
Computational & Mathematical Organization Theory, 11, 119–139, 2005

A Network Analysis Model for Disambiguation of Names in Lists

BRADLEY MALIN*
Data Privacy Laboratory, Institute for Software Research International, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA;Center for the Computational Analysis of Social and Organizational Systems, Institute for Software Research International, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
email: malin@cs.cmu.edu

EDOARDO AIROLDI
Data Privacy Laboratory, Institute for Software Research International, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
email: eairoldi@cs.cmu.edu

KATHLEEN M. CARLEY
Center for the Computational Analysis of Social and Organizational Systems, Institute for Software Research International, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
email: kathleen.carley@cs.cmu.edu

Abstract

In research and application, social networks are increasingly extracted from relationships inferred by name collocations in text-based documents. Despite the fact that names represent real entities, names are not unique identifiers and it is often unclear when two name observations correspond to the same underlying entity. One confounder stems from ambiguity, in which the same name correctly references multiple entities. Prior name disambiguation methods measured similarity between two names as a function of their respective documents. In this paper, we propose an alternative similarity metric based on the probability of walking from one ambiguous name to another in a random walk of the social network constructed from all documents. We experimentally validate our model on actor-actor relationships derived from the Internet Movie Database. Using a global similarity threshold, we demonstrate random walks achieve a significant increase in disambiguation capability in comparison to prior models.

Keywords: disambiguation, social networks, link analysis, random walks, clustering

1. Introduction

Link analysis is increasingly performed on networks constructed from personal name relationships extracted from text-based documents (e.g. Coffman et al., 2004; Culotta et al.,

*To whom correspondence should be addressed.
This paper is an extension of research presented at the 2005 SIAM Workshop on Link Analysis, Counterterrorism, and Security (Malin 2005).
This research was supported in part by the Data Privacy Laboratory at Carnegie Mellon University and by NSF IGERT grant number 9972762 in CASOS.

2004; Harada et al., 2004; Diesner and Carley, 2005, Thompson, 2005). In such networks, a vertex corresponds to a particular name and an edge specifies the relationship between two names. Before such a network can be analyzed for centrality, grouping, or intelligence gathering purposes, the correctness of the network must be maximized. Specifically, it must be decided when two pieces of data correspond to the same entity or not. Failure to ensure correctness can result in the inability to discover certain relationships or cause the learning of false knowledge.

Names are not unique identifiers for specific entities and, as a result, there exist many confounders to the construction of correct networks. Firstly, the data may consist of typographical error. In this case, the name "John" may be accidentally represented as "Jon" or "Jhon". There exist a number of string comparator metrics (Winkler, 1995; Cohen et al., 2003; Wei, 2004) to account for typographical errors, many of which are in practice by various federal statistical agencies, such as the U.S. Census Bureau. However, even when names are devoid of typographical errors, there are additional confounders to data correctness. For instance, there can exist name variation, where multiple names correctly reference the same entity. Or, more pertinent to our research, there can exist name ambiguity, such that the same name correctly references multiple entities. While both problems must be accounted for, this paper concentrates on the basic aspects, and how to resolve, ambiguity. The basic question we ask is, "How can a computer resolve which particular entity is referred to, or disambiguate, various observations of the same name?"

Disambiguation is by no means a trivial feat, and the manner by which a human makes a decision is often contingent on contextual clues as well as prior background knowledge. For example, when a reader encounters the name "George Bush", the reader must decide if the name represents "George H.W. Bush"—the 41st President of the United States of America, or "George W. Bush"—the 43rd president, or some other individual of lesser notoriety. When the name is situated in a traditional communiqué, such as a news story, humans tend to rely on linguistic and biographical cues. If the name was situated in the following sentence, "George Bush was President of the United States of America in 1989", then, with basic knowledge of American history, it is clear the story refers to the elder "George H.W. Bush".

Though spoken conversations and written communications between entities are structured by known grammars there is no requirement for text-based documents to provide traditional semantic cues. One such counter scenario occurs when documents are merely rosters that consist of nothing but names (Sweeney, 2004). To relate information corresponding to the same entity in this type of environment, disambiguation methods must be able to leverage list-only information. Models developed for natural language processing (Vronis and Ide, 1999), such as those available in the sentence regarding the American President, are not designed to account for this new breed of semantics.

Recently, the data mining community has focused on the design of less structure dependent disambiguation methods (Bhattacharya and Getoor, 2004; Jensen and Neville, 2000; Kalashnikov et al., 2005). These methods are often tailored to assumptions and characteristics of the environments where the references reside. For example, some methods leverage the covariates of references (i.e. the observation of two references in the same source) or require that social groups function as cliques (Bhattacharya and Getoor, 2004). This model

expects environments in which strong correlations exist between pairs or sets of entities, such that they often co-occur in information sources. While closely knit groups of entities provide an ideal scenario, it is not clear if such social settings manifest in the real world. In contrast, it is feasible, and intuitive, to leverage less directly observed relationships. This is precisely the route explored in this paper.

In this paper, we consider networks of the references in question, such that one can leverage "community" structures among entities (Girvan and Newman, 2002). By studying communities of entities, we exploit relationships between entities which have minimal, or no, observed interactions. This is extremely powerful, since it allows for disambiguation when covariates are weak or the social network of entities is less centralized. We investigate the degree to which disambiguation methods can be automated using relational information only. More specifically, given only a set of observations of names from information sources, such as webpages, can we construct an automated system to determine how many entities correspond to each particular name? Furthermore, can we determine which particular name observation corresponds to which underlying entity?

The remainder of this paper is organized as follows. In the following section we review related research in disambiguation models from the natural language processing and data mining research communities. In Section 3, we introduce a formal model of our network analysis and evaluation methods. Next, in Section 4, we report experiments on a dataset consisting of movie-actor lists derived from the Internet Movie Database (IMDB). Findings from this analysis suggest that community similarity, which leverages network similarity are more reliable for disambiguation than document similarity. Then, in Section 5, we discuss limitations of network-based similarity metrics and possible extensions to this research.

2. Background

There exist a number of approaches that have been applied to disambiguation. In this section, we briefly review previous disambiguation research and where the work presented in this paper differs.

In general, disambiguation methods can be taxonomized on two features: (1) information type and (2) supervision. Information type specifies to whom data corresponds and there are two main types often used for disambiguation: (a) personal and (b) relational. Personal information corresponds to static biographical (e.g. George H.W. Bush was the 41st President) and grammatical (e.g. *fall* used as a noun vs. as a verb) information. To leverage this information, disambiguation methods usually use sets of rules for discerning one meaning from another. In contrast, relational information specifies the interactions of multiple values or terms (e.g. George H.W. Bush tends to collocate with Ronald Reagan whereas George W. Bush tends to collocate with *Dick Cheney*).

The second taxonomizing feature is the supervision of the disambiguation process. In supervised learning systems, each disambiguation method is trained on labeled sample data (e.g. first sample corresponds to first meaning, second sample corresponds to second meaning, etc.). In an unsupervised learning system, methods are not trained, but instead attempt to disambiguate based on observed patterns in the data.

