Disambiguating Web Appearances of People in a Social Network

Ron Bekkerman
Dept. of Computer Science
University of Massachusetts
Amherst, MA 01003, USA
ronb@cs.umass.edu

Andrew McCallum
Dept. of Computer Science
University of Massachusetts
Amherst, MA 01003, USA
mccallum@cs.umass.edu

ABSTRACT

Say you are looking for information about a particular person. A search engine returns many pages for that person's name but which pages are about the person you care about, and which are about other people who happen to have the same name? Furthermore, if we are looking for multiple people who are related in some way, how can we best leverage this social network? This paper presents two unsupervised frameworks for solving this problem: one based on link structure of the Web pages, another using Agglomerative/Conglomerative Double Clustering (A/CDC)—an application of a recently introduced multi-way distributional clustering method. To evaluate our methods, we collected and hand-labeled a dataset of over 1000 Web pages retrieved from Google queries on 12 personal names appearing together in someones in an email folder. On this dataset our methods outperform traditional agglomerative clustering by more than 20%, achieving over 80% F-measure.

Categories and Subject Descriptors

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

General Terms

Algorithms, Management, Experimentation

Keywords

Web appearance, name disambiguation, social network, document clustering, link structure, information bottleneck.

1. INTRODUCTION

We face an era not only of an information explosion, but also a tremendous increase in the extent of our relations to other people. We are constantly presented with new people names, chances to meet and communicate with people, and opportunities to add people to our social network—in our work, from the media, and from our social and business use of the Internet. It is now common that we do not actually meet (or even phone) our acquaintances; instead we communicate through email, chatrooms and discussion forums. We correspond with hundreds of people simultaneously. Our social network is tens of times larger than that of our grandparents, and will likely grow more with time. Even when we have trouble tracking all these connections, we (intentionally or unintentionally) add new ones.

We are in need of personalized tools that will help us manage our social network—both to track people we know already, and also to tell us about new people we meet. For example, when we receive email messages from people whose names we do not know, it would be useful to be able to rapidly search for any public facts about them. This may help us know how to rate the importance of the message, or prioritize our effort in making replies. For example, a message from the head of an industrial research lab who works in your research area may warrant a higher priority than a corporate recruiter working for a company with little relation to you, even when the remainder of the body of the message is substantially similar. A useful summary of public information about a person could often be gathered from the Web: news articles, corporate pages, university pages, discussion forums, etc. contain a lot of information about people. But how would the system identify whether certain Web pages are about the person in question or a different person with the same name? Can we find not just a few pages, but a comprehensive set of pages?

For example, consider David Mulford,[1] the US Ambassador to India. When the query *"David Mulford"* is issued as a query to Google, most of the pages retrieved are actually related to the Ambassador; however, there are also two business managers, a musician, a student, a scientist, and a few others. If we are looking for information about a particular person, we want to filter out information about other namesakes, while also preserving the maximum amount of relevant information. It is sometimes quite difficult to determine if a page is about a particular person or not. In case of Ambassador David Mulford, much of the information that can be found at first may seem to be unrelated: one site states that in the late 1950s David attended Lawrence University and was a member of its athletic team; other sites mention his work at different positions in governmental departments and commercial structures, including Chairman International of Credit Suisse First Boston (CSFB) in London; a few sites (mostly in Spanish) relate his name to a financial scandal in Argentina. It is a difficult challenge to automatically determine whether all of these sites discuss the same person.

[1]An example name actually appearing in our dataset described in Section 4.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-046-9/05/0005.

In previous work [5] we addressed the problem of automatically populating a database of contact information of people in a user's social network. Given a personal name extracted out of a user's mailbox, we queried Google in order to locate the person's homepage. We then applied conditional random fields [12] to extract institution, job title, address, phone, fax, email and other information from the homepage. The main problem of our homepage finding approach was that we used a simple heuristic for disambiguating person names, which sometimes failed. So, in some cases we extracted the contact information of *namesakes* of people from the user's social network.

In this paper, we address the problem not simply of finding homepages, but finding *all* search engine hits corresponding to a person, and separating them from hits about namesakes. We look beyond homepages because significant further information is often found elsewhere. Moreover, the person's homepage may be old and abandoned, containing out-of-date information, and this may be discovered if we have a broader view on the person's Web appearances.

Rather than using simple heuristics, we present results with two statistical frameworks for addressing this problem: one based on link structure, and another based on the recently introduced multi-way distributional clustering method [3]. Furthermore, and crucially, rather than searching for people individually, we leverage an existing social network of people, or lists of people who are known to be somewhat connected, and use this extra information to aid the disambiguation.

