Two Supervised Learning Approaches for Name Disambiguation in Author Citations

Hui Han
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA, 16802
hhan@cse.psu.edu

Lee Giles
School of Information Sciences and Technology
The Pennsylvania State University
University Park, PA, 16802
giles@ist.psu.edu

Hongyuan Zha
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA, 16802
zha@cse.psu.edu

Cheng Li
Department of Biostatistics
Harvard School of Public Health
Boston, MA, 02115
cli@hsph.harvard.edu

Kostas Tsioutsiouliklis
NEC Laboratories America, Inc.
4 Independence Way,
Princeton, NJ 08540
kt@nec-labs.com

ABSTRACT

Due to name abbreviations, identical names, name misspellings, and pseudonyms in publications or bibliographies (citations), an author may have multiple names and multiple authors may share the same name. Such name ambiguity affects the performance of document retrieval, web search, database integration, and may cause improper attribution to authors. This paper investigates two supervised learning approaches to disambiguate authors in the citations [1]. One approach uses the naive Bayes probability model, a generative model; the other uses Support Vector Machines(SVMs) [39] and the vector space representation of citations, a discriminative model. Both approaches utilize three types of citation attributes: co-author names, the title of the paper , and the title of the journal or proceeding. We illustrate these two approaches on two types of data, one collected from the web, mainly publication lists from homepages, the other collected from the DBLP citation databases.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval

General Terms

Algorithms

[1]"Citations" refer to an author's publication list in the citation format.

JCDL'04, June 7–11, 2004, Tucson, Arizona, USA.

Keywords

Naive Bayes, Name Disambiguation, Support Vector Machine

1. INTRODUCTION

Due to name variation, identical names, name misspellings, and pseudonyms, we observe two types of name ambiguities in research papers or bibliographies (citations). The first type is that an author has multiple name labels. For example, the author "David S. Johnson" may appear in multiple publications under different name abbreviations such as "David Johnson", "D. Johnson", or "D. S. Johnson", or a misspelled name such as "Davad Johnson". The second type is that multiple authors may share the same name label. For example, "D. Johnson" may refer to "David B. Johnson" from Rice University, "David S. Johnson" from AT&T research lab, or "David E. Johnson" from Utah University (assuming the authors still have these affiliations).

Name ambiguity can affect the quality of scientific data gathering, can decrease the performance of information retrieval and web search, and can cause the incorrect identification of and credit attribution to authors. For example, identical names cause the ambiguity of the "author page" in the web DBLP (Digital Bibliography & Library Project) [2]. The author page of "Yu Chen" in the DBLP contains citations from three different people with the same name: Yu Chen from University of California, Los Angeles; Yu Chen from Microsoft Beijing; Yu Chen as the senior professor from Renmin University of China. Such name ambiguity causes the incorrect identification of authors. For example, the author page of "Jia Li" in the DBLP refers to the "Jia Li" from the Department of Statistics at the Pennsylvania State University. However, the "Home Page" link in her author page directs to the professor with the identical name in the Department of Mathematical Sciences at the University of Alabama in Huntsville. We observe from CiteSeer [18] the incorrect attribution to the authors due to similar ambiguity. "D. Johnson" is the most cited author in Computer Science accord-

[2]http://WWW.Informatik.Uni-Trier.DE/∼ley/db/index.html

ing to CiteSeer's statistics in May 2003 (http://citeseer.nj.nec.com/mostcited.html). However, the citation number that "D. Johnson" obtained in CiteSeer's statistics is actually the sum of several different authors such as "David B. Johnson", "David S. Johnson", and even "Joel T. Johnson".

This paper investigates the name disambiguation in the context of citations. We propose the idea of a canonical name, i.e. a name that is the minimal invariant and complete name entity for disambiguation. Such a name may have more than just the name of the individual as constituents. A possible example of a canonical name would be a name entity that has all the characteristics of a name including abbreviations and AKA's. "Authorized name" is a similar concept in library practice. Getty's ULAN (Union List of Artist's Names) [1] and the Library of Congress name authority file [2] are good demonstrations of the canonical, or authoritative form of names.

Name ambiguity is a special case of the general problem of *identity uncertainty*, where objects are not labeled with unique identifiers [30]. Much research has been done to address the identity uncertainty problem in different fields using different methods, such as record linkage [17], duplicate record detection and elimination [8, 26, 29], merge/purge [22], data association [6], database hardening [11], citation matching [28], name matching [7, 37, 9], and name authority work in library cataloging practice [40, 15, 19].

Citation matching, name matching and name authority work are the work most similar to ours. Citation matching and name matching are similar to our method in citation context and the choice of citation attributes for computation. However, the records identified by related work are actually duplicate records in different syntactic formats, while what we identify are different records authored by the same name entity. Another difference is that many name matching algorithms are string-based [7, 37, 9]. Bilenko et al. show in their work that the string-based similarity computation works better than token-based methods, which may be due to many misspellings in their datasets [7]. We expect token-based methods to better fit the name disambiguation task because the problem can be treated at the token level, and misspellings and abbreviations are not the main source of the citation differences for these cases. Name authority is the process through which librarians for the past centuary have intellectually provided disambiguation for personal and corporate names in the world's bibliographic output. However, much name authority work is conducted manually. DiLauro et. al. [40, 15] propose a semi-automatic algorithm using Bayes probabilities to disambiguate composers and artists in the Levy music collection. However, their algorithm largely depends on the Library of Congress name authority file.

We study two machine learning approaches for name disambiguation, one based on a generative model and the other based on a discriminative model. A generative model can create other examples of the data, usually provides good insight into the nature of the data and facilitates easy incorporation of domain knowledge [4]. We observe that an author's citations usually contain the information of the author's research area and his or her individual patterns of coauthoring. Therefore, we propose a naive Bayes model, a generative statistical model frequently used in word sense disambiguation tasks [16, 38], to capture all authors' writing patterns. Discriminative models such as Support Vector Machines, are basically classifiers. Other differences are that the naive Bayes model uses only positive training citations to model an author's writing patterns, while the SVMs learn from both positive and negative training citations the distinction between different authors' citations. Also, the naive Bayes model classifies a citation to an author based on the probabilities, while the SVMs uses a distance measure [36]. In addition, a probability model allows us to systematically combine different models [23], and is easily extensible to more information; the vector space representation of citations in classification approaches usually needs to tune weights for different attributes [7, 37].

Our approaches assume the existence of a citation database (training data) indexed by the canonical name entities. Such a citation database can be constructed in several ways. For example, constructing the database based on existing databases such as DBLP; collecting publication lists from researchers' home pages (Usually these publication lists are in the citation format); or clustering citations according to the name entities, as shown by our previous work [21].