2.1. *Personal Disambiguation*

Word sense disambiguation methods initially gained momentum in natural language processing. Early computational methods tagged sentences with parts of speech and disambiguated words/phrases based on the tags (Brill and Resnick, 1994; Jensen and Binot, 1987). With the incorporation of a database-backed model, IBM's "Nominator" system (Wacholder et al., 1997), used phrase context (e.g. punctuation, geographic position in sentence, and capitalization) in parallel with prior knowledge (e.g. known type of entity for names) for disambiguation. Names encountered by the system were matched to names whose context and knowledge were previously specified.

Bagga and Baldwin (1998) introduced an unsupervised disambiguation model based on sentence comparison without prior knowledge. Sentences are parsed into vector-space summaries of words or concepts. Summary pairs are compared and similarity scores above a certain threshold are predicted as the same entity. Mann and Yarowsky (2003) extended summaries to parse for structured biographical data, such as birth day, birth year, occupation, and place of birth. The name observations were then clustered based on similarity of biographies.

The aforementioned methods require prior specification of rules, grammars, and multiple attributes for comparison and, as a result, there is a lack of accountability for unstructured information. An alternative approach for natural language disambiguation is based on probabilistic models of word usage. Lesk (1986) extended rule based models to account for the relationship of an ambiguous word with its surrounding words. He demonstrated that overlap in the dictionary definitions' of surrounding text words can be used to disambiguate. Gale et al. (1992) showed dictionary definitions were unnecessary, provided a representative sample of word covariation was available. They verified this claim in a supervised environment, in which a naïve Bayes classifier was trained for each ambiguous word based on the usage of surrounding words, or covariance. Over the years, additional statistical models for word and concept covariates have been studied (Brown et al., 1991; Ginter et al., 2004; Hatzivassiloglou et al., 2001; Ng 1997; Yarowsky 1992).

2.2. *Relational Disambiguation*

Networks provide a way to construct robust patterns from minimally structured information. Certain word disambiguation methods have employed semantic networks from corpora for more robust similarity measures (Chan and Franklin, 1998, Hiro et al., 1996; Veronis and Ide, 1999). Similarly, other models have considered belief propagation networks and Bayesian models for disambiguation (Chao and Dyer, 2000).

Recent research has investigated link structure and social networks for disambiguation. Bekkerman and McCallum (2005) study disambiguation of names in a linked environment, such as the World Wide Web. Their model leverages hyperlinks and the distance between pages where ambiguous names are found. For our research, we consider an environment in which there is no link structure between documents. In contrast, Bhattacharya and Getoor (2004) investigate a specific case of social networks for disambiguation of names residing in documents representative of co-authorship. In the latter's research, both

ambiguity and variation problems are tackled simultaneously using an iterative approach akin to expectation-maximization. The model measures distance between groups, where a group is a clique of entities representative of the document in which the reference resides, as predicted from a previous iteration. Based on its design, the approach skews predictions towards groups which are not only equivalent, but function as cliques. This model is not necessarily representative of the space of social networks. It is unclear if this model generalizes to other types of networks (Albert and Barabási, 2002; Newman, 2003), such as small-world, hierarchical, or cellular.

Clique detection requires what we informally term *exact* similarity, such that relationships between entities must be directly observed (e.g. *Alice* and *Bob* are related if they collocate in the same source). As applied in this research, we make use of *community* similarity to relax the direct observation requirement and permit relationships to be established between entities indirectly. For instance, *Alice* and *Bob* may never be observed together, but both *Alice* and *Bob* can collocate with *Charlie*, *Dan* and *Fran*. Though community similarity measures do not necessarily account for all types of networks, the goal of this research is to demonstrate their capability in comparison to exact similarity in a controlled environment. We suspect that in a less centralized system, similarity measures based on community provide more robust metrics.

In the following section, we introduce several methods. The first is dependent on exact similarity, while the latter is an alternative method which measures community similarity.

3. Methods

In this section, we introduce terminology, notation, and formally define the disambiguation problem.

An *entity* is the basic element of the population of interest, e.g., a person. However, in our problem, entities are not observed, thus we introduce the set of entities in our model by means of a set of latent variables $H = \{h_1, h_2, \ldots, h_k\}$. There is a latent variable for each entity in the model. Our methodology will estimate the number of underlying entities in a given dataset, hence we do not have to specify how many entities are in the model a priori. An observation corresponds to a set of measurable characteristics of an entity, e.g., a last name and initials of first and middle name. The observed full names manifest in a set of information sources $S = \{s_1, s_2, \ldots, s_m\}$, such that each source s_i consists of a set of extracted names N_i. For example, one can consider a single webpage as a single source. The set of distinct names observed in S is represented by $E = \{e_1, e_2, \ldots, e_n\} = N_1 \cup N_2 \cup \ldots \cup N_m$. In our problem, an observation (e.g., a first name) may correspond to several underlying entities *at face value* although it correctly references a single entity only; in this case we say that such an observation is *ambiguous* to multiple entities. An observation that refers to k different entities is called *k-ambiguous*. This is the scenario depicted in figure 1, where the name *Alice* correctly represents e_1 in s_1 and e_3 in s_3.

Our data differs from the typical observations in social and link analysis in that edges correspond to relationships between measurable characteristics, rather than entities, whereas the goal of the analysis is to understand the relational structure among unobservable entities. That is, we want to investigate the relational structure among entities by investigating an

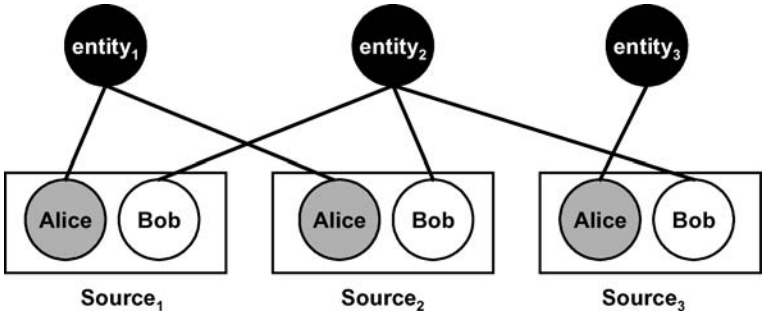

Figure 1. An example of a 1-ambiguous name (Bob) and a 2-ambiguous name (Alice).

ambiguous version of such structure. The goal of this research is to propose techniques to resolve the ambiguity; one leverages directly observed relationships, another incorporates unobserved, though meaningful, relations. The first technique is a version of hierarchical clustering on sources with ambiguous names only. The second constructs social networks from all sources, regardless of the existence of the ambiguous name of interest. The following sections explain these methods in detail.

3.1. Hierarchical Clustering

For the first method, each source is represented as a Boolean vector $s_i = [e_{i1}, \ldots, e_{in}]$, where $e_{ij} = 1$ if name e_j is in source s_i and 0 otherwise. Hierarchical clustering is performed using an average linkage criterion, which has been applied in prior disambiguation research (Bagga and Baldwin, 1998; Bhattacharya and Getoor, 2004; Mann and Yarowsky, 2003), which is calculated as follows (Duda et al., 2001). Each source to be clustered is initialized as a singleton cluster. Then, similarity between two clusters c_i, c_j, denoted $csim(c_i, c_j)$, is measured as

$$csim(c_i, c_j) = (|c_i||c_j|)^{-1} \sum_{s \in c_i, t \in c_j} ssim(s, t),$$

where the similarity between two sources s_i, s_j, denoted $ssim(s_i, s_j)$, can be measured using any distance or similarity function. The similarity function of choice for this research is one minus the cosine distance of the vectors of the two source vector representations. More specifically, cosine similarity between two sources is calculated as:

$$ssim(s_i, s_j) = \frac{\sqrt{\sum_{x=1}^{n} e_{ix} e_{jx}}}{\sqrt{\sum_{x=1}^{n} e_{ix}} \sqrt{\sum_{x=1}^{n} e_{jx}}}.$$

The most similar clusters are then merged into a new cluster. This process proceeds until

either a pre-specified stopping criterion is satisfied or all sources reside in one common cluster.

