Search Engine Driven Author Disambiguation

Yee Fan Tan and Min-Yen Kan
Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543
{tanyeefa, kanmy}@comp.nus.edu.sg

Dongwon Lee
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802
dongwon@psu.edu

ABSTRACT

In scholarly digital libraries, author disambiguation is an important task that attributes a scholarly work with specific authors. This is critical when individuals share the same name. We present an approach to this task that analyzes the results of automatically-crafted web searches. A key observation is that pages from rare web sites are stronger source of evidence than pages from common web sites, which we model as Inverse Host Frequency (IHF). Our system is able to achieve an average accuracy of 0.836.

Categories and Subject Descriptors: H.3.3 **Information Systems** – Information Search and Retrieval

General Terms: Algorithms

Keywords: Entity Resolution, Author Disambiguation, IHF

1. INTRODUCTION

Bibliographic digital libraries such as DBLP [4] and CiteSeer [1] contain a large number of publication metadata records and make these records searchable for academics. A common use of such repositories is to assess the impact of individual researchers on the community. A problem occurs when different individuals share the same name. This leads to mismatch problems in which citations to different authors may be mixed together in a single list (*e.g.*, W. Wang). Such problems can hinder scientific data gathering, information retrieval and even credit attribution [2].

Previous works have focused on using knowledge encoded in the citation records to form appropriate author clusters. For example, Lee *et al.* [3] considered similarity between citations and authors as well as performed blocking on coauthor information, and Han *et al.* [2] used spectral clustering on various fields of the citation record. However, the citation records themselves often contain hidden information that may be difficult to extract. For example, two citations on the same topic may use disjoint keywords in their titles.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
JCDL'06, June 11–15, 2006, Chapel Hill, North Carolina, USA.
Copyright 2006 ACM 1-59593-354-9/06/0006 ...$5.00.

A key differentiating factor in our work is that we leverage resources external to the citation data to resolve this problem. In particular, we leverage the collective information on the web to do disambiguation, by employing a web search engine. Specifically, we attribute a citation to a particular author based on the pages returned by a search engine in response to web queries.

2. ALGORITHM

We deal with a restricted version of the author disambiguation problem. Given an author string name X (representing a known k unique individuals) and a list of citations C containing the name X, our system identifies which citations are attributed to which of the k authors. This problem formulation is identical to the *mixed citation problem* in [3], and can be seen as a standard k-way classification problem.

When lay people are faced with the author disambiguation task and are given unfamiliar publications, they may query a search engine with the publication titles and use the results to help them distinguish between the different authors. Our method tries to approximate this process.

Our proposed algorithm is as follows: For each citation $c \in C$, we query a search engine using the title of c as a phrase search to obtain a set of relevant URLs. Each citation $c \in C$ is then represented by a feature vector, whose features are the relevant URLs and weighted by their IHFs (explained below). Next, we compute the pairwise similarity of two citations $c_1, c_2 \in C$ using cosine similarity. Finally, we perform hierarchical agglomerative clustering (HAC) on C using the similarity values to derive k clusters. The final clusters represent the k individual authors.

IHF Weighting. URLs returned by a search engine are not equally useful, as some may belong to aggregator services. To overcome this problem, we desire a weighting scheme that weighs aggregator web sites with low values and personal and group publication web pages with high values. Akin to the Inverse Document Frequency measure used in information retrieval, we formulate an Inverse Host Frequency (IHF) to gauge the relative rarity of an Internet host among a suitable corpus of web documents. Hosts that correspond to aggregator services will have a high frequency among searches on publication titles, thus a low IHF.

We first form a corpus to establish IHF values. We obtain the citations belonging to the top 100 author strings (by number of citations) in DBLP, and then query a search engine with each citation's title. The returned URLs are truncated to its hostname. If a hostname h has frequency $f(h)$, then its *inverse host frequency* is computed as:

$$\mathrm{IHF}(h) = \log_2 \frac{\max_h f(h) + 1}{f(h) + 1} + 1$$

In our implementation we notice that using the hostname alone may have problems, especially when the host has multiple names or is represented by an IP address (*e.g.*, www.informatik.uni-trier.de, ftp.informatik.uni-trier.de and 136.199.54.185, all the same host) with dissimilar distributions. To correct for these anomalies, we also tried using the domain instead (*e.g.*, uni-trier.de) and resolving all hostnames to IP addresses before processing.

3. EVALUATION

To evaluate our approach we use a manually-disambiguated dataset of computer science citations with 24 ambiguous names, as used in [3]. These names represented 2 unique authors ($k = 2$) in all but one case where it represented 3. Each name is attributed to 30 citations on average, and the proportion of the largest class ranges from 50% to 97%. We used Google (http://www.google.com/) as our search engine, and we attempt to retrieve 10 URLs per citation.