Given a full citation with the query name implicitly omitted, our name disambiguation is to predict the most likely canonical name from the citation database. For example,"[J. Anderson], S. Baruah, K. Jeffay. Parallel Switching in Connection-Oriented Networks. IEEE Real-Time Systems Symposium 1999: 200-209" is a test citation. "J. Anderson" is the omitted query name. The naive Bayes approach estimates the author-specific probabilities, such as the prior probability of each author, and his/her probabilities of coauthoring with coauthors, using certain keywords in the title of the paper, and publishing papers in certain places, as described in detail in Section 2. Given a new citation and its query author name, name disambiguation is to search the database and choose the canonical name entry with the highest posterior probability of producing this citation. The SVM approach considers each author as a class, and classifies a new citation to the closest author class. With the SVM approach, we represent each citation in a vector space; each coauthor name and keyword in paper/journal title is a feature of the vector.

Both approaches use three attributes of the citations associated with each canonical name entry in the citation database: coauthor names, paper titles, and journal titles. By "journal titles", we actually refer to the titles of all the publication sources, such as proceedings and journals. Author names in citations are represented by the first name initial and last name, the minimal name information seen in citations. Citation attributes can be extracted by methods such as regular expression matching, rule-based system [10], hidden Markov models [33, 34, 35], or Support Vector Machines [20]. To minimize the effect of inaccurate citation parsing on the study of two approaches, we use regular expression matching and manual correction to parse the citations in "J Anderson" and "J Smith" datasets, as discussed in Section 4.1. The DBLP citation datasets are already in the XML format with parsed attributes.

The rest of the paper is organized as follows: Section 2 describes the naive Bayes approach; Section 3 describes the SVM approach; Section 4 reports experiments and results; Section 5 concludes and discusses future work.

2. THE NAIVE BAYES MODEL

We assume that each author's citation data is generated by the naive Bayes model, and use his/her past citations as the training data to estimate the model parameters. Based on the parameter estimates, we use the Bayes rule to calculate the probability that each name entry $X_i (i \in [1, N]$, where N is the total number of candidate name entries in the citation database) would have generated the input citation.

2.1 Model Overview

Given an input test citation C with the omission of the query author, the target function is to find a name entry X_i in the citation database with the maximal posterior probability of producing the

citation C, i.e.,

$$max_i P(X_i|C) \quad (1)$$

Using the Bayes rule, the problem becomes finding

$$max_i P(C|X_i)P(X_i)/P(C) \quad (2)$$

where $P(X_i)$ denotes the prior probability of X_i authoring papers, and is estimated from the training data as the proportion of the papers of X_i among all the citations. The prior is useful to incorporate the knowledge, such that a prolific author can have large $P(X_i)$. $P(C)$ denotes the probability of the citation C and is omitted since it does not depend on X_i. Then Function 2 becomes

$$max_i P(C|X_i)P(X_i) \quad (3)$$

We assume that coauthors, paper titles, and journal titles are independent citation attributes, and different elements in an attribute type are also independent from each other. The different attribute element here refers to the individual coauthor, the individual keyword in the paper title, and the individual keyword in the journal title. By "keyword", we mean the remaining words after filtering out the stop words (such as, "a", "the" "of", etc.). Therefore, we decompose $P(C|X_i)$ in Function 3 as

$$P(C|X_i) = \prod_j P(A_j|X_i) = \prod_j \prod_k P(A_{jk}|X_i) \quad (4)$$

where A_j denotes the different type of attribute; that is, A_1 - the coauthor names; A_2 - the paper title; A_3 - the journal title. Each attribute is decomposed into independent elements represented by A_{jk} ($k \in [0..K(j)]$). $K(j)$ is the total number of elements in attribute A_j. For example, $A_1 = (A_{11}, A_{12}, ..., A_{1k}, ..., A_{1K(1)})$, where A_{1k} indicates the kth coauthor in C. To avoid underflow, we store log probabilities in our implementation, and the target function becomes:

$$max_i P(X_i|C) = max_i[\sum_j \sum_k log(P(A_{jk})) + log(P(X_i))] \quad (5)$$

where $j \in [1, 3]$ and $k \in [0, K(j))$. The above attribute independence assumption may not hold for real-world data, since there exist cases such as multiple coauthors always appearing together. However, empirical evidence shows that naive Bayes often performs well in spite of such violation. Friedman, Domingos and Pazzani show that the violation of the word independence assumption sometimes may affect slightly the classification accuracy (Friedman 1997; Domingos and Pazzani 1996).

2.2 Model Parameters and Estimation

Next we describe the decomposition and estimation of the coauthor conditional probability $P(A_1|X_i)$ from the training citations, where $A_1 = (A_{11}, A_{12}, ..., A_{1k}, ..., A_{1K(1)})$. The probability estimation is the maximum likelihood estimation for parameters of multinomial distributions. The pseudo count 1 is added in parameter estimation to avoid zero probability in the estimation results.

$P(A_1|X_i)$ is decomposed into the following conditional probabilities.

- $P(N|X_i)$ - the probability of X_i writing a future paper alone conditioned on the event of X_i, estimated as the proportion of the papers that X_i authors alone among all the papers of X_i. (N stands for "No coauthor", and "Co" below stands for "Has coauthor").
- $P(Co|X_i)$ - the probability of X_i writing a future paper with coauthors conditioned on the event of X_i. $P(Co|X_i) = 1 - P(N|X_i)$.
- $P(Seen|Co, X_i)$ - the probability of X_i writing a future paper with previously seen coauthors conditioned on the event that X_i writes a future paper with coauthors. We regard the coauthors coauthoring a paper with X_i at least twice in the training citations as the **"seen coauthors"**; the other coauthors coauthoring a paper with X_i only once in the training citations is considered as the **"unseen coauthors"**. Therefore, we estimate $P(Seen|Co, X_i)$ as the proportion of the number of times that X_i coauthors with "seen coauthors" among the total number of times that X_i coauthors with any coauthor. Note that if X_i has n coauthors in a training citation C, we count that X_i coauthors n times in citation C.
- $P(Unseen|Co, X_i)$ - the probability of X_i writing a future paper with "unseen coauthors" conditioned on the event that X_i writes a paper with coauthors. $P(Unseen|Co, X_i) = 1 - P(Seen|Co, X_i)$
- $P(A_{1k}|Seen, Co, X_i)$ - the probability of X_i writing a future paper with a particular coauthor A_{1k} conditioned on the event that X_i writes a paper with previously seen coauthors. We estimate it as the proportion of the number of times that X_i coauthors with A_{1k} among the total number of times X_i coauthors with any coauthor.
- $P(A_{1k}|Unseen, Co, X_i)$ - the probability of X_i writing a future paper with a particular coauthor A_{1k} conditioned on the event that X_i writes a paper with unseen coauthors. Considering all the names in the training citations as the population and assuming that X_i has equal probability to coauthor with an unseen author, we estimate $P(A_{1k}|Unseen, Co, X_i)$ as 1 divided by the total number of author (or coauthor) names in the training citations minus the number of coauthors of X_i. However, the small citation size may underestimate the population of new coauthors that X_i will coauthor with in the real-world. This may in turn underestimates the probability of an author coauthoring with previously seen coauthors. In this case we can set a larger population size.
- $P(A_1|X_i) = P(N|X_i)$ if $K(1) = 0$
- $P(A_1|X_i) = P(A_{11}|X_i)...P(A_{1k}|X_i)...P(A_{1K}|X_i)$ if $K(1) > 0$, where
  $P(A_{1k}|X_i) = P(A_{1k}, N|X_i) + P(A_{1k}, Co|X_i)$
  $= 0 + P(A_{1k}, Co|X_i)$
  $= P(A_{1k}, Seen, Co|X_i) + P(A_{1k}, Unseen, Co|X_i)$
  $= P(A_{1k}|Seen, Co, X_i) * P(Seen|Co, X_i) * P(Co|X_i) +$
  $P(A_{1k}|Unseen, Co, X_i) * P(Unseen|Co, X_i) * P(Co|X_i)$