3.2. Random Walks and Network Cuts

An alternative method considered in this research is the analysis of social networks constructed via names with high certainty. Mainly, we are interested in the partitions of networks as prescribed by random walks from nodes of ambiguous names. One principle difference between the random walk method described in this section and the hierarchical clustering of the previous section is the walk is permitted to proceed over nodes (names) which occur in sources devoid of ambiguous names. By doing so, we exploit weak ties, which taken in combination, can lead to the discovery of community structures in the graph.

From the set of sources S, a social network is constructed in the following manner. Every distinct name in S is set as a node in the network. An edge exists between two nodes if the corresponding names collocate in a source at least one time. The weight of the edge between two nodes i, j is based on reasoning initially specified by Adamic and Adar (2003). In their research, users of an email list were related based on the number of topics in common and the popularity of each topic. In general, the likelihood two users were related was inversely proportional to the number of users mentioning the topic. For our research, we calculated the weight between two names as $w_{ij} = |\mathrm{s}|^{-1} \Sigma_{s \in S} \quad \theta_{ijs}$, where θ_{ijs} is an indicator variable with value 1 if names for nodes i and j collocate in source s and 0 otherwise. Our assumption is the fewer the number entities observed in a source, the greater the probability the entities have a strong social interaction. For instance, a website which depicts a list of all students, faculty, and staff of a university conveys less specific information than the class roster for a machine learning graduate course.

In order to group names (i.e., observations) that correctly refer to the same entity into the same cluste, r we start by constructing a network where each name is a node. Initially, we assume every name is a unique identifier for an entity, except for a single name to be disambiguated. An example network is depicted in figure 2 for the name *Alice*. In this network, *Gil* is indirectly connected to *Alice* through her acquaintances *Dan* and *Fran*.

Given this initial social network, we then proceed with random walks over the graph. Each walk begins at a node which represents the name of interest. The probability a step is taken from node a to node b is the normalized weight of the edge with respect to all edges originating from node a. This probability is calculated as $P(a \rightarrow b|a) = w_{ab} \Sigma_j w_{aj}$. Note the probability $P(a \rightarrow b|a) = 0$.

The random walk proceeds until either (1) a name node with the name of interest is encountered or (2) a maximum number of steps are taken. In our preliminary studies, we limited the maximum number of steps to 50.

3.2.1. Posterior Probability Base Heuristic. After a certain number of random steps, we approximate the posterior probability of reaching b given the walk originated at a and the observed network, which is represented as $P(a \Rightarrow \mathrm{b})$. The posterior probabilities inform us about which sets of observations are intimately connected given observations about their local social interactions. If two observations are close in their observable social space it

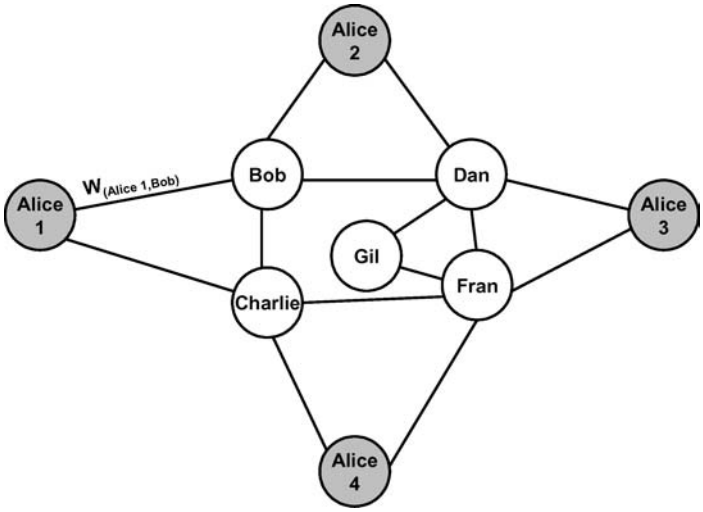

Figure 2. Social network with four ambiguous name observations.

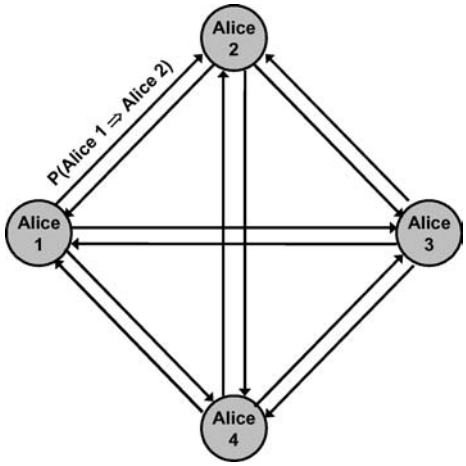

Figure 3. Posterior network of ambiguous observations from figure 2.

is reasonable to believe that they correspond to the same underlying entity. For example, figure 3 represents the posterior network for the ambiguous names of interest. The similarity between nodes a and b is set to the average of the probability of reaching a given b as a start node and vice versa, or $[P(a \Rightarrow b) + P(b \Rightarrow a)]/2$.

The advantage of this heuristic with respect to hierarchical clustering is mainly practical, that is, it is more intuitive to set a threshold on the set of posterior probabilities that the random walk returns, rather than to define a stopping criterion to threshold similarities in any hierarchical clustering method. A limitation of this method, however, is that the information

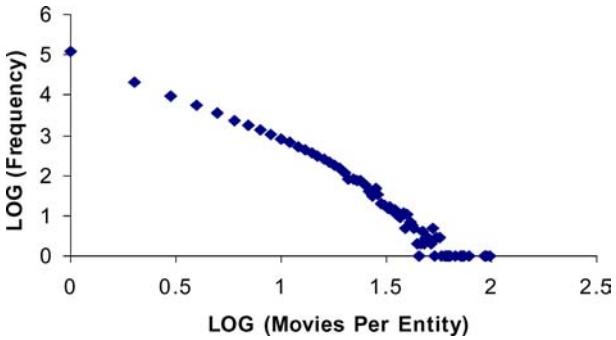

Figure 4. Log-log plot of movies per entity.

a random walk provides is much more substantial than the posterior probability of reaching one node from another, and yet our method clusters observations using only the latter probabilities. For example, the edge weights between name-pairs can inform about the size of the sources such name-pairs were extracted but not about the number of sources each name appeared or the number of random walk failures that were neglected.

3.2.2. Posterior Probability Full Normalization. To normalize the results produced by the random walk procedure we utilize information that was neglected in the basic heuristic, namely, the number of times random walks failed to reach an ambiguous name node. This information is useful in estimating the probability that a node will reach any other node, according to the intuition that the higher the number of random walk failures which originate at a and are supposed to end in b, the lower $P(a \Rightarrow \text{b})$ should be.

