UTILITY-BASED CONTROL FEEDBACK IN A DIGITAL LIBRARY SEARCH ENGINE: CASES IN CITESEERX

Jian Wu, Alexander Ororbia, Kyle Williams, Madian Khabsa, Zhaohui Wu, C. Lee Giles

CiteSeer^{X}

Pennsylvania State University

Outline

- Introduction
  - Utility-based control feedback
- Three types of feedback paradigms
  - User-Correction (Metadata correction)
  - Ill-Conditioned Metadata Detection (Metadata correction)
  - Crawl Whitelist Generation (Crawl Coverage)
- Future Work
  - Crawl Scheduler
  - Dynamical Topic-Driven Crawling

Introduction

- High-level policies:

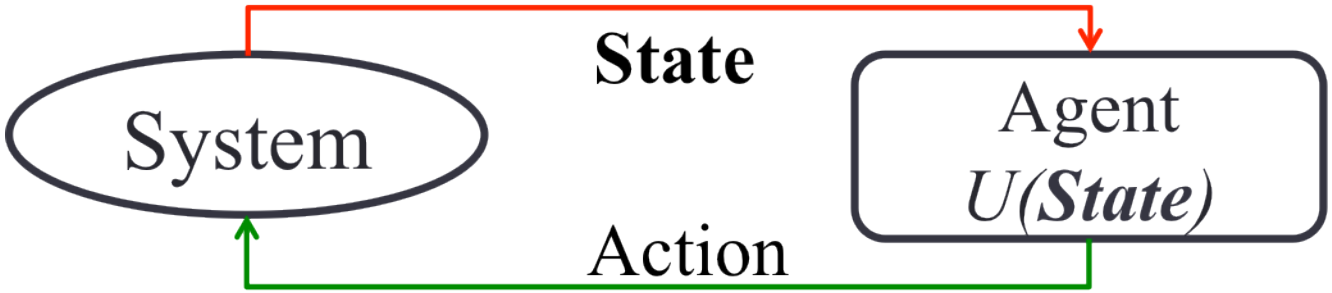


Figure: The utility-based control feedback loop.

- System State – S: service attribute vector
- Utility function – $U(S)$: maps any possible system state (S) to a scalar value
- Agent: a controller that does the following jobs
  - adapts by learning the mapping from actions to service level attributes
  - applies the utility function
  - chooses the action that maximizes utility

User-correction Feedback

- User-correction: allows users to directly correct paper metadata from the web interface – user-based feedback