2. PROBLEM STATEMENT AND RELATED WORK

We state the problem of Web presence identification as inferring a model that ultimately provides a function f answering whether or not Web page d refers to a particular person h, given a model $\mathcal{M}$ and background knowledge $\mathcal{K}$.

Obviously, the perfect background knowledge $\mathcal{K}$ is in most cases unavailable, so the discrimination process must be made using some limited available information. Note that given no background knowledge at all, the problem becomes ill-defined: in order to automatically perform the task, the person h must have an electronic representation, which cannot be built without having any prior knowledge about the person.

The background knowledge $\mathcal{K}$ can be of various kinds. For instance, $\mathcal{K}$ can include training data—pages that are related or unrelated to the person. In this case, the problem is reduced to a binary classification task that is widely addressed in the machine learning literature of the past decade (see, e.g., [11]). However, in real-world situations, labeled examples are difficult and expensive to obtain. Positive instances of a person's Web presence could possibly be obtained by making use of the person's email messages, but obtaining negative instances could be much more difficult. In this paper, we employ unsupervised solutions.

The problem of disambiguating collections of Web appearances has been explored surprisingly little. There has been much work on homepage finding, starting from the early years of the Internet. In 1997 Shakes et al. [17] launched *AHOY!*—the first system for homepage finding. They primarily used heuristics and pattern matching for recognizing URLs of homepages. Later on, standard IR techniques have been used for this task. The TREC homepage finding competition was held in 2002 (see, e.g., [1]).

The problem of person name disambiguation has been approached in the domain of research paper citations (see, e.g., [10]), with various supervised methods proposed for its solution. There has been some research on person name disambiguation in the Web domain [2, 13, 7], within the general framework of entity coreference (see, e.g. [16, 9]). Agglomerative clustering has been applied in all three. Bagga and Baldwin [2] use agglomerative clustering over traditional vector space models of text windows around a personal name mention. Mann and Yarowsky [13] propose a richer document representation involving automatically extracted features. Their clustering technique however can be basically used only for separating two people with the same name. Recently, Fleischman and Hovy [7] construct a MaxEnt classifier to learn distances between documents that are then clustered. This method needs to be provided with a large training set.

Note that these all use average-link clustering methods: the distance between data points and cluster centroids is considered, not the distance between individual data instances. This lacks the benefits of transitivity: if page d_1 is related to the same person as page d_2, while page d_2 is related to the same person as page d_3, then pages d_1 and d_3 are probably related to the same person, although the distance between them can be relatively large.

In this paper we propose two Web appearance disambiguation methods that also involve clustering, but are better adapted to our specific task at hand. The first method is based on the link structure of Web pages. This method focuses on constructing only one cluster (of relevant pages), which nicely fits into our binary framework. The second technique employs Agglomerative/Conglomerative Double Clustering (A/CDC)—an application of a new multi-way distributional clustering method [3], which does not directly compute distances between clusters. The A/CDC objective can be also derived from the Multivariate Information Bottleneck (MIB) clustering principle [8]. In addition, we experiment with a hybrid approach combining the Link Structure and A/CDC methods. All three of these methods outperform a baseline agglomerative clustering technique by more than 20% F-measure on a large real-world dataset.

In our attempts to use as little background knowledge as possible, we propose the following application scenario: given a *group* of people $H = \{h_1, \ldots, h_N\}$ who are related to each other, we would like to identify the Web presence of *all of them* simultaneously. Therefore, instead of solving one problem, we solve N interrelated problems: for each person h_i in the group H we find Web pages that refer to h_i.

Dealing with a group of people instead of dealing with an individual is not overly burdensome. One can imagine many situations where a personal name is given within the context of people whom the person communicates with. Examples include coauthors of a scientific paper, participants in a newsgroup, or correspondents in a user's email. Moreover, given a separate name without any additional information about the person, it is often fundamentally ambiguous to whom it refers. But given a group of names of connected people, we can usually see to what group of people it refers, even if we do not know some of the names in the group.

For example, when searching for a person on the Web, one personal name is usually ambiguous. Although Google finds

only one person named *Ron Bekkerman*, it finds at least a dozen of unrelated people named *Andrew McCallum*. However, if both names are provided to Google, pages that refer to only one of those Andrew McCallums will be retrieved. Thus, as little background knowledge about the person as his or her membership in a group of people makes the Web appearance disambiguation problem feasible.