A related issue revolves around how much information failures should contribute to the similarity calculation. The absence of an edge means $P(a \Rightarrow b) = 0$, but in our procedure this can happens because of either (1) the absence of a path between a and b, or (2) the random walk did not find a path within the maximum number of steps. In the ideal case, the posterior probabilities would be dependent only on the first condition (i.e. the absence of a path). When this holds true, prior research on social network partitioning (Neville et al., 2003) suggests equal weighting of presence and absence of edges. To resolve this issue we modulate the weight of the number of failures with the parameter ε, so that the weight of non-failures is $(1 - \varepsilon)$. If ε reflects the expected number of failures than the cost function implicitly used by our fully normalized procedure will be consistent with spectral clustering and a normalized cut of the table of probabilities corresponding to a random walk without failures.

The intuition behind our fully normalized procedure is that the structure of pair-wise relations in the table of probabilities is captured by its coefficients of constant association (Bishop et al., 1975) and we plan to maintain those constant, while normalizing the table to having row and column entries sum to one with an iterative proportional fitting procedure (henceforth IPFP) (Fienberg, 1970). Intuitively, in step (a) we dampen the bias introduced by the failures; in step (b) we create a proper posterior probability table (rows/columns sum

to one) while maintaining the pair-wise (or local) correlation structure of the edge weights in the graph entailed by our random walks; in step (c) we project the nodes in the graph into the space of names (by dropping the node corresponding to failures), or alternatively, we extract the sub-graph that correspond to name observations from the larger graph that includes failures, while preserving the pair-wise (or local) correlation structure of the edge weights.

In more detail, in a situation with n names, (a) we first introduce information about failures, in terms of the frequency of random walks that failed starting from each name, in column $n + 1$, and we multiply entries in column $n + 1$ by ε, and entries in all other columns by $(1 - \varepsilon)$; (b) we then use IPFP to constrain the probability table to having rows and columns sum to on, while maintaining the pair-wise correlation structure; (c) we then remove column $n + 1$, and we use IPFP again to constrain row and column sums to one, while maintaining the pair-wise correlation sub-structure. Geometrically, IPFP keeps the normalized table of probabilities on the same hyper-surface as the initial one, as defined by the set of coefficients of constant association, thus maintaining its initial correlation structure (Airoldi et al., 2005).

The similarity scores are then averaged and clustered as before, such that edges are removed if their similarity is below a threshold value. Each resulting component of the graph corresponds to a particular latent variable, or entity. The set of names for each component correspond to the names for a particular entity.

3.3. F-Scores for Multiclass Accuracy

Given a clustering of names, we measure the accuracy of the predictions through the F-score (Larsen and Aone, 1999). This metric was initially introduced in the information retrieval community for testing the accuracy of clusters with greater than two predefined classes, such as the topics of webpages (e.g. baseball vs. football vs. tennis vs. etc.). As applied to disambiguation, the F-score is measured as follows. Let $H_e = \{h_1, h_2, \ldots, h_m\}$ be the set of entities referenced by a specific name. Let $S_e = \{s_{e1}, s_{e2}, \ldots, s_{em}\}$ be a set of sets of sources, such that s_{ei} corresponds to the set of sources that entity h_i occurs in. For this research, we only consider sources which contain a single occurrence of an ambiguous name. Thus, for all $s_{ei}, s_{ej} \in S_e$, $s_{ei} \cap s_{ej} = \emptyset$. Now, let $C = \{c_1, \ldots, c_k\}$ be a set of clusters of the sources in S_e. Furthermore, let $T = \{t_1, \ldots, t_k\}$ be the set of sources for each cluster in C.

The F-score is a performance measure, which uses the harmonic mean of precision and recall statistics for a multi-class classification system. In information retrieval, recall R is defined as the fraction of known relevant documents which were retrieved by the system. In contrast, precision P is defined as the fraction of the retrieved documents which are relevant. For a specific class in the system, which is simply an entity, we define recall and precision for an arbitrary cluster as $R(e_i, c_j) = |s_i \cap t_j|/|s_i|$ and $P(e_i, c_j) = |s_i \cap t_j|/|t_j|$, respectively. The F-score for an arbitrary entity-cluster pair, $f(e_i, c_j)$, which is referred to as the *local* F-score, is taken as the harmonic mean of the recall and precision, or $f(e_i, c_j) = 2R(e_i, c_j) * P(e_i, c_j)/(R(e_i, c_j) + P(e_i, c_j))$.

While the local F-score provides correlation for a single entity class and a single cluster, it is the complete system partitioning which we are interested in. To measure the accuracy

of the complete system we compute a global F-score, which is basically the sum of the largest local F-scores for each entity class. More specifically, the global F-score for an E, C pair is:

$$F(E, C) = \frac{\sum_{s \in S_e} |s| \max_{c \in C} f(e, c)}{\bigcup_{s \in S_e} s}$$

4. Experiments

In this section, we report results of the disambiguation strategies on a real world dataset. The dataset chosen to evaluate the methods was the Internet Movie Database (IMDB). A publicly available dataset was downloaded from the IMDB's ftp site and was parsed into a relational database for processing purposes (Internet Movie Database, 2004). The database contains approximately 115 years worth of actor lists for movies, television shows, straight to video and dvd. A subset of the IMDB dataset was chosen for evaluation purposes. This subset covered the ten year period 1994–2003 and consists of all movies with greater than 1 actor. For completeness purposes, the following summary statistics were gathered. There are ∼37,000 movies and ∼180,000 distinct entities. The distribution of number of movies per actor is depicted in figure 4, and it can be validated that it follows a log-log linear model, or power law distribution. The average number of entities per movie is 8 with a standard deviation of ∼9.9. Furthermore, it can be validated in figure 5 that the number of entities per movie follows a similar trend. As noted by Barabási and Albert (1999), the degree distribution of the actor-to-actor network constructed from IMDB data follows a power law distribution as well.

To construct a set of k-ambiguous names, entities were grouped by last name. There are ∼176,000 distinct last names. The distribution of number of entities per last name also follows a power law distribution, as shown in figure 6. For our experiments, we concentrated on 2-ambiguous names only. To put these numbers in perspective, there are approximately 18,000 2-ambiguous names. For resolution purposes, we the IMDB staff labels every entity

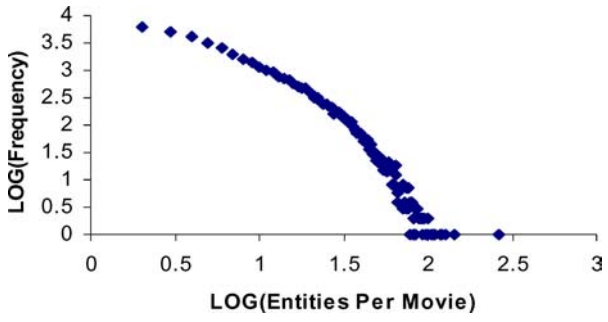

Figure 5. Log-log plot of entities per movie.

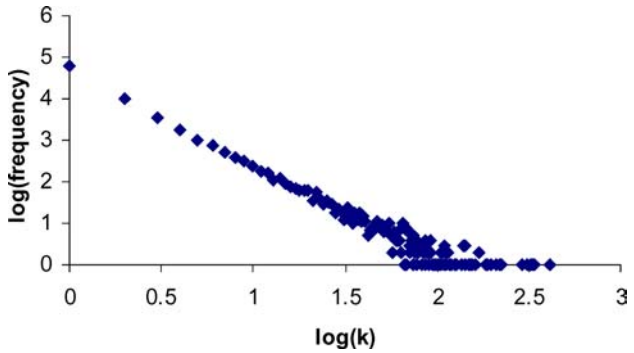

Figure 6. Log-log plot of k-ambiguous name frequency.