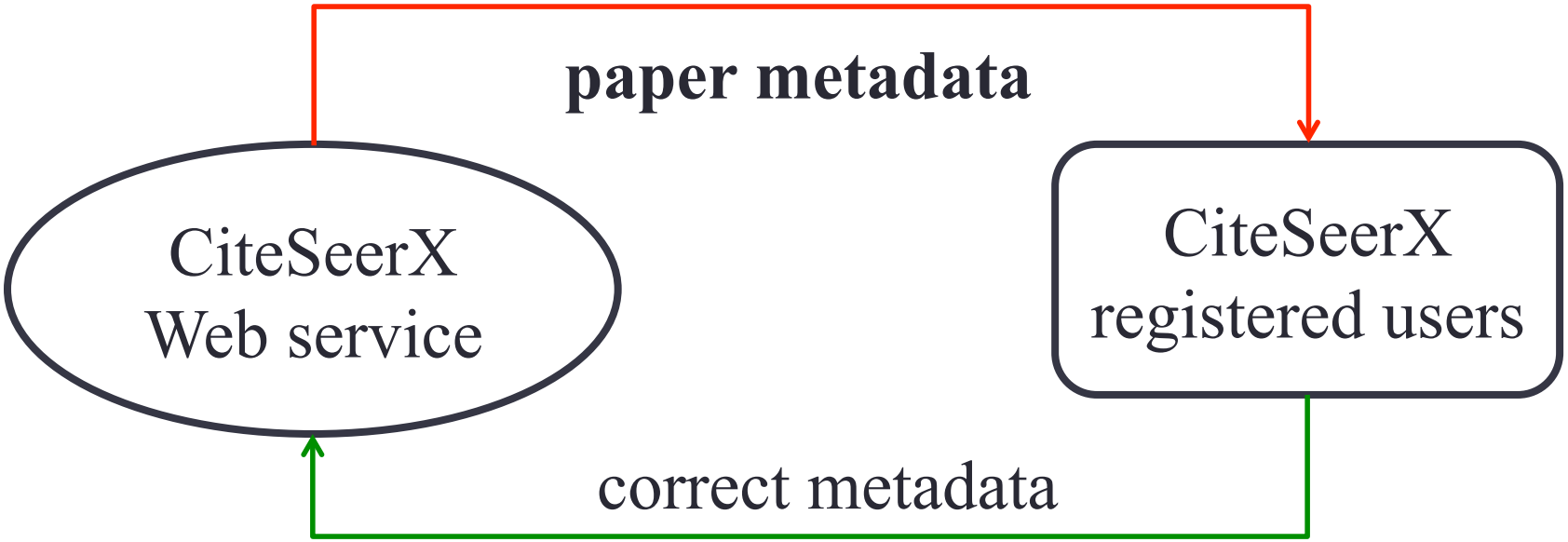


Figure: Feedback diagram for user-correction.

Ill-conditioned Metadata Detection Feedback

- Ill-conditioned Metadata Detection: detects papers with ill-conditioned metadata by checking citation and download history – long-term feedback

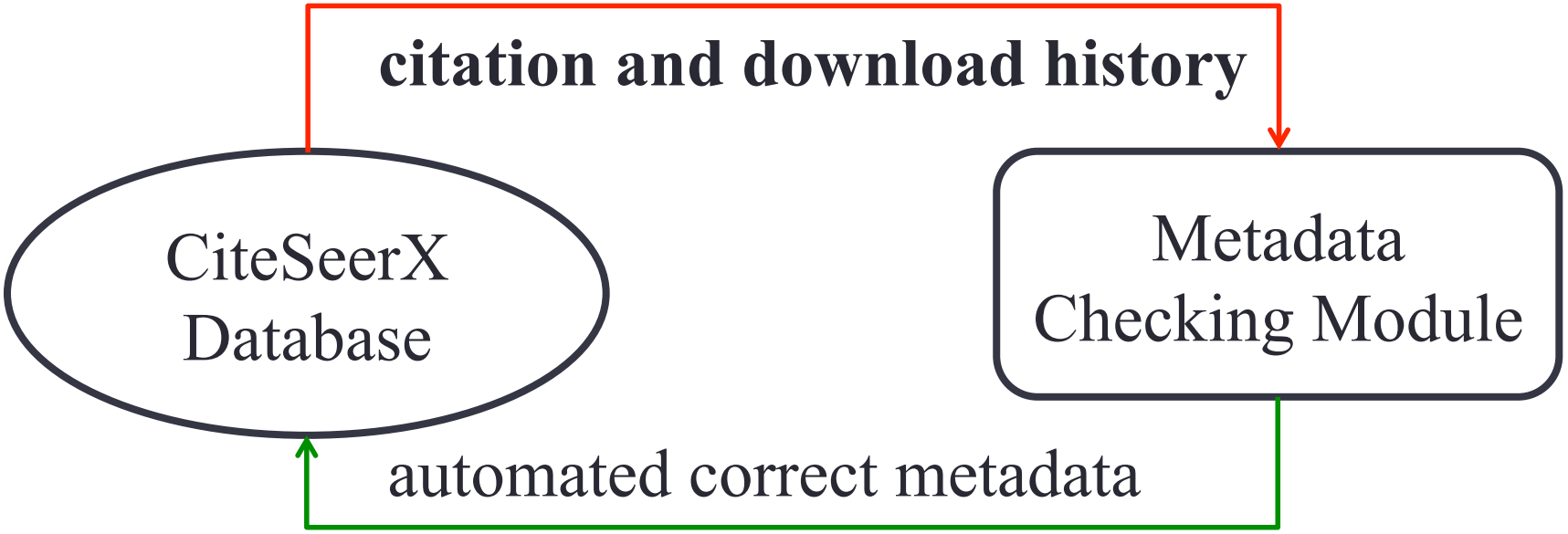


Figure: Feedback diagram for Ill-conditioned metadata detection.

