Crawling the Hidden Web

by S. Raghavan & H.G. Molina

Presenter: Jack Ng
Oct 30, 2002

Background Info

- ***Hidden Web*** - databases whose content is accessible only through search forms
- Why is it important to tap into the hidden Web?

Background Info

- According to "*The Deep Web: Surfacing Hidden Value*", 2001:
  - 500 billion documents; 500 times > PIW
  - 7500 TB of data; 19 TB for PIW
  - grows much faster than the PIW
  - High quality, topic-specific information
  - 95 % is publicly accessible - no fees or subscriptions

Background Info

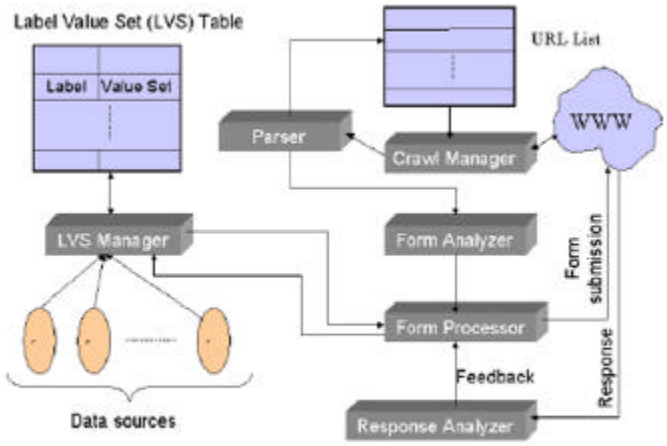
- Challenges faced by crawlers to extract content from the hidden Web:
  - size of hidden Web is enormous!
  - content not reachable by following hypertext links
  - "form-filling" is a human activity

"Training" a crawler is very difficult!!

Background Info

- Authors' approach to address the challenges:
  - task-specificity
  - human-assistance
- Propose:
  - model of hidden Web crawler
  - model of form page
  - LITE (Layout-based Information Extraction Technique) for content extraction
- Implementation - HiWE (Hidden Web Exposer)

HiWE Architecture



HiWE Data Structures - LVS Table

- Task-specific DB
- Organized by concepts
- Vocabularies for filling out forms

Fuzzy set: membership function assigns 'confidence' to each value

Task: search for game reviews

Platform	{Xbox, PS2, GameCube, PC}
Genre	{action, RPG, strategy, sports}
Developer	{EA, Sega, Squaresoft, Bioware}
Release date	{1999, 2000, 2001, 2002}

HiWE - Form Processing Strategy

- Given internal form representation:

$$F = (\{E_1, E_2, \ldots, E_n\}, S, M)$$

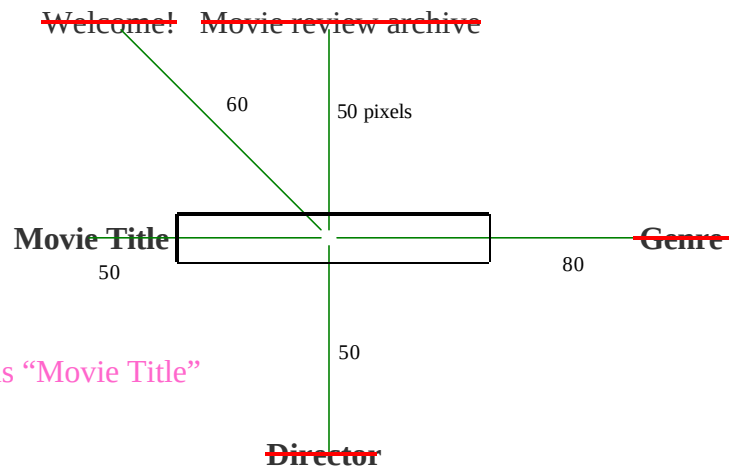
- For each infinite domain element, label matching algorithm finds closest match in LVS table and assigns value set to it
- Rank value assignments to ensure quality submission
  - *Fuzzy conjunction* --> conservative
  - *Average*
  - *Probabilistic* --> aggressive
- Submit only if rank is greater than threshold

What is LITE?

- Label extraction *heuristics* based on how page is laid out for *human* viewing
- *Idea:* label is often visually adjacent to widget (e.g., textbox) and obvious to viewer
- Partial layout is sufficient to determine adjacency --> prune unnecessary elements (see Figure 4 in paper)
- Applications in HiWE:
  - form page analysis
  - response page analysis

LITE Application - Form Page Analysis

How LITE heuristics identify label of form element



Label is “Movie Title”

LITE Application - Response Page Analysis

- Based on idea that results must be obvious to viewers
- Prune page to find visually *center-most* portion & intrepret it as results location
- To identify error pages:
  - search center portion for common error text (e.g., "No results")
  - compute hash value for center portion
    - common hash values = error pages

Experimental Results

- Value assignment ranking
  - *fuzzy conj.* --> best submission efficiency
  - *average* ✓ --> most successful submissions
  - *probabilistic* --> poor performance
- LITE outperforms other label extraction techniques; overall **93 %** accuracy

Thoughts...

- Strength & novelty of solution
  - +flexible framework
  - +works with non-cooperative DBs
  - +crawler has learning capability
  - +crawls both PIW and hidden Web
  - +‘mines’ visual layout info for semantics

Thoughts...

- *Implementation* limitations
  - LVS table - how to handle semantically ambiguous labels?
  - what about image labels?
  - doesn't consider relationships among elements when assigning values
  - 'all-or-none' form submission policy

Thoughts...

- Presentation of paper
  - easy to follow and understand
  - right level of details
  - goals & pre-conditions clearly defined

- overall, a good paper!