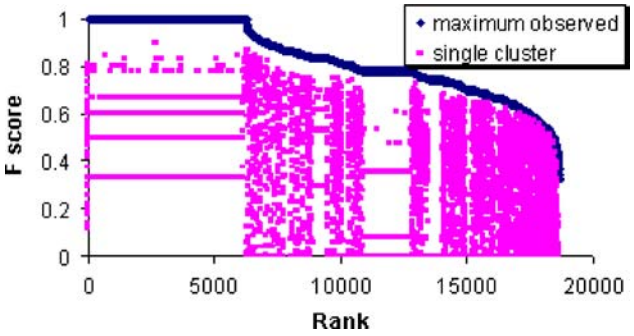

Figure 7. F-score of hierarchical clustering of sources for 2-ambiguous names. The topline corresponds to the best F-score achieved during clustering. The plot below is the difference of the best F-score minus a baseline F-score of all sources classified as a single cluster.

uniquely, so even entities with ambiguous names are provided with unique primary IDs in the form of an appended roman numeral (i.e. *John Doe (I)* vs. *John Doe (II)*). As a result, for each name subjected to disambiguation (e.g. *Doe*), we were able to guarantee that every other name was unambiguous. After disambiguation predictions were made, we used the underlying truth to generate F-scores.

4.1. Hierarchical Clustering Results

The IMDB dataset was subject to hierarchical clustering using the average linkage criterion described above. For clustering raw sources, we considered a continuum of similarity thresholds for stopping the clustering procedure. figures 7 through 9 depict the best global F-scores achieved for 2-ambiguous names from this dataset. The x-axis is ordered by best observed F-score. The predicted F-scores were compared against several baseline methods. In figures 7 through 9 the upperline corresponds to the best observed F-score. In

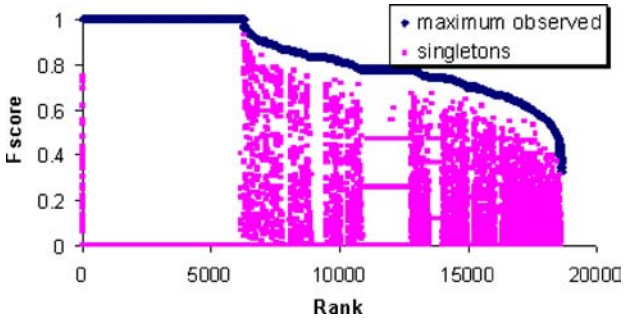

Figure 8. Same as figure 7, except the plot below corresponds to the difference of the best F-score minus a baseline F-score of all sources classified as a singleton clusters.

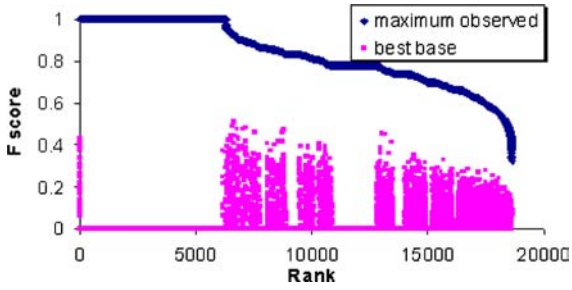

Figure 9. Same as figures 7 and 8, except the baseline is the difference between best F-score minus the maximum F-score of both single cluster and singleton baseline.

figures 7 and 8, the plot below the best score line corresponds to the difference between the best score and the baseline. The baseline method in figure 7 assumes all ambiguous names are distinct entities. In contrast, the baseline in figure 8 assumes all ambiguous names correspond to a single entity. These baselines are referred to as *AllSingletons* and *OneClusterOnly*, respectively. In figure 7, the first 70,000 points correspond to 1-ambiguous names, which explains is why the single cluster baseline predicts perfectly (i.e. F-score of 1).

To consider a more specific case where the baseline is not guaranteed to score perfectly, figure 9 depicts disambiguation results for 2-ambiguous names, where the number of sources is greater than 2. In contrast to figures 7 and 8, the plot in figure 9 presents the difference between the best F-score from hierarchical clustering and the maximum score achievable from a baseline method.

To an extent, the images of figures 7 through 9 skew the clustering prediction results. Though the plots imply that clustering provides F-scores above baseline scores, it must be taken into account that these are the best F-scores possible. The only way to discover the maximum F-score is to check the accuracy of each disambiguation prediction against the underlying truthful values. It is unfair to compare the power of hierarchal clustering to maximum F-score of the baseline tests for similar reasons. Just as we cannot consider all

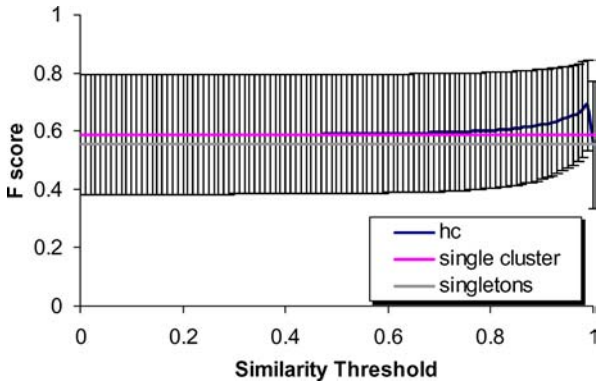

Figure 10. Average F-score of hierarchical clustering (hc), singletons, and single cluster baselines over continuum of global cosine similarity threshold values. The vertical lines correspond to 1 standard deviation.

partitions of the hierarchical clustering process simultaneously, we cannot simply take the max of both baselines—we must choose one or the other. In reality, an automated method must be able to find a point at which clustering automatically stops.

A simple method which was tested for automatic stopping was to average out the F-scores at various similarity threshold values. The resulting scores are demonstrated in figure 10 with the label "hc". In contrast to the previous experiements, the average F-scores for all singletons and single cluster baselines are reported. The vertical line in the graph depicts one standard deviation around the average hierarchical clustering F-score. A threshold of 0 corresponds to the *OneClusterOnly* baseline and a threshold of 1 corresponds to the *AllSingletons* baseline. In figure 8, as the threshold increases from 0 to 1, the F-score increases. The average F-score reaches a maximum value close to a similarity of 0.99, at which point the average F-score and all clusterings within 1 standard deviation achieve better than the best baseline of all singletons. This is encouraging, except with such a high similarity threshold it is implied that we should only merge clusters with extremely high structural equivalence in their vectors. This is quite peculiar, and appears to be completely antithetical to the belief that community structures permit greater capability for disambiguation.