Crawl Whitelist Generation Feedback

- Crawl Whitelist Generation: selects high quality URLs based on the number of scholarly papers – automated feedback

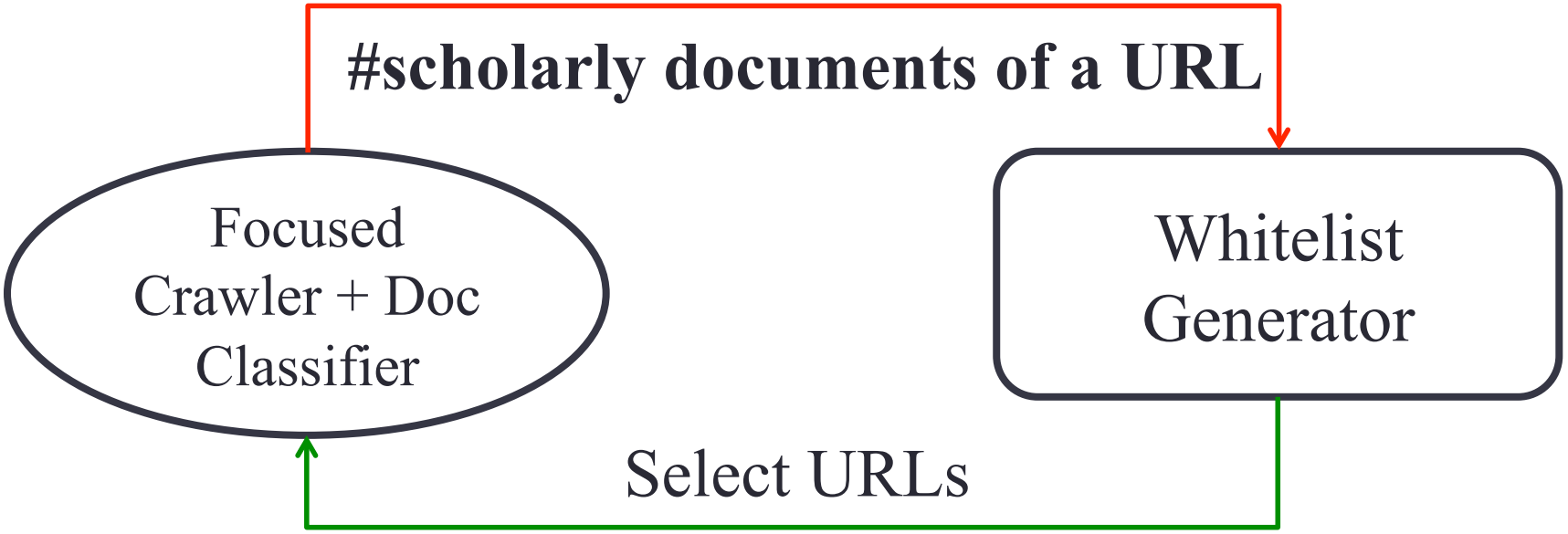


Figure: Feedback diagram for whitelist generation.

