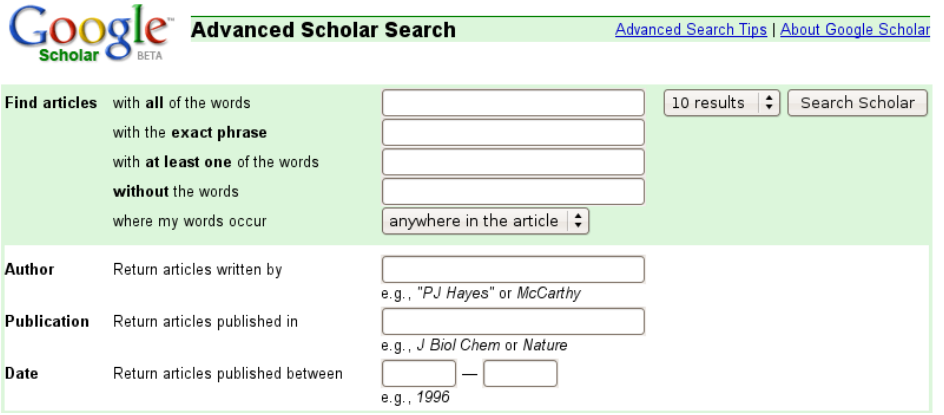
Notes on the Extensional Approach

- Main issues:
  - Discovering services
  - Choosing appropriate data to submit forms
  - Use of data found in result pages to bootstrap the siphoning process
  - Ensure good coverage of the database
- Approach favored by Google [MHC+06], used in production
- Not always feasible (huge load on Web servers)

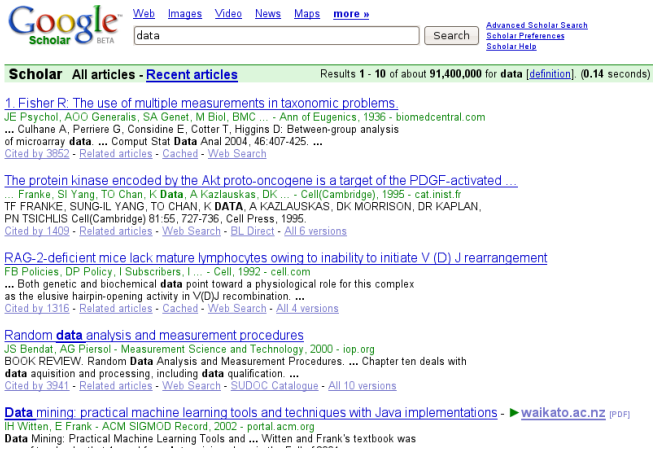
Intensional Approach



discovery



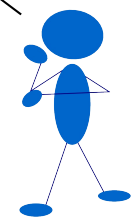
probing



analyzing

Form wrapped as a Web service

query



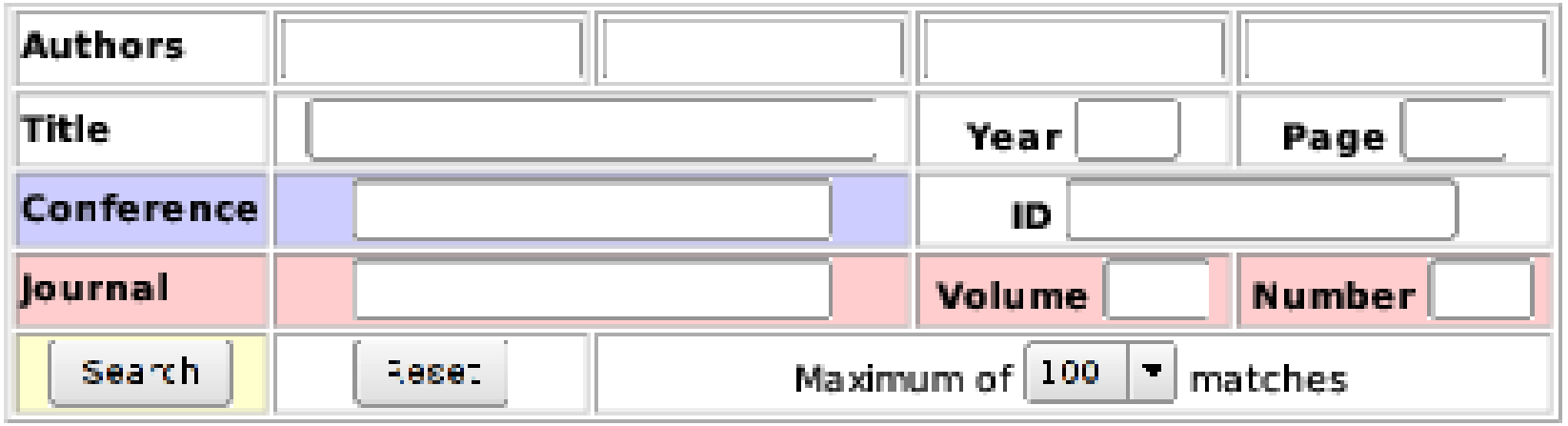
Notes on the Intensional Approach

- More ambitious [CHZ05, SMM+08]
- Main issues:
  - Discovering services
  - Understanding the structure and semantics of a form
  - Understanding the structure and semantics of result pages (wrapper induction)
  - Semantic analysis of the service as a whole
- No significant load imposed on Web servers