The above decomposition is motivated by the following hypotheses: (1) Different authors X_i have different probabilities of writing papers alone, writing papers with previously seen coauthors or previously unseen coauthors. (2) Each author X_i has his/her own list of previously seen coauthors, and a unique probability distribution on these previously seen coauthors to write papers with. If the above hypotheses hold, we expect these conditional probabilities to capture the coauthoring history and pattern of X_i, and to help disambiguate the omitted author from the rest of a citation C. Similarly, we can estimate the conditional probability $P(A_2|X_i)$ that an author writes a paper title, and the conditional probability $P(A_3|X_i)$ that he publishes in a particular journal. Taking each title word of the paper and journal as an independent element, we estimate the probabilities that X_i uses a certain word for a future paper title, and publishes a future paper in a journal with a particular word in the journal title. Here the goal is to use author-specific

probabilities to capture information such as the research field, keywords in the research direction, and the preference of title word usage from past citations of X_i.

2.3 Computational Complexity

Suppose a citation database consists of N canonical authors, where each author has an average of M training citations, and each citation has an average of K attribute elements. The computational complexity for training (estimating the probabilities) the above model is $O(MNK)$; the computational complexity for the query step using coauthor information alone is $O(NK)$ for each query citation. This complexity indicates the scalability of our algorithm to real-world applications.

3. SUPPORT VECTOR MACHINES

This approach considers each author as a class, and trains the classifier for each author class. Given a full citation with the omission of the query name, the goal of name disambiguation is to classify this citation to the closest author class. Each citation is represented by a feature vector, with each coauthor name and keyword in the paper/journal title as a feature and its frequency in the citation as the feature weight. We use the $\|X\|_\infty$ to normalize the weight of features with different ranges of values, which was shown to improve the classification performance [20].

We choose Support Vector Machines [39, 12] as classifiers because of their good generalization performance and ability in handling high dimensional data. All experiments use SVM^{light} [24].

3.1 Support Vector Machine Classification and Feature Selection

The SVM is designed for two class classification problem. Let $\{(\vec{\mathbf{x_1}}, y_1), ..., (\vec{\mathbf{x_N}}, y_N)\}$ be a two-class training dataset, with $\vec{\mathbf{x_i}}$ a training feature vector and their labels $y_i \subset$ (-1, +1). The SVM attempts to find an optimal separating hyperplane to maximally separate two classes of training data. The corresponding decision function is called a classifier. In the case where the training data is linearly separable, computing an SVM for the data corresponds to minimizing $\|\vec{\mathbf{w}}\|$ such that

$$y_i(\vec{\mathbf{w}} \cdot \vec{\mathbf{x}}_i + w_0) - 1 \geq 0, \forall i \quad (6)$$

The linear decision function is

$$f(\vec{\mathbf{x}}) = sgn\{(\vec{\mathbf{w}} \cdot \vec{\mathbf{x}}) + w_0\} = sgn\{\sum_i^n \alpha_i^* y_i(\vec{\mathbf{x}}_i \cdot \vec{\mathbf{x}}) + w_0^*\} \quad (7)$$

If $f(\vec{\mathbf{x}}) > 0$, the data $\vec{\mathbf{x}}$ belongs to class 1; otherwise, $\vec{\mathbf{x}}$ belongs to class 2. The absolute value of $f(\vec{\mathbf{x}})$ indicates the distance of $\vec{\mathbf{x}}$ from the other class. In the final decision function $f(\vec{\mathbf{x}})$, the training samples with non zero coefficients α_i^* lie closest to the hyperplane, and are called support vectors. As Equation 7 shows, $f(\vec{\mathbf{x}})$ is a weighted sum of all features, plus a constant term as the threshold. n is the number of support vectors. Zhang et. al [43] propose to rank the features according to their contribution in separating the differences between two classes. We formalize such a contribution of a feature by Expression 8, where x_{ij} is the weight of the feature j in support vector i. We use such ranking of features to analyze the classification performance by SVMs (Section 4.1).

$$\sum_i^n \alpha_i^* y_i x_{ij} \quad (8)$$

We extend SVMs to multi-class classification using the "One class versus all others" approach, i.e., one class is positive and the remaining classes are negative.

4. EXPERIMENTS

4.1 Datasets and Experiment Design

We apply both approaches on two types of data. The first type of data is publication lists collected from the web, mostly from researchers' homepages. This type of data contains two datasets, one from 15 different "J Anderson"s, shown in Table 1, the other from 11 different "J Smith"s [3]. Both "J Anderson" and "J Smith" are ambiguous names in the database of our EbizSearch system - a CiteSeer like search engine specializing in the E-Business area [32]. We query "Google" using name information such as "J Anderson", or the full name information available in our EbizSearch databases such as "James Anderson", and the keyword "publications". We manually check the returned links, recognize each researcher under the same first name initial and last name, and collect their publication web pages to construct our datasets.

The other type of data are downloaded from the DBLP website, which contains more than 300,000 bibliographic XML citation records with parsed citation attributes. We form the three attributes in each citation as a string. We then cluster author names with the same first name initial and the same last name; each name is associated with the citations where the name appears. We sort the formed name datasets by the number of citations in each set. 9 large name datasets with each having more than 10 name variations are chosen for experiments, as shown in Table 7. We observe that many names in the DBLP have complete name information. To avoid tedious manual checking, we choose from each name dataset the full names that have more than five citations, and consider each such name to represent a canonical name entity.