Metadata Correction

- Metadata in CiteSeerX
  - Header: titles, authors, affiliations, year, venue, abstract etc.
  - Citations: titles, authors, year, venue, page, volume, issue, citation context etc.
- How does CiteSeerX acquire metadata
  - Actively crawling the Web
  - ***Automatically*** extracting metadata from scholarly documents
    - Header – SVM based extraction tool
    - Citations – CRF-based parsing and tagging tool
    - CiteSeerX extracts acknowledgements, algorithms, figures and tables
  - Challenges: near-duplication (ND), accuracy and efficiency
  - Mistakes occur in metadata extraction
  - Requires correction – user correction and automated correction

User-Correction (uC)

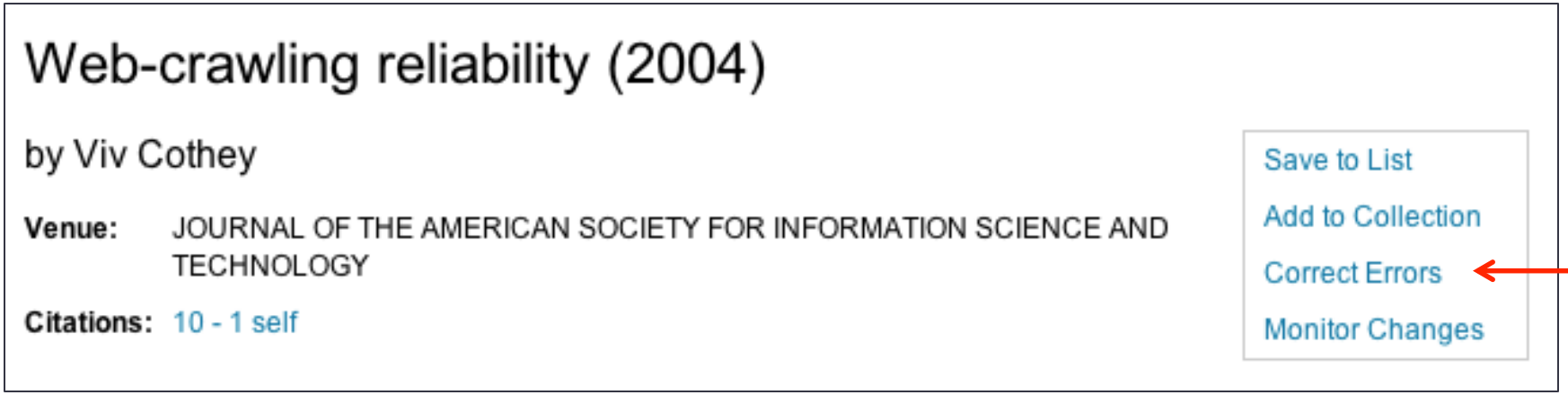
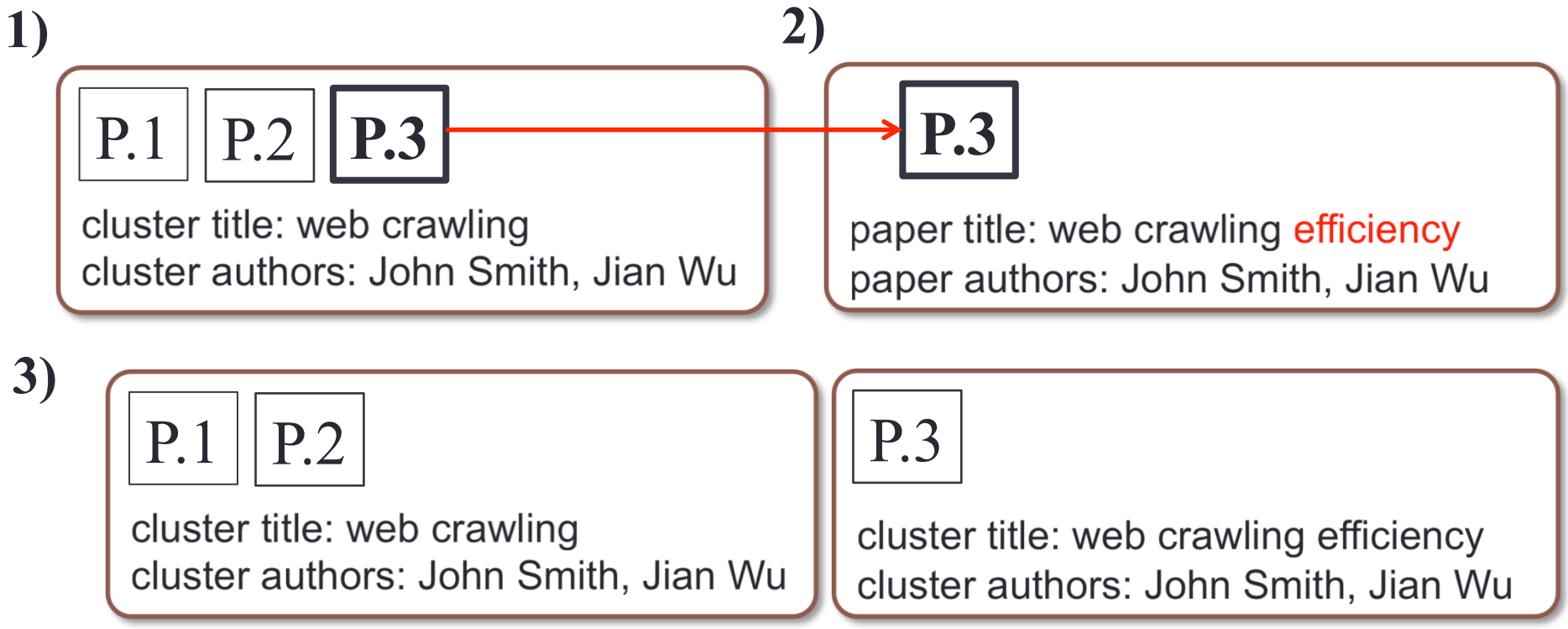