4.2. Random Walk Results

However, once we consider the results from the random walk clustering, the previous findings appears to be less counterintuitive than initially implied. In figure 11 we present average the F-scores for random walk partitioning using similarity based on the raw affinity matrix. There were 100 random walks initiated from each ambiguous node. Recall, similarity is actually the mean of the probability of walking between ambiguous name observations a and b within 50 steps. The graph is then thresholded, such that probabilities below the threshold are removed, and the resulting network components are set as the predicted

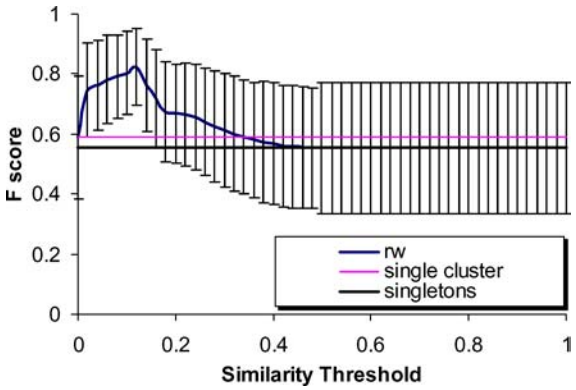

Figure 11. Average F-score of random walk network partitioning, singletons, and single cluster baselines over continuum of global similarity threshold values. Vertical lines correspond to ± 1 standard deviation.

clusters. From the plot in figure 11, it is apparent that a maximum F-score is achieved at a relatively low threshold, specifically a probability of $\sim$0.12. Moreover, the average F-score maximum at this point is greater that the maximum for simple hierarchical clustering by approximately 0.1. This is a significant improvement and supports the community structure hypothesis. Nodes and edges which are not directly related to the ambiguous names provide a significant amount of power for disambiguation purposes.

Given the significant improvement over hierarchical clustering, we continued with a subset of names for comparison of disambiguation using the raw counts matrix versus the IPFP normalized matrix. Specifically, we selected 500 names, such that each underlying entity occurred in at least 2 movies. In general, these names permit more variation in the F-scores and neither baseline model can produce an F-score of value 1. In figure 12 we depict the average F-score values over the continuum of threshold similarity scores. Our

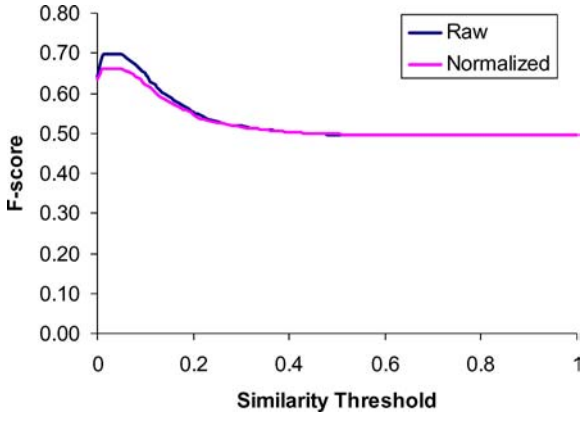

Figure 12. Average F-scores of raw and IPFP normalized walk matrices for sample of 500 2-ambiguous names.

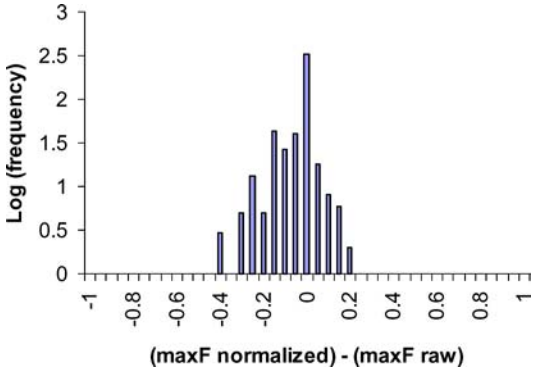

Figure 13. Distribution of best F-scores achieved using raw and IPFP normalized walk matrix for sample of 500 2-ambiguous names. Scores less than 0 favor raw matrix and scores greater than 0 favor IPFP matrix.

results indicate that clustering of the raw counts matrix outperforms the IPFP normalized matrix with failures on average. In this figure, we smoothed failures with ε set to 0.1 before normalizing with IPFP, but the exact same findings were observed for values of the smoothing parameter $\varepsilon = 0.01$ and 0.001.

The fact that raw probabilities outperform normalized probabilities suggests that failures are not very informative about the community structures which entities are engaged in. Moreover, normalization implies equally weighting presence and absence of a social tie in the (implicit) cost function can negatively impact performance. This intuition was confirmed by additional evaluations we attempted with spectral clustering with local scaling (Zelnik-Manor and Perona, 2004), which which ranked last in terms of disambiguation power (results not shown).

The trend in figure 12 seems to indicate normalization hinders the disambiguation procedure. Upon closer inspection, however, we discover such a conclusion might be too hasty. To investigate, we consider the difference in the best possible F-score achievable by both disambiguation methods across all threshold levels. In figure 13, we plot the distribution of best F-score achieved by the normalized matrix minus best F-score achieved by the raw counts matrix. Based on this difference, scores less than 0 along the X-axis correspond to cases where the raw counts matrix outperformed the normalized matrix. Similarly, scores greater than 0 along the X-axis favor the normalized matrix. For illustrative purposes, the y-axis is presented on a log scale. First, we note is that for approximately 320 cases ($\sim$64%) both methods display equivalent potential for disambiguation. Next, we note the distribution is left skewed, such that approximately 40 cases ($\sim$8%) favor the raw counts matrix. In comparison, the number of cases favored by the raw counts is about two times that of the normalized matrix (i.e. 20 cases).

This difference is not significant, thus, the disambiguation performance of the raw counts matrix cannot be statistically distinguished from that of the normalized matrix. Nonetheless, our results point to the raw counts matrix as the stronger candidate for measuring similarity (p-value ≈ 0.388), hence the simple heuristic should be preferred.

5. Discussion

The results of the previous section demonstrate community similarity provides an advantage over exact similarity for disambiguation. Yet, while the datasets which these results are derived correspond to real world observations, the experiments and models of disambiguation are based on a highly controlled environment. Some of the limitations of this environment, and possibilities for extension are addressed in the following sections.

5.1. Random Walks: Raw Counts, Failures and Normalization

In our experiments, the simple heuristic based on raw counts yielded the best disambiguation performance. The simple heuristic seemingly outperformed a sounder procedure that integrated information about the number of random walk failures.

The reason why failures are not informative to the disambiguation problem may be due in part to the IMDB's representation of social phenomena. Specifically, the inferred network is not necessarily indicative of personal relations because actors do not have complete control over whom they work with in a movie. Those decisions are made by external controllers of the cast, such as directors and producers, which are latent factors in the generative process for our observations not accounted for by our methodology. In addition, there may be other reasons, e.g., our model weights all actors in a movie equivalently. In other words, if there are two leads in a movie, say *Tom Cruise* and *Catherine Zeta-Jones*, and one extra, *John Smith*, then all three will be allocated the same relational weight for this movie. Clearly, not all actors in the same movie have the same relationship. For actors that are more prominent, accumulating weights over movies diminishes such relationships. Yet, for less prominent actors, we suspect their transient nature across communities and genres make them much more difficult to disambiguate. It is possible to overcome this problem, for example, by weighting actors according to their order in the movie. This information is difficult to derive from lists, and it may be more useful if our methods were evaluated on a dataset with more robust social and organization structures, such as those observed in the Enron dataset (Diesner and Carley, 2005; Klimt and Yang, 2004; Shetty and Adibi, 2004).