We prepare the training/testing datasets, preprocess the data, and construct the citation databases, in the same way for all datasets. We preprocess the datasets on author names, paper title words and journal title words as follows. All the author names in the citations are simplified to first name initial and last name. For example, "Yong-Jik Kim" is simplified to "Y Kim". A reason for the simplification is that the first name initial and last name format is popular in bibliographic records. Since more name information usually helps name entity disambiguation, we think that insufficient name information from simplified name format would be good for evaluating our algorithm. Moreover, the simplified name format may avoid some cases of name misspellings. We stem the words of paper titles and journal titles using Krovetz's stemmer [25], and remove the stop words such as "a", "the", etc. We also replace the conference or journal title abbreviations by their full names for more information. The full names of the conference or journal titles are obtained from the DBLP websites [4].

Each name dataset is randomly split, with half of them used for training, and the other half for testing. For example, the "J Anderson" dataset contains 117 citations for training and 112 citations for testing; the "J Smith" dataset contains 172 training citations and 166 testing citations. A citation database is then constructed for each name dataset, based on the parsed and pre-processed training citations. For example, the citation database of "J Anderson" contains 15 canonical name entries for 15 different "J Anderson"s, with each name entry associated with available identity information, such as full name, affiliation, research area, as well as authored citations.

With each approach, we conduct 10 experiments with randomly split dataset for each experiment. In each experiment, we explore

[3] http://www.personal.psu.edu/users/h/x/hxh190/projects/name_project.htm

[4] http://www.informatik.uni-trier.de/~ley/db/conf/indexa.html and http://www.informatik.uni-trier.de/~ley/db/journals/index.html

J Anderson	Full name	Affiliation	Research area	Training size	Test size
1	James Nicholas Anderson	UK Edinburgh	Communication interface research	4	4
2	James E. Anderson	Boston College	Economics	7	7
3	James A. Anderson	Brown Univ.	Neural network	2	1
4	James B. Anderson	Penn. State Univ.	Chemistry	3	3
5	James B. Anderson	Univ. of Toronto	Biologist	11	10
6	James B. Anderson	Univ. of Florida	Entomology	9	8
7	James H. Anderson	U. of North Carolina at Chapel Hill	Computer processors	27	27
8	James H. Anderson	Stanford Univ.	Robot	2	2
9	James D. Anderson	Univ. of Toronto	Dentistry	3	2
10	James P. Anderson	N/A	Computer Security	2	1
11	James M. Anderson	N/A	Pathology	3	2
12	James Anderson	UK	Robot vision and philosophy	9	10
13	James W. Anderson	Univ. of KY	Medicine	5	5
14	Jim Anderson	Univ. of Southampton	Mathematician	10	10
15	Jim V. Anderson	Virginia Tech Univ.	Plant pathology	20	20

Table 1: The citation dataset of 15 "J Anderson"s. Column 2, 3 & 4 shows the available "Identification information" of a "J Anderson", e.g., the full name of each "J Anderson", his or her affiliation and research area. "Training/test size" lists the number of citations used for training/testing. For space limitation, we do not list here the web sites where we download the citations.

Scheme	Coauthor		Paper title		Journal title		Hybrid I		Hybrid II
Approach	Bayes	SVM	Bayes	SVM	Bayes	SVM	Bayes	SVM	Bayes
Mean	71.3%	64.4%	77.9%	82.9%	72.1%	74.4%	91.3%	95.6%	93.5%
StdDev	2.1%	3.8%	3.3%	1.9%	2.1%	3.0%	1.6%	1.7%	1.8%
P Value	1.38E-05		0.003		0.012		0.0003		

Table 2: The mean and the standard deviation (StdDev) of the 10 name disambiguation accuracy trials on the "J Anderson" dataset, with both the naive Bayes approach(Bayes) and the SVM approach(SVM); and the statistical significance (two tail P value) of the performance difference by the two approaches.

multiple schemes based on different combinations of the utilized citation attributes. The motivation is to study the contributions of different citation attributes on name disambiguation. Both approaches use three schemes which use alone one citation attribute, and at least one of two "Hybrid" schemes which combine aspects of all three attributes. In the naive Bayes model approach, "Hybrid I" computes the equal joint probability of different attributes. In the SVM approach, "Hybrid I" combines different attributes in the same feature space. The "Hybrid II" scheme is specific to the naive Bayes model and uses the coauthor attribute alone when a coauthor relationship exists between a coauthor in the test citation and a candidate name entry in the citation database; otherwise, "Hybrid II" uses the equal joint probability of all the three attributes. Flexibility of manipulating attributes is an advantage of using the probability model. The absence of a particular attribute can be handled by omitting the corresponding probabilities. "Hybrid II" is motivated by the experimental observation that with the "J Anderson" dataset, adding title words decreases the number of disambiguated names when using only the co-author information. We observe that the coauthor information is valuable for name disambiguation, and design the "Hybrid II" scheme to preserve the names disambiguated by using coauthor information alone.

We evaluate the experiment performance by "accuracy", and define the "accuracy" as the percentage of the query names correctly predicted. The next section shows experiment results and analysis on the all name datasets.

4.2 Name Disambiguation on the First Type of Data

Table 2 shows the mean and the standard deviation (StdDev) of the 10 name disambiguation accuracy trials on the "J Anderson" name dataset, using both approaches. Table 3 shows the similar trials on the "J Smith" name dataset. The experiment results on these two name datasets are similar, most likely due to the two name datasets having similar probability distributions, since most citations in both datasets are derived from labeled homepages. We analyze the experiment results in detail as follows:

(1) Different attributes have different contributions for name disambiguation

Consider the "J Anderson" dataset as an example. Table 2 shows that using paper title words alone achieves higher average accuracy (77.9%, 82.9%) than using either coauthor (71.3%, 64.4%) or journal information alone (72.1%, 74.4%) with both approaches. Table 4 shows in detail one experiment using the naive Bayes approach; all other 9 experiments show similar results. We observe that authors in this dataset have higher probabilities of reusing title words than collaborating with previously seen coauthors. Table 4 shows an example of the probability distribution of each attribute. For example, Row 4 in Column 2&3 (with header "Seen") shows that 92.0% ((86+17) out of 112) test citations reuse the words in paper titles; Row 5 in Column 2&3 shows that 84.8% ((79+16) out of 112) test citations reuse words in journal titles; and Row 3 in Column 2&3 shows that only 57.1% (64 out of 112) test citations have the previously seen coauthor relationship.