Figure: user-correction Web Interface (WI) on a CiteSeerX paper summary page.

- Features
  - Users must login (limited crowd sourcing)
  - Users can change almost all metadata fields
  - New values are effective immediately after changes are submitted
  - Metadata can be changed multiple times
  - Version control

What happens behind the WI after a uC?



1) P.1, P.2 and P.3 are grouped into the same cluster
2) The title of P.3 is corrected by a user
3) The initial cluster is deleted. Papers are re-clustered.

Evaluate the uC

- CiteSeerX has received more than 277,000 user corrections on 251,000 papers since 2008
- Tens to thousands of uC's daily
- Preliminary evaluation:
  - 50 uC instances
  - Compare metadata before and after corrections
  - Four types of corrections:
    - WC – real corrections, from wrong to correct
    - CW – correct to wrong
    - WW – wrong to wrong: metadata quality not improved
    - WD – a wrong value is deleted

Sample of uC tagging results

metadata fields	WC	WW	CW	WD
title	23	1	0	0
abstract	21	1	2	4
(author) name	19	0	1	0
year	15	0	0	1
(author) affiliation	13	0	0	1
(author) address	7	0	0	1
(author) email	7	0	0	0
venue	5	0	2	0
author**	0	0	0	8
#papers corrected	45	2	5	13