We measure performance using classification accuracy. Suppose the classes are labelled from 1 to N. Let a_i and p_i be the actual and predicted classes of the ith citation respectively. Let $s(x, y)$ be 1 if $x = y$, 0 otherwise. Then,

$$\text{accuracy} = \max_{\pi \in S_N} \sum_i s(\pi(p_i), a_i),$$

where S_N is the set of permutations on the classes 1 to N.

We investigated using different clustering schemes with the IHF data. We tested three HAC schemes: single link, complete link and groupwise average. The accuracies averaged over all names are summarized in Table 1.

	Hostname	Domain	IP address
Single link	0.827	0.807	**0.836**
Complete link	0.726	0.798	0.734
Group average	0.805	0.811	0.812

Table 1: Average accuracy over all author names.

From Table 1, we see that single link always performs as well as or better than complete link and groupwise average. One of the main reasons could be that a publication page of an author sometimes omits some of his or her publications. For example, if a publication page contains only citations c_1 and c_2, and another contains only c_2 and c_3, then single link is best suited for merging all of c_1, c_2 and c_3 into the same cluster. As highlighted, resolving all hostnames to IP addresses gives the best accuracy. We believe that single link may perform better when authors have disparate areas of research, and are not well represented by a centroid vector.

4. DISCUSSION AND CONCLUSION

An investigation of per-name accuracy is shown in Figure 1 using the single link HAC scheme. First, we see an apparent correlation between the accuracies and the average number of URLs returned per citation, which is shown in Figure 2. Those author names with few URLs returned per citation tend to fare poorly, because those results will mostly be aggregator web pages and serves little for the disambiguation task. Second, we do not observe any apparent

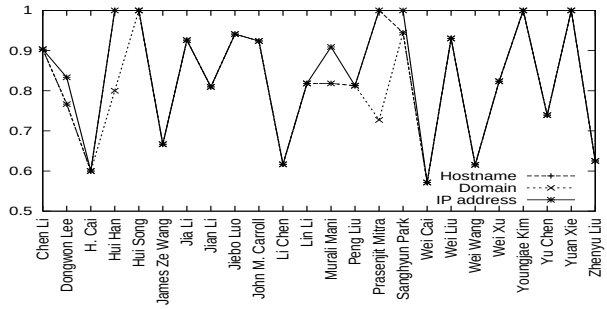

Figure 1: Per-name accuracies using single link.

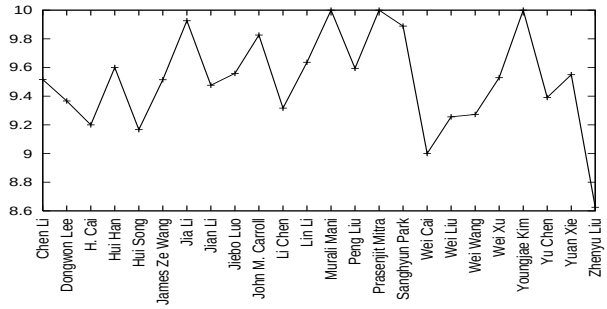

Figure 2: Per-name average number of URLs returned per citation.

relation between the accuracies and number of citations for a given author name. We take this to mean that our algorithm will scale, even when the number of citations is large, provided that there is enough evidence from the URLs returned. As for time efficiency, the analysis of the returned URLs is very fast, and the execution time is dominated by the search engine querying. In scholarly digital libraries, such querying may already be done during spidering, making our approach particularly time-efficient.

We have focused on using the URLs returned from searching the citation titles, and obtained a respectable average accuracy of 0.836 using IP addresses with single link HAC clustering. However, we can also explore other sources of information, such as the publication venues of the citations as well as utilizing the actual contents of the web pages. Lastly, our algorithm is complementary to algorithms that uses internal knowledge, and we plan to combine knowledge gained externally and internally to obtain improved performance.

5. REFERENCES

[1] C. L. Giles, K. D. Bollacker, and S. Lawrence. CiteSeer: An automatic citation indexing system. In *ACM Conf. on Digital Libraries*, 1998.

[2] H. Han, H. Zha, and C. L. Giles. Name disambiguation in author citations using a K-way spectral clustering method. In *JCDL*, 2005.

[3] D. Lee, B.-W. On, J. Kang, and S. Park. Effective and scalable solutions for mixed and split citation problems in digital libraries. In *IQIS*, 2005.

[4] M. Ley. The DBLP computer science bibliography: Evolution, research issues, perspectives. In *SPIRE*, 2002.