The above probability distribution indicates that authors in this dataset tend to use the same words for multiple papers, probably because multiple papers are about the same project. And the authors in some research areas such as Biology or Plant pathology tend to have a few places they prefer to submit papers. For example, J. Anderson 15 (Jim V. Anderson; J. Anderson 15 refers to the 15th table entry) publishes 37.5% (15 out of 40) of his papers in the same journal "Plant physiology". Such consistent information contained in the journal title helps name entity disambiguation more than the paper title words, especially when the name entities to be disambiguated have diverse research areas.

(2) Bayes model better captures the coauthoring patterns of an author than the SVM approach

Table 2 and Table 3 show that the naive Bayes model (71.3%, 75.2% average accuracy) outperforms the SVM approach (64.4%, 60.0% average accuracy) when using coauthor information alone in

Scheme	Coauthor		Paper title		Journal title		Hybrid I		Hybrid II
Approach	Bayes	SVM	Bayes	SVM	Bayes	SVM	Bayes	SVM	Bayes
Mean	75.2%	60.0%	82.3%	84.2%	76.3%	78.4%	92.9%	94.5%	93.0%
StdDev	3.0 %	2.9%	3.5%	1.7%	2.2%	2.3%	2.0%	1.3%	2.1%
P Value	1.2E-09		0.074		0.035		0.031		

Table 3: The mean and the standard deviation (StdDev) of the 10 name disambiguation accuracy trials on the "J Smith" dataset, with both the naive Bayes approach(Bayes) and the SVM approach(SVM); and the statistical significance (two tail P value) of the performance difference by the two approaches.

Scheme(Accuracy)	Seen		Unseen		Alone	
	Correct	Wrong	Correct	Wrong	Correct	Wrong
Coauthor71(63.4%)	64(100%)	0	3(20%)	12(80%)	4(12.1%)	29(87.9%)
Paper title words(76.8%)	86(83.5%)	17(16.5%)	0(%)	9(100%)	N/A	N/A
Journal title words(72.3%)	79(83.2%)	16(16.8%)	2(11.8%)	15(88.2%)	N/A	N/A
Hybrid I(90.2%)	60(93.8%)	4(6.2%)	14(93.3%)	1(6.7%)	27(81.8%)	6(18.2%)
Hybrid II(93.8%)	64(100%)	0(0%)	14(93.3%)	1(6.7%)	27(81.8%)	6(18.2%)

Table 4: The name disambiguation performance on the "J Anderson" dataset, using five schemes of the attributes in the naive Bayes approach. The first column is the scheme used and the associated overall accuracy . The other columns show the distribution (number and relative percentage under each category) of correct and incorrect name disambiguation in three categories "Seen", "Unseen" and "Alone" respectively. For the 4th and 5th row of the table, "Seen" means that the true name entity uses a subgroup of the paper/journal title words in the training data; "Unseen" means otherwise. For the other rows of the table, "Seen" means the existence of a previous coauthorship between the true name entity and at least one given coauthor in the test citation; "Unseen" means no existence of previous coauthorship between the true name entity and any coauthor in the test citation; "Alone" means the query citation has only a single author (the query author).

both datasets. The reason is that the SVM approach is incapable of handling the cases when the test citation contains no coauthor seen in the training set. However, the naive Bayes model reasonably captures the probabilities of an author coauthoring with both previously seen and unseen coauthors. The prior of the author helps to disambiguate the single author of a citation. For example, Table 4 also shows that using coauthor information alone disambiguates correctly 64 (100%) out of 64 query names with coauthors having previously seen coauthorship with the true name, 3 (20%) out of 15 query names with coauthors having no previously seen coauthorship with the true name, and 4 (12.1%) out of 33 query names authoring alone.

(3) "Hybrid II" performs best (93.5% average accuracy) among all five schemes in the naive Bayes approach

The Bayes probability model has the flexibility of manipulating citation attributes, and enables the easy construction of two hybrid schemes. Both "Hybrid I" and "Hybrid II" perform better than using each citation attribute alone in all experiments."Hybrid II" performs best (93.5% average accuracy) among all five schemes. The hybrid schemes perform better than using each citation attribute alone because three attributes together provide additional information. For example, using coauthor information alone, the system has limitations to predict correctly the cases where no given coauthor in the test citation has seen a coauthorship with the given true name, or the query name has no coauthors. Comparing row 6 &7 with row 3 in column 4 & 6 of Table 4 shows that both hybrid schemes predict $34(= (14-3)+(27-4))$ extra citations, i.e., $34/(3+4) = 485.7\%$ extra citations correctly than using coauthor information alone for the cases when no previously seen coauthor relationship exists.

Row 6 in Column 2 of Table 4 shows that "Hybrid I" has less disambiguated names than using coauthor information alone (Row 3 in Column 2), where the previously seen coauthorship exists. This suggests that incorporating the title words information from papers and journals may add noise, and thus decreases the number of correctly disambiguated names, from 64 to 60 in this case. This motivates our "Hybrid II" model, which preserves the disambiguation results obtained by using coauthor information alone when the previously seen coauthorship exists. Table 2 shows that "Hybrid II" improves the name disambiguation accuracy in "J Anderson" dataset from 91.3% to 93.5% on average.

Feature	Ranking(score) in SVM		Probability estimated by Bayes	
	J Smith 2	J Smith 5	J Smith 2	J Smith 5
evaluate	3 (0.18)	846 (-0.01)	0.72%	0.09%
option	11 (0.09)	888 (-0.01)	0.36%	0.09%
research	69 (0.01)	130 (0.03)	0.16%	4.33%

Table 5: The rankings (ranking scores) of three features by SVMs in author class "J Smith 2" and "J Smith 5", and the probabilities "J Smith 2" and "J Smith 5" use these three features , as estimated by the naive Bayes Model.

(4) The SVM approach slightly outperforms naive Bayes approach

Except in the case of using coauthor information alone, the SVM approach slightly outperforms naive Bayes approach in both name datasets. Such better performance from SVM is statistically significant, except in the case of using paper title words alone. One reason may lie in the nature of the two approaches. While the naive Bayes approach models an author's writing patterns only based on the citations of this author, the SVMs look at the citations of all authors and maximize the distinction between an author class and other author classes. Therefore, SVMs can capture and highly rank the features unique to a class, while the naive Bayes model ranks the same features unique or not unique to an author class, assuming an author has the same distribution of these features.