However, failures may provide more information about the underlying social network as we move away from a controlled setting. It appears that the extra variability in the F scores entailed by the raw probabilities lead to a better separation between names corresponding correctly to different entities. However, this may be a downside in real-world scenarios, where the global threshold must be chosen from limited information. In such a case the basic heuristic would entail more variable scores and thus a less robust threshold. This argues for the IPFP normalization procedure for the raw probabilities. Unfortunately, the value of the information added by the failures in a real-world scenario is still unclear. In fact, the limited number of steps introduces bias in the posterior probability estimates, that is, by introducing confusion between those paths that are not possible at all, and those that happen with low probability and are thus likely not to be discovered by a short random walk.

5.2. Building a Better Stopping Criteria

One limitation of this work stems from its dependency on a global threshold as a stopping criteria of the clustering process. This is an age old concern regarding hierarchical clustering and, for the most part, all stopping criteria are based on heuristics which are tailored to a researcher's respective environment. Airoldi and Malin (2004) have recently proposed a statistical test for stopping the clustering process based on geometric intuition regarding the growth rates of clusters. In their research, clustering utilizes a single linkage criterion and thus has yet to be proven if such geometric insights hold for more complex clustering criteria such as the average linkage method employed for this paper's analysis. It is possible such tests could be adapted and in future research we hope to address this issue in more depth.

Third, the random walks were arbitrary specified to time out after 50 steps. By this construction, a walk completed successfully (i.e. reaches an ambiguous name node) in 2 steps is given equal weight in the similarity measure than a successful walk of 50 steps. It is possible that a discounting model may be more appropriate, such that as the number of steps increases, the score provided to a successful completion tends toward zero. In future research we expect to design more formal probabilistic representations of community similarity.

5.3. Disambiguation in Uncertainty

Controlling for certainty is useful in the evaluation of the relative performance of disparate disambiguation procedures, but obviously this is an unrealistic assumption. In the real world, it is not clear if any observed name ever has complete certainty. This suggests probabilistic models of certainty may be useful for disambiguating names when many names are potentially ambiguous. For instance, strategies akin to expectation-maximization (Jensen and Neville, 2000; Kalashnikov et al., 2005) models over the graph have been considered.

With respect to this research, we propose a basic iterative algorithm, which can be used to cluster and classify relational data by leveraging names of high certainty, which can be fixed, or removed, during the learning process. By doing so, we can take advantage of high certainty knowledge to resolve lesser certain situations.

6. Conclusion

This paper evaluated several methods for disambiguating names in a relational environment (actor collaborations in the Internet Movie Database) were presented. First, we implemented a baseline method, modeled on prior research, which used hierarchical clustering of sources in which ambiguous names are observed. We then introduced a novel alternative which leveraged social networks constructed from all sources, such that random walks originating from ambiguous name nodes were used to estimate posterior distributions of relations to partition the graph into components. We controlled social networks to study a single ambiguous name, and our findings suggest methods which leverage community, in contrast to exact, similarity provide more robust disambiguation capability. In the future we expect to evaluate our methods on data that is more indicative of real world social interactions

and extend our methods to account for networks that consist of more than one ambiguous name. We suspect the limited number of steps introduces bias into the posterior probability estimates, and intend to explicitly compute and control for the bias.

Acknowledgment

The authors wish to acknowledge useful discussions with various members of the Data Privacy Laboratory, especially Dr. Latanya Sweeney, and various attendees of the SIAM 2005 Workshop on Link Analysis, Counterterrorism, and Security, where this work was originally presented, notably Dr. Yun Chi.

Note

1. A failure occurs every time a random walk from a to b is terminated because it reaches the maximum number of steps, rather than because it reaches its target node, i.e., b in this case.

References

Adamic, L. and E. Adar (2003), "Friends and Neighbors on the Web," *Social Networks*, 25(3), 211–230.

Airoldi, E., A. Slavkovic, S. Fienberg (2005), "Interactive Tetrahedron Applet: A Tool for Exploring the Geometry of 2×2 Contingency Tables," *Department of Statistics Technical Report CMU-STAT-05-824*, Carnegie Mellon University: Pittsburgh, PA.

Airoldi, E. and B. Malin (2004), "Data Mining Challenges for Electronic Safety: The Case of Fraudulent Intent Detection in E-mails," in *Proceedings of the IEEE Workshop on Privacy and Security Aspects of Data Mining*, Brighton, England, pp. 57–66.

Albert, R. and A.L. Barabási (2002), "Statistical Mechanics of Complex Networks," *Reviews of Modern Physics*, 74, 47–97.

Bagga, A. and B. Baldwin (1998), Entity-based Cross-Document Coreferencing Using the Vector Space Model," in *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics*, San Francisco, CA, pp. 79–85.

Banko, M. and E. Brill (2001), "Scaling to Very Large Corpora for Natural Language Disambiguation," in *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, pp. 26–33.

Barabási, A.L. and R. Albert (1999), "Emergence of Scaling in Random Networks," *Science*, 286, 509–512.

Bekkerman, R. and A. McCallum (2005), "Disambiguating Web Appearances of People in a Social Network," in *Proceedings of the 2005 World Wide Web Conference*, Chiba, Japan.

Bhattacharya, I. and L. Getoor (2004a), "Iterative Record Linkage for Cleaning and Integration," in *Proceedings of the 9th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, Paris, France, pp. 11–18.

Bhattacharya, I. and L. Getoor (2004b), "Deduplication and Group Detection Using Links," in *Proceedings of the 2004 ACM SIGKDD Workshop on Link Analysis and Group Detection*, Seattle, WA.

Bishop, Y., S. Fienberg and P. Holland (1975), *Discrete Multivariate Analysis: Theory and Practice*, The MIT Press, Cambridge, MA.

Brill, E. and P. Resnick (1994), "A Rule-based Approach to Prepositional Phrase Attachment Disambiguation," in *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan, pp. 1198–1204.

Brown, P., S. Della Pietra, V. Della Pietra and R. Mercer (1991), "Word-sense Disambiguation using Statistical Methods," in *Proceedings of the 29*th *Annual Meeting of the Association for Computational Linguistics*, Berkeley, CA, pp. 264–270.

Chan, S. and J. Franklin (1998), "Symbolic Connectionism in Natural Language Disambiguation," *IEEE Transactions on Neural Networks*, 9(5), 739–755.

Chao, G. and M.G. Dyer (2000), "Word Sense Disambiguation of Adjectives using Probabilistic Networks," in *Proceedings of the 17th International Conference on Computational Linguistics*, Saarbrucken, Germany, pp. 152–158.

Coffman, T., S. Greenblatt and S. Marcus (2004), "Graph-Based Technologies for Intelligence Analysis," *Communications of the ACM*, 47(3), 45–47.

Cohen, W., P. Ravikumar and S. Fienberg (2003), "A Comparison of String Matching Tasks for Names and Addresses," in *Proceedings of the IJCAI Workshop on Information Integration on the Web*, Acapulco, Mexico.

Culotta, A., R. Bekkerman and A. McCallum (2004), "Extracting Social Networks and Contact Information from Email and the Web," in *Proceedings of the First Conference on Email and Anti-Spam*, Mountain View, CA.

Diesner, J., and K. Carley (2005), "Exploration of Communication Networks from the Enron Email Corpus," in *Proceedings of the 2005 SIAM Workshop on Link Analysis, Counterterrorism and Security*, Newport Beach, CA, pp 3-14.

Duda, R.O., P.E. Hart and D.G. Stork (2001), *Pattern Classification, 2nd Edition*, Wiley, New York, NY.