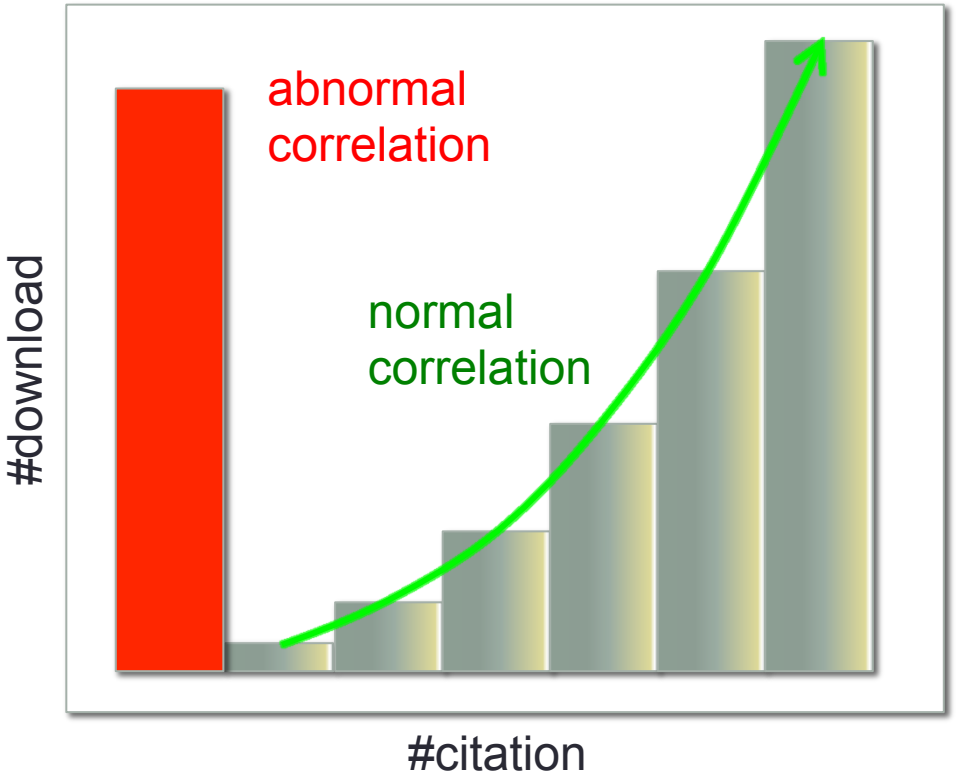
**author: author blocks, including author names, affiliations, addresses and emails.

Implication of uC tagging results

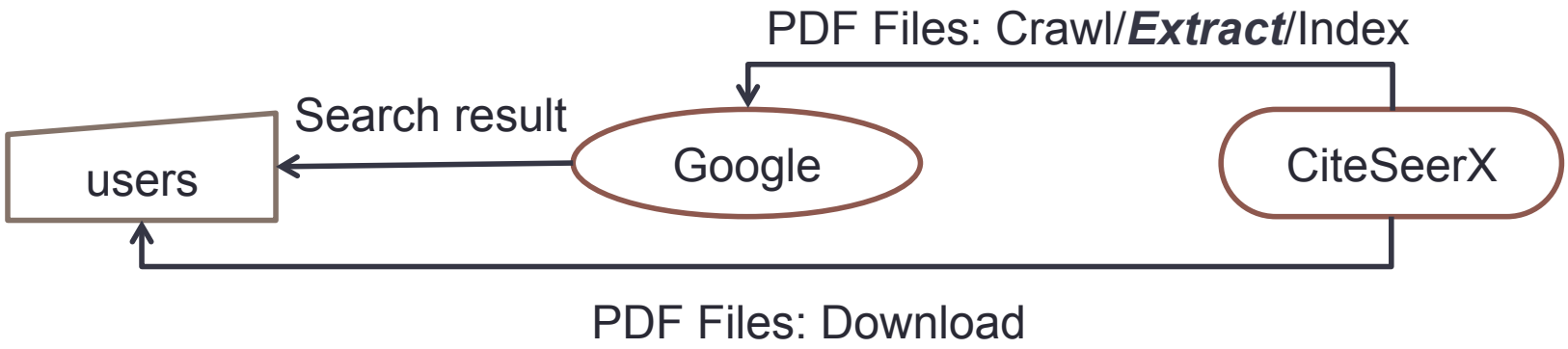
- Most uCs (90%) contain real corrections
- Most corrected fields:
  - titles, abstracts, author names, years, author affiliations
- Fields deleted:
  - Some redundant / wrong author blocks
  - Some abstract
- Most papers are corrected only once
- Estimation
  - About 116,000 paper titles are corrected (3% of all CiteSeerX papes)
  - Author names are corrected in about 100,000 papers
  - These are among the ***most downloaded papers***
- Overall, the uC is a useful method to improve metadata

Ill-Conditioned Metadata Detection

- Access logs over long term can be used to detect anomalies and errors
- Normal correlation – the number of times a given paper is downloaded and the number of citations it has received
- Abnormal documents: large number of downloads with zero citations
- Correct metadata from publisher website or a secondary digital library or use a better extractor



Why are (#download,#citation) used as feedback?