For example, "*[James E. Smith], Kevin F. McCardle. Options in the Real World: Some Lessons Learned in Evaluating Oil and Gas Investments. Operations Research.*" is a test citation of "J Smith 2". The paper title keywords "evaluate" and "option" are unique to "J Smith 2" and are seen in training citations. However, "Hy-

Scheme	Coauthor		Paper title		Journal title		Hybrid I		Hybrid II
Approach	**Bayes**	**SVM**	**Bayes**	**SVM**	**Bayes**	**SVM**	**Bayes**	**SVM**	**Bayes**
S Lee	61.3%	61.7%	14.3%	16.1%	43.8%	41.0%	68.2%	62.2%	**65.4%**
J Lee	70.9%	65.8%	17.7%	18.4%	39.9%	34.8%	67.1%	65.8%	**75.9%**
J Kim	57.1%	54.5%	18.8%	26.8%	40.2%	34.8%	60.7%	63.4%	**66.1%**
Y Chen	78.5%	77.4%	14.0%	16.1%	26.9%	23.7%	74.2%	67.7%	**81.7%**
S Kim	69.0%	60.0%	13.8%	11.5%	27.6%	31.0%	64.4%	57.5%	**70.1%**
C Lee	72.2%	65.3%	13.9%	11.1%	43.1%	40.3%	69.4%	66.7%	**75.0%**
A Gupta	75.0%	71.9%	25.6%	25.6%	50.6%	47.5%	71.9%	68.8%	**78.1%**
J Chen	66.3%	51.8%	31.3%	25.3%	44.6%	47.0%	72.3%	69.9%	**72.3%**
H Kim	73.7%	70.2%	21.1%	29.8%	43.9%	36.8%	73.7%	66.7%	**75.4%**
Mean	**69.3%**	**64.3%**	**18.9%**	**20.1%**	**40.0%**	**37.4%**	**69.1%**	**65.4%**	**73.3%**
StdDev	**6.8%**	**8.3%**	**6.1%**	**6.9%**	**7.9%**	**7.6%**	**4.5%**	**3.8%**	**5.4%**
P Value	**0.010**		**0.497**		**0.053**		**0.009**		

Table 6: The name disambiguation accuracy, mean and standard deviation on 9 DBLP datasets of different names, using multiple schemes of attributes with both the naive Bayes approach(Bayes) and the SVM approach(SVM); and the statistical significance (two tail P value) of the performance difference by the two approaches.

brid I" of the naive Bayes model predicts "J Smith 2" as the second most likely author, and wrongly predicts "J Smith 5" as the most likely author due to his higher prior and higher probability of using the journal title keyword "Research". The "Hybrid I" of the SVM highly ranks features that are unique to "J Smith 2", and correctly classifies this citation to "J Smith 2". Table 5 shows the features ranked by SVM, and the probabilities "J Smith 2" and "J Smith 5" generate these features as estimated by the naive Bayes probability model. For example, the keyword "evaluate" and "option" are respectively ranked as the 3rd and 11th most important feature of "J Smith 2", by the ranking algorithm described in Section 3.1.

4.3 Name Disambiguation on the Second Type of Data

Name set	Name variations	Training size	Test size
S Lee	35	244	217
J Lee	33	172	158
J Kim	25	127	112
Y Chen	24	108	93
S Kim	20	94	87
C Lee	18	80	72
A Gupta	16	172	160
J Chen	13	91	83
H Kim	11	63	57

Table 7: The 9 DBLP datasets of different names and the data size. The column "Name variations" lists the number of name variations each ambiguous name has, e.g. "H Kim" has 11 name variations in the dataset, such as "Hang Joon Kim", "Hae Yong Kim", "Hyogon Kim" , etc. The columns "Training/Test size" list the number of citations in the training/test dataset.

Table 6 shows the performance on the 9 DBLP name sets by both approaches. Because of the different citation data quality and probability distributions these datasets have from the "J Anderson" and "J Smith" datasets, both approaches achieve different performance than on the previous two datasets. We analyze the experiment results in detail as follows.

1. The two approaches achieve worse performance mainly due to the poorer data quality of these DBLP datasets.

- Simplifying each name as first name initial and last name introduces name ambiguity. For example, different names "Sung Jin Kim " and "Seon-Kyu Kim" are simplified to the same name label "S Kim". To investigate this problem, we did another set of experiments on each dataset, and represent the first name by its first three characters. This improves significantly the performance by using coauthor information alone and by using the "Hybrid II" model. Especially, the performance on the "SLee" dataset is improved from 65.4% to 74.2% using "Hybrid II". This indicates that our algorithm can perform better with more available name information.

- Since most authors in the DBLP datasets come from the Computer Science community, they share words such as those of the same word stem "comput". Different researchers are likely to have overlapping research interests, and publish papers in the same research area. The common title/journal keywords shared by different people are in fact "ambiguous" information. This makes the classification harder and may reduce the performance by using the title/journal keywords.

- We approximate the canonical name entities with the full names each having more than five citations. For example, we consider "Hang Joon Kim" as a canonical name entity if he has more than five citations. We do not choose "H Kim" and the citations coauthored by "H Kim" to avoid possible name ambiguity. However, this may miss the citations coauthored by "Hang Joon Kim" but with the name abbreviation "H Kim". Besides, the DBLP does not necessarily contain all the publications of a person. Such incomplete citation information in our DBLP datasets affects the estimation on the probability distribution, and thus the name disambiguation performance.

- The full journal title information we obtain does not cover and replace all the journal title abbreviations in the datasets. This may under-exploit the journal information.

- The DBLP citations under the same name do not always belong to the same name entity, as shown by the examples of the author pages of "Yu Chen" and "Jia Li" in Section 1.

2. The naive Bayes approach significantly outperforms the SVM approach when using coauthor information alone and the two hybrid schemes.

This conforms to the previous observation that the naive Bayes approach well captures the coauthoring patterns of an author. Besides, the test citations in DBLP datasets usually contain more unseen coauthors or keywords. While the SVM feature vector model relies only on the seen features (coauthors or keywords), the naive Bayes probability model captures more information.

Another reason is that many authors in the DBLP datasets have close research areas and share overlapping keywords; the SVM may under-rank such features in an author class. For example

	PCseen	PCunseen	PN	PKunseen	PJunseen
J Anderson	30.0%	42.6%	27.4%	66.6%	53.3%
J Smith	29.7%	45.1%	25.2%	57.5%	46.8%
DBLP	40.3%	46.1%	13.6%	87.2%	53.1%

Table 8: The average conditional citation attribute probability distribution of an author X_i from the "J Anderson" dataset, "J Smith" dataset, and the DBLP datasets. (The probability estimation is shown in Section 2.2). PCunseen: the probability of X_i writing a future paper with previously unseen coauthors; PCseen: the probability of X_i writing a future paper with previously seen coauthors; PN: the probability of X_i writing a future paper alone; PKunseen: the probability of X_i using unseen words for a future paper title; PJunseen: the probability of X_i publishing a future paper in a journal (or proceeding) with different title words from previous journal titles.