Discovering deep Web forms

- Crawling the Web and selecting forms
- But not all forms!
  - Hotel reservation
  - Mailing list management
  - Search within a Web site
- **Heuristics**: prefer GET to POST, no password, no credit card number, more than one field, etc.
- Given domain of interest: use focused crawling to restrict to this domain

Web forms



- **Simplest case**: associate each form field with some domain concept
- **Assumption:** fields independent from each other (not always true!), can be queried with words that are part of a domain instance

Structural analysis of a form (1/2)

- Build a context for each field:
  - `label` tag;
  - `id` and `name` attributes;
  - text immediately before the field.
- Remove stop words, stem
- Match this context with concept names or concept ontology
- Obtain in this way candidate annotations

Structural analysis of a form (2/2)

For each field annotated with concept *c*:

- Probe the field with nonsense word to get an error page
- Probe the field with instances of concept *c*
- Compare pages obtained by probing with the error page (e.g., clustering along the DOM tree structure of the pages), to distinguish error pages and result pages
- Confirm the annotation if enough result pages are obtained

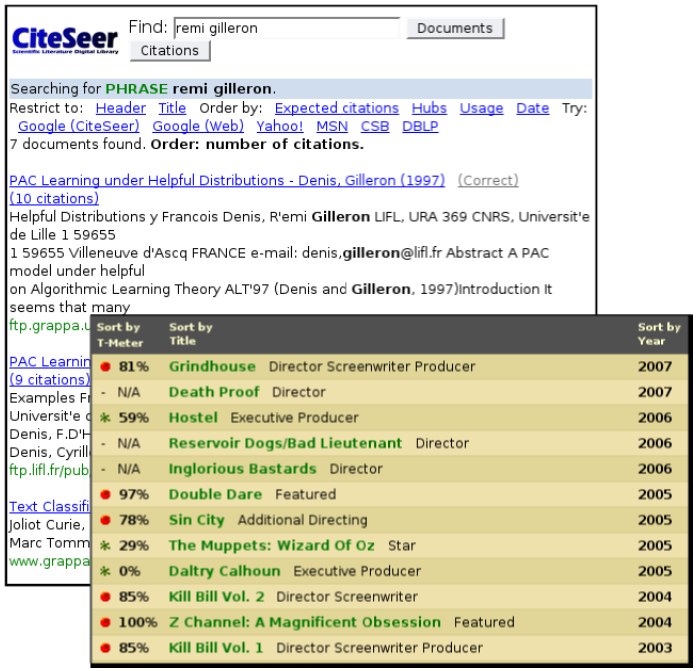
Bootstrapping the siphoning

- Siphoning (or probing) a deep Web database requires many relevant data to submit the form with
- **Idea**: use most frequent words in the content of the result pages
- Allows bootstrapping the siphoning with just a few words!

Inducing wrappers from result pages

Pages resulting from a given form submission:

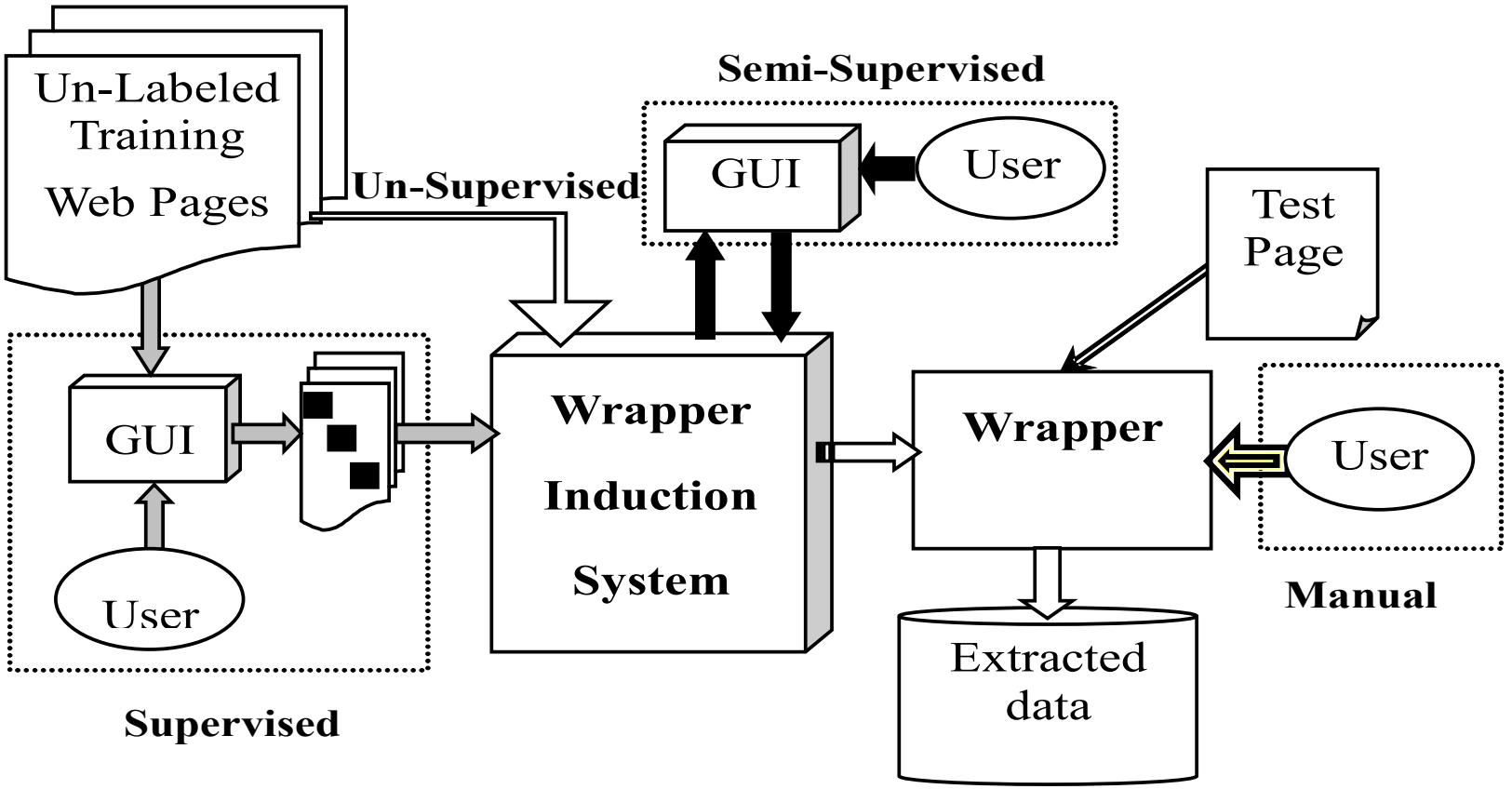
- share the same structure
- set of records with fields
- unknown presentation!



Goal

Building wrappers for a given kind of result pages, in a fully automatic way.

Information extraction systems
[CKGS06]



Unsupervised Wrapper Induction

- Use the (repetitive) structure of the result pages to infer a wrapper for all pages of this type
- Possibly: use in parallel with annotation by recognized concept instances to learn with both the structure and the content

Some perspectives

- Dealing with complex forms (fields allowing Boolean operators, dependencies between fields, etc.)
- Static analysis of JavaScript code to determine which fields of a form are required, etc.
- A lot of this is also applicable to Web 2.0/AJAX applications

References

[Bri00] BrightPlanet. **The deep Web: Surfacing hidden value**. White paper, July 2000.

[CHZ05] K. C.-C. Chang, B. He, and Z. Zhang. **Towards large scale integration: Building a metaquerier over databases on the Web**. In *Proc. CIDR,* Asilomar, USA, Jan. 2005.

[CKGS06] C.-H. Chang, M. Kayed, M. R. Girgis, and K. F. Shaalan. **A survey of Web information extraction systems**. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1411-1428, Oct. 2006.

[CMM01] V. Crescenzi, G. Mecca, and P. Merialdo. **Roadrunner: Towards automatic data extraction from large Web sites**. In *Proc. VLDB*, Roma, Italy, Sep. 2001.

[HPWC07] B. He, M. Patel, Z. Zhang, and K. C.-C. Chang. **Accessing the deep Web: A survey**. *Communications of the ACM*, 50(2):94–101 May 2007.

[MHC+06] J. Madhavan, A. Y. Halevy, S. Cohen, X. Dong, S. R. Jeffery, D. Ko, and C. Yu. **Structured data meets the Web: A few observations**. *IEEE Data Engineering Bulletin*, 29(4):19–26, Dec. 2006.

[SMM+08] P. Senellart, A. Mittal, D. Muschick, R. Gilleron et M. Tommasi, **Automatic Wrapper Induction from Hidden-Web Sources with Domain Knowledge**. In *Proc. WIDM*, Napa, USA, Oct. 2008.