A paper citing my paper
Title: A Review of Feedback Computing in the last decade
…
Reference:
[1] Jian Wu et al. *Utility-based feedback in a digital library search engine*: Cases In CiteSeerX, Feedback Computing 2014

However, the title of my paper was not extracted correctly
Title: We described a utility-based feedback control model and

Crawl Whitelist Generation

- CiteSeerX focused crawler: ***citeseerxbot***
  - Targets: scholarly documents in PDF formats
- Improve crawl efficiency – improve seed quality
- Whitelist vs. blacklist:
  - Blacklist – contains domains and URLs to be filtered out
    - A large fraction of ***non***-scholarly documents – no feedback
  - Whitelist – contains high quality URLs (URLs outside of whitelist domains are not crawled)
    - Example: `http://clgiles.ist.psu.edu/papers/`, whitelist domain: `psu.edu`
    - Use #scholarly documents (n_{paper}) as a feedback
    - URLs with $n_{\text{paper}} > 1$
- How does CiteSeerX select scholarly documents?
  - A rule-based filter – looking for keywords/keyphrases

Evaluate the Feedback Mechanism

- Experimental setup
  - Set P (10 experiments)
    - each experiment: 500 seed URLs randomly selected from 200,000 parent URLs
  - Set W (10 experiments)
    - each experiment: 500 seed URLs randomly selected from the whitelist (generated out of 200,000 parent URLs)
- Crawl efficiency is improved from 22.87% to 44.83% after using n_{paper} as a feedback

Crawl Efficiency Comparison

Experiments	Set P (no feedback)			Set W (with feedback)		
	n_{paper}	n_{PDF}	n_{paper}/n_{PDF}	n_{paper}	n_{PDF}	n_{paper}/n_{PDF}
1	6905	29276	23.58%	698	1308	53.36%
2	2088	8924	12.90%	1152	1735	66.38%
3	3784	16186	23.38%	575	1668	34.47%
4	2438	11141	19.61%	1002	2413	41.52%
5	2740	13974	19.61%	2362	3951	59.78%
6	2259	9395	24.04%	2126	4850	43.84%
7	1845	9873	18.69%	1498	3298	45.42%
8	3089	9432	32.75%	1252	4606	27.18%
9	2079	7486	27.77%	1214	4316	28.13%
10	1998	8284	24.12%	1298	2694	48.18%
Average	-	-	22.87±5.04%	-	-	44.83±12.14%

n_{paper}: # scholarly papers; n_{PDF}: number of total PDF documents crawled

Future Work

- Existing feedback
  - Quantify the significance and importance
- Improve the crawl scheduler by adopting more feedback
  - Utilizes feedbacks from
    - n_{paper}: # of scholarly papers
    - λ: updating rate – how long a scholarly web page is updated (based on the crawl history)
    - How to estimate λ

$$\hat{\lambda} = -\log\left(\frac{\bar{X} + 0.5}{n + 0.5}\right), \bar{X} = n - X$$

      - X – the number of detected changes
      - n – the number of accesses within a time period of T
- Dynamical Topic-Driven Crawler
  - Train a artificial neural network (ANN) model based on a labeled sample
  - Automatically classify URLs on-the-fly
  - Dynamical crawl navigator

Summary

- Three utility-based control feedback paradigms for a digital library search engine
- The user-based feedback allows registered users to perform online changes to metadata. In more than 90% of cases, the users provides correct changes.
- The download and citation history are long-term feedback to detect ill-conditioned metadata and select those papers for corrections
- The # of scholarly documents of URLs (n_{paper}) is used as feedback to generate a URL whitelist. The crawl efficiency is boosted by at least 20%.