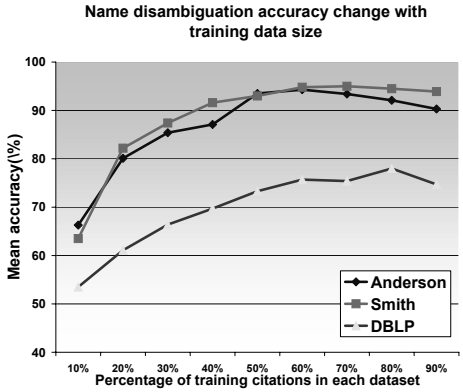

Figure 1: The mean name disambiguation accuracy change with the size of training data using the "Hybrid II" scheme.

"Sukho Lee, Dongseop Kwon, Sangjun Lee. Allocation of Shared Data Based on Mobile User Movement. Mobile Data Management." is a citation of "S Lee 32", and has no seen coauthors. The training citations of "S Lee 32" frequently use keywords related to"Data" and "Base", which are also used by other "S Lee"s. The above test citation contains an unseen keyword "Mobile", which is a highly ranked feature of "S Lee 33" by SVM. This citation is wrongly classified by "Hybrid I of the SVM to "S Lee 33". However, the "Hybrid I" of the probability model considers more factors than the seen features to predict correctly. Besides calculating the probabilities of using the seen keywords "data" and "base", "S Lee 32" has a larger prior, and a higher probability of coauthoring with a new person than "S Lee 33".

3. Using coauthor information alone performs significantly better than using title/journal keywords alone.
The DBLP datasets have different probability distributions for citation attributes. Table 8 shows that authors in the DBLP datasets are more likely (40.3%) to collaborate with previously seen coauthors than those in the "J Anderson" (30.0%) and the "J Smith" (29.7%) datasets. Moreover, authors in the DBLP datasets have higher probabilities to use previously unseen (different) words for a future paper title (87.2%) than those in the "J Anderson" (66.6%) and the "J Smith" (57.5%) datasets. Therefore, we see less contribution from using paper title words. When using paper title words alone, the authors with high priors and high probabilities of using previously unseen words for a future paper dominate the prediction. Journal title words perform significantly better (40.0% accuracy) than paper title words (18.9% accuracy) on average. This indicates that the journal title words are more consistent than paper title words.

4. The "Hybrid II" model performs better (73.3% accuracy on average) than the "Hybrid I" model (69.1% accuracy on average).
This verifies our hypothesis that the "Hybrid II" model preserves the disambiguation results based on the previously seen coauthorship. This also shows that the coauthor information is useful and robust for name disambiguation, because of the consistently good performance on all the 11 datasets.

4.4 Training Dataset Size and the Performance of "Hybrid II"

Generating the labeled training data is the rather expensive price that has to be paid for supervised learning systems. Therefore, we study the performance of our name disambiguation algorithm with the change of training dataset size. We vary the percentage of the datasets for training from 10% to 90% with a step size of 10%, and did a set of experiments. Figure 1 shows the corresponding change of the average name disambiguation accuracies with two types of datasets. We observe from Figure 1 that encouraging results are achieved with only 20% of the datasets for training.

5. CONCLUSIONS

This paper describes two supervised learning approaches to disambiguate name entities in citations. The naive naive Bayes approach determines the author with the highest posterior probability of writing the paper of a citation; the SVM approach classifies a test citation to the closest author class. Both approaches use three types of citation attributes: coauthor names, paper title keywords and journal title keywords, and achieve more than 90.0% accuracies in disambiguating 15 "J Anderson"s and 11 "J Smith"s. "Hybrid" schemes of naive Bayes approach significantly outperforms (73.3% average accuracy) the "Hybrid I" scheme of SVM approach (65.4% average accuracy) in the 9 DBLP name datasets. Coauthor names appear to be the most robust attribute for name disambiguation; using coauthor information alone performs consistently well in all the datasets. Using journal title words usually gives better performance than using paper title words.

Both approaches have advantages. While SVMs highly rank the features specific to an author class, the naive Bayes probability model captures information beyond the seen features, e.g., the unseen coauthors and keywords, and the prior of an author. The Bayes model especially well captures the coauthoring patterns of an author. The flexibility of manipulating different attributes is the advantage of such a probability model. The "Hybrid II" scheme selectively uses predictive features, i.e. seen coauthors, and achieves best results among all five schemes in the naive Bayes approach. To achieve similar effect to "Hybrid II", the feature vector model usually needs to experimentally tune the weights for different attributes based on performance on training or validation datasets. Both approaches allow "non-existence call" if the confidence of the prediction, i.e., the probability in the naive Bayes model, or the distance in the SVM classification, is too low. In this case the algorithms recommend a new name entity to the database.

Further improvements can be obtained, i.e. semantic word clustering on paper titles and journal titles [41]. A researcher usually has a research area or areas that do not change over a period of time, and his/her paper or submitted journal titles are closely related to

his/her research topic. However, the paper and journal title words are sparse, and an author may not reuse a certain group of title words with high probabilities. Therefore, it is reasonable to cluster "similar" title words into research fields and model the probabilities that this researcher uses similar words in the paper title. Such a word cluster reduces feature sparseness, and has more robust probability estimates by averaging statistics for similar words [3]. Existing word clustering methods we can apply include methods based on the Word Net [5], distributional word clustering [3, 31, 13, 14], bipartite word clustering [42], and committee-based word clustering [27]. Research keywords classification schemes such as the ACM classification may also help to map different title words into a research category.

Both approaches can also be applied to the author disambiguation in the context of documents. More attributes can be used, such as the author's affiliation. Words and bigrams from the paper abstracts may also provide useful information for disambiguation. To address real-world problems, we would take wrong spelling and all other author name problems into account, to find the canonical name of an author. We would also like to disambiguate similar corporate names appeared in academic and publishing worlds, for example, the "Loyola" college.

In addition, we see extensions to many types of name disambiguation in digital documents, i.e. potential applications in home page disambiguation. To disambiguate two homepages H1 of Author 1 and H2 of Author 2 with publication lists (in citation formats), we can use the cumulative probability of all citations in the publication list as the probability of the corresponding home page, or we can regard all citations in a home page as meta-citation. Then we use the citations in H1 to train a model for Author 1, and compute the probabilities of Author 1 authoring the citations of H2, and vice versa. If both probabilities are large, then H1 and H2 refer to the same author. We can also train a SVM classifier for Author 1 using the citations in H1, and classify the Author 2, and vice versa. If both classifiers classify the two authors as the same, then H1 and H2 refer to the same author. Future work can use combinations of the above methods.