Fienberg, S. (1970), "An Iterative Procedure for Estimation in Contingency Tables," *Annals of Mathematical Statistics*, 41(3), 907–917.

Gale, W.A., K.W. Church and D. Yarowsky (1992), "A Method for Disambiguating Word Senses in Large Corpora," *Computers and Humanities*, 26, 415–439.

Ginter, F., J. Boberg, J. Jarvinen and T. Salakoski (2004), "New Techniques for Disambiguating in Natural Language and Their Application to Biological Text," *Journal of Machine Learning Research*, 5, 605–621.

Girvan, M. and M. Newman (2002), "Community Structure in Social and Biological Networks," in *Proceedings of the National Academy of Sciences, USA*, 99, 7821–7826.

Hatzivassiloglou, V., P.A. Duboue and A. Rzhetsky (2001), "Disambiguating Proteins, Genes, and RNA in text: A Machine Learning Approach," *Bioinformatics*, 17, 97–106.

Internet Movie Database. http://www.imdb.com. Accessed June 20, 2004.

Harada, M., S. Sato and K. Kazama (2004), "Finding Authoritative People on the Web," in *Proceedings of the Joint Conference on Digital Libraries*, Tucson, AZ.

Hiro, K, H. Wu and T. Furugori (1996), "Word-Sense Disambiguation with a Corpus-Based Semantic Network," *Journal of Quantitative Linguistics*, 3, 244–251.

Jaro, M. (1989) "Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida," *Journal of the American Statistical Association*, 89, 414–420.

Jensen, K. and J.L. Binot (1987), "Disambiguating Prepositional Phrase Attachments by Using Online Definitions," *Computational Linguistics*, 13(3/4), 251–260.

Jensen, D. and J. Neville (2000), "Iterative Classification in Relational Data," in *Proceedings of the AAAI-2000 Workshop on Learning Statistical Models From Relational Data*, pp. 13–20.

Kalashnikov, D., S. Mehotra and Z. Chen (2005), "Exploiting Relationships for Domain-independent Data Cleaning," in *Proceedings of the 2005 SIAM International Conference on Data Mining*, Newport Beach, CA, pp. 262–273.

Klimt, B. and Y. Yang (2004), "The Enron Email Corpus: A New Dataset for Email Classification Research," in *Proceedings of the 15th European Conference on Machine Learning*, Pisa, Italy, pp. 217–226.

Larsen, B. and C. Aone (1999), "Fast and Effective Text Mining Using Linear-time Document Clustering," in *Proceedings of the 5*th *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, CA, pp. 16–22.

Lesk, M. (1986), "Automatic Sense Disambiguation: How to Tell a Pine Cone from an Ice Cream Cone," in *Proceedings of the 1986 ACM SIGDOC Conference*, New York, NY, pp. 24–26.

Malin, B. (2005), "Unsupervised Name Disambiguation via Social Network Similarity," in *Proceedings of the 2005 SIAM Workshop on Link Analysis, Counterterrorism, and Security*, Newport Beach, CA, pp. 93–102.

Mann, G. and D. Yarowsky (2003), "Unsupervised Personal Name Disambiguation," in *Proceedings of the 7th Conference on Computational Natural Language Learning*, Edmonton, Canada, pp. 33–40.

Neville, J., M. Adler and D. Jensen (2003), "Clustering Relational Data using Attribute and Link Information," in *Proceedings of the IJCAI Text Mining and Link Analysis Workshop*, Acapulco, Mexico.

Newman, M. (2003), "The Structure and Function of Complex Networks," *SIAM Review*, 45, 167–256.

Ng, H.T. (1997), "Exemplar-Based Word Sense Disambiguation: Some Recent Improvements," in *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, *Association for Computational Linguistics*, Somerset, New Jersey, pp. 208–213.

Shetty, J. and J. Adibi (2004), "Enron Email Dataset: Database Schema and Brief Statistical Report," Information Sciences Institute Technical Report, University of Southern California, 2004.

Sweeney, L. (2004), "Finding Lists of People on the Web," *ACM Computers and Society*, 34(1).

Thompson, P. (2005), "Text Mining, Names, and Security," *Journal of Database Management*, 16(1), 54–59.

Vronis, J. and N. Ide (1999), "Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionaries," in *Proceedings of the 13th International Conference on Computational Linguistics*, Helsinki, Finland, pp. 389–394.

Wacholder, N., Y. Ravin and M. Coi (1997), "Disambiguation of Proper Names in Text," in *Proceedings of the 5th Applied Natural Language Processing Conference*, Washington, DC, pp. 202–208.

Wei, J. (2004), "Markov Edit Distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3), 311–321.

Winkler, W. (1995), "Matching and Record Linkage," in Cox, B. et al. (ed.), in *Business Survey Methods*, Wiley, New York, NY, pp. 355–384.

Yarowsky, D. (1992), "Word-sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora," in *Proceedings of the 30*th *Annual Meeting of the Association for Computational Linguistics*, Nantes, France, pp. 454–460.

Zelnik-Manor, L. and P. Perona (2004), "Self-Tuning Spectral Clustering," in *Advances in Neural Information Processing Systems 17*, Vancouver, Canada, pp. 1601–1608.

Bradley A. Malin is a Ph.D. candidate in the School of Computer Science at Carnegie Mellon University. He is an NSF IGERT fellow in the Center for Computational Analysis of Social and Organizational Systems (CASOS) and a researcher at the Laboratory for International Data Privacy. His research is interdisciplinary and combines aspects of bioinformatics, data forensics, data privacy and security, entity resolution, and public policy. He has developed learning algorithms for surveillance in distributed systems and designed formal models for the evaluation and the improvement of privacy enhancing technologies in real world environments, including healthcare and the Internet. His research on privacy in genomic databases has received several awards from the American Medical Informatics Association and has been cited in congressional briefings on health data privacy. He currently serves as managing editor of the Journal of Privacy Technology.

Edoardo M. Airoldi is a Ph.D. student in the School of Computer Science at Carnegie Mellon University. Currently, he is a researcher in the CASOS group and at the Center for Automated Learning and Discovery. His methodology is based on probability theory, approximation theorems, discrete mathematics and their geometries. His research interests include data mining and machine learning techniques for temporal and relational data, data linkage and data privacy, with important applications to dynamic networks, biological sequences and large collections of texts. His research on dynamic network tomography is the state-of-the-art for recovering information about who is communicating to whom in a network, and was awarded honors from the ACM SIG-KDD community. Several companies focusing on information extraction have adopted his methodology for text analysis. He is currently investigating practical and theoretical aspects of hierarchical mixture models for temporal and relational data, and an abstract theory of data linkage.

Kathleen M. Carley is a Professor of Computer Science in ISRI, School of Computer Science at Carnegie Mellon University. She received her Ph.D. from Harvard in Sociology. Her research combines cognitive science, social and dynamic networks, and computer science (particularly artificial intelligence and machine learning techniques) to address complex social and organizational problems. Her specific research areas are computational social and organization science, social adaptation and evolution, social and dynamic network analysis, and computational text analysis. Her models meld multi-agent technology with network dynamics and empirical data. Three of the large-scale tools she and the CASOS group have developed are: BioWar a city, scale model of weaponized biological attacks and response; Construct a models of the co-evolution of social and knowledge networks; and ORA a statistical toolkit for dynamic social Network data.