6. ACKNOWLEDGMENTS

We would like to acknowledge partial support from NSF Grant 0121679, CCF-0305879, and helpful comments from reviewers.

7. REFERENCES

[1] Getty's ULAN (Union List of Artist's Names). http://www.getty.edu/research/conducting_research/vocabularies/ulan/.

[2] The library of congress name authority file. http://www.loc.gov/marc/authority/index.html.

[3] L. D. Baker and A. K. McCallum. Distributional clustering of words for text classification. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval*, pages 96–103, 1998.

[4] A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra. Generative model-based clustering of directional data. In *Proceedings of The 9th ACM SIGKDD Conference on Knowledge Discovery and Data Mining(KDD)*, pages 19–28, 2003.

[5] S. Banerjee and T. Pedersen. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics*.

[6] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

[7] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg. Adaptive name matching in information integration. *IEEE Intelligent Systems*, 18(5):16–23, 2003.

[8] D. Bitton and D. J. DeWitt. Duplicate record elimination in large data files. *ACM Transactions on Database Systems*, 8(2):255–265, 1983.

[9] L. K. Branting. Name-matching algorithms for legal case-management systems. *Journal of Information, Law and Technology (JILT)*, 1, 2002.

[10] M. E. Califf and R. J. Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of the 16th National Conference on Artificial Intelligence*, pages 328–334, 1999.

[11] W. W. Cohen, H. A. Kautz, and D. A. McAllester. Hardening soft information sources. In *Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining*, pages 255–259, 2000.

[12] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.

[13] I. Dagan, F. C. N. Pereira, and L. Lee. Similarity-based estimation of word cooccurrence probabilities. In *Meeting of the Association for Computational Linguistics*, pages 272–278, 1994.

[14] I. Dhillon, S. Manella, and R. Kumar. A divisive information-theoretic feature clustering for text classification. *Journal of Machine Learning Research(JMLR)*, 3:1265–1287, 2003.

[15] T. DiLauro, G. S. Choudhury, M. Patton, J. W. Warner, and E. W. Brown. Automated name authority control and enhanced searching in the levy collection. *D-Lib Magazine*, 7(4), 2001.

[16] G. Escudero, L. arquez, and G. Rigau. Naive bayes and exemplar-based approaches to word sense disambiguation. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI)*, 2000.

[17] I. P. Fellegi and A. B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64:1183–1210, 1969.

[18] C. L. Giles, K. Bollacker, and S. Lawrence. CiteSeer: An automatic citation indexing system. In *Proceedings of the 3rd ACM Conference on Digital Libraries*, pages 89–98, 1998.

[19] P. Gillman. National name authority file: Report to the national council on archives. Technical Report British Library Research and Innovation Report 91, The British Library Board, 1998.

[20] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, and E. A. Fox. Automatic document metadata extraction using support vector machines. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital libraries*, pages 37–48, 2003.

[21] H. Han, H. Zha, and C. L. Giles. A model-based k-means algorithm for name disambiguation. In *Proceedings of the 2nd International Semantic Web Conference (ISWC-03) Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data*, 2003.

[22] M. A. Hernandez and S. J. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Mining and Knowledge Discovery*, 2(1):9–37, 1998.

[23] T. Hofmann. Probabilistic latent semantic analysis. In

Proceedings of Uncertainty in Artificial Intelligence, UAI'99, 1999.

[24] T. Joachims. A statistical learning model of text classification with support vector machines. In *Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval*, pages 128–136, 2001.

[25] R. Krovetz. Viewing Morphology as an Inference Process,. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 191–203, 1993.

[26] M.-L. Lee, T. W. Ling, and W. L. Low. Intelliclean: a knowledge-based intelligent data cleaner. In *In 6th International Conference on Knowledge Discovery and Data Mining*, pages 290–294, 2000.

[27] D. Lin and P. Pantel. Concept discovery from text. In *Proceedings of Conference on Computational Linguistics*, pages 577–583, 2002.

[28] A. McCallum, K. Nigam, and L. H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *Knowledge Discovery and Data Mining*, pages 169–178, 2000.

[29] A. E. Monge and C. Elkan. An efficient domain-independent algorithm for detecting approximately duplicate database records. In *Research Issues on Data Mining and Knowledge Discovery*, pages 23–29, 1997.

[30] H. Pasula, B. Marthi, B. Milch, S. Russell, and I. Shpitser. Identity uncertainty and citation matching. In *Proceedings of Neural Information Processing Systems: Natural and Synthetic*, number 15, 2002.

[31] F. C. N. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In *Meeting of the Association for Computational Linguistics*, pages 183–190, 1993.

[32] Y. Petinot, P. B. Teregowda, H. Han, C. L. Giles, S. Lawrence, A. Rangaswamy, and N. Pal. ebizsearch: An oai-compliant digital library for ebusiness. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital libraries*, pages 199–209, 2003.

[33] K. Seymore, A. McCallum, and R. Rosenfeld. Learning hidden Markov model structure for information extraction. In *Proceedings of AAAI 99 Workshop on Machine Learning for Information Extraction*, 1999.

[34] M. Skounakis, M. Craven, and S. Ray. Hierarchical hidden markov models for information extraction. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003.

[35] A. Takasu. Bibliographic attribute extraction from erroneous references based on a statistical model. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital libraries*, pages 49–60, 2003.

[36] K. Takeuchi and N. Collier. Use of support vector machines in extended named entity, 2002.

[37] S. Tejada, C. Knoblock, and S. Minton. Learning domain-independent string transformation weights for high accuracy object identification. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 350–359, 2002.

[38] Y. Tsuruoka and J. Tsujii. Training a naive bayes classifier via the em algorithm with a class distribution constraint. In *Proceedings of Computational Natural Language Learning(CoNLL)*, pages 127–134, 2003.

[39] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

[40] J. W. Warner and E. W. Brown. Automated name authority control. In *Proceedings of the 1st ACM/IEEE-CS Joint Conference on Digital libraries (JCDL01)*, 2001.

[41] Y. Y. Yao, S. Wong, and L. S. Wang. A non-numeric approach to uncertain reasoning. *International Journal of General Systems*, 23(4):343–359, 1995.

[42] H. Zha, X. He, C. Ding, M. Gu, and H. Simon. Bipartite graph partitioning and data clustering. In *Proceedings of ACM CIKM 2001, the 10th International Conference on Information and Knowledge Management*, pages 25–32, 2001.

[43] X. Zhang and W. H. Wong. Recursive sample classification and gene selection based on svm: method and software description. In *Technical Report, Department of Biostatistics, Harvard School of Public Health*, 2001.