3. METHODS

We will now describe our three proposed methods of solving the Web appearance disambiguation problem.

3.1 Link Structure Model

An important observation is that Web pages of a group of acquaintances are likely to be interconnected. On the other hand, it is hard to imagine that pages of their namesakes would be interconnected. Indeed, the namesakes probably have nothing in common, while the actual people of the group often tend to maintain homepages on the same domain (when they are colleagues), tend to refer to the same resources, and tend to be referred to from the same Web sites. However, the existence of a direct hyperlink from one relevant page to another may be rare, so the term "interconnectedness" should be carefully defined (see Section 3.1.1). Meanwhile we define that two Web pages are *linked to each other* if their hyperlinks share something in common.

According to the problem statement in Section 2, we construct a function f that discriminates between relevant and irrelevant pages d for a person h with name t_h. Our background knowledge $\mathcal{K}$ is a set of names $T_H = \{t_{h_1}, \ldots, t_{h_N}\}$ in a group of N people in a user's social network. Our set of Web pages D is constructed by providing a search engine with queries $t_{h_1}, \ldots, t_{h_N}$ and retrieving top K hits for each one of the query, so that $N \times K$ Web pages are retrieved overall. Note that in this way every page d is already associated with a personal name t_{h_i}: the name t_{h_i} was in fact the query that retrieved page d. However, it is yet unknown whether the page d refers to the actual person h or to his/her namesake (or to neither). We now construct our model $\mathcal{M}$ given the set of Web pages D.

Let graph $G_{LS} = (V, E)$ be the *Link Structure Graph* over a set of Web pages D if nodes of the graph are the Web pages ($V \equiv D$) and there exists an edge between any pair of nodes d_i and d_j iff d_i and d_j are *linked to each other*.

In graph G_{LS} linked Web pages compose connected components. We naturally expect relevant pages to interconnect much more than irrelevant pages would interconnect. Of special importance is that relevant pages that refer to *different* people are likely to interconnect, while irrelevant pages that refer to different people would probably not connect to each other. We might decide that the Maximal Connected Component (MCC) of graph G_{LS} consists of only relevant pages, so the MCC would be the "core" of our model. However, there can be a case where the MCC consists only of Web pages retrieved in response to a *single* query—this can happen when pages of one person h are heavily interconnected. If this person h appears to be an irrelevant namesake of a relevant person, such MCC will be totally irrelevant. Therefore, we come up with the following definition:

Definition 1. Let us denote *central cluster* C_0 as the largest connected component in G_{LS} that consists of pages retrieved by more than one query.

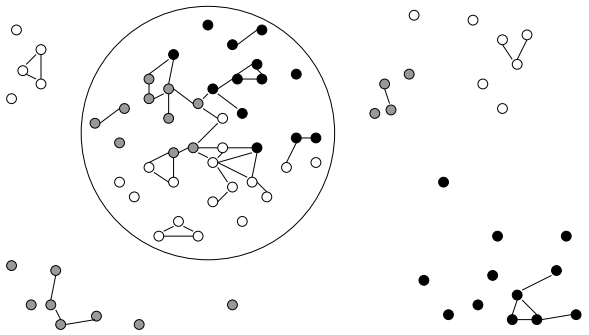

Figure 1: Relevant and irrelevant Web pages according to the Link Structure model. Relevant pages are within the δ-radius from the *"central cluster"*. White, gray and black colors indicate that the pages are retrieved by three different queries.

We denote other connected components in graph G_{LS} as *clusters* $C_1, \ldots, C_M$, where $M < N \times K$. We are now ready to define our link structure model:

Definition 2. The *Link Structure Model* $\mathcal{M}_{LS}$ is a pair $(\mathcal{C}, \delta)$, where $\mathcal{C}$ is the set of all connected components of the graph G_{LS} (note that $C_0 \in \mathcal{C}$), and δ is a distance threshold.

So, our discrimination function f is defined as:

$$f(d, h|\mathcal{M}(\mathcal{K})) = \begin{cases} 1, & \text{if } d \in C_i : \ \|C_i - C_0\| < \delta, \ i = 0..M \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The intuition behind this definition is that the pages of the central cluster and of a few clusters that are close to the central cluster are considered to be relevant, while others are irrelevant. Figure 1 illustrates this intuition.

3.1.1 Particular design choices

In the description of our Link Structure model we intentionally did not specify the following design choices:

1. How to decide whether two pages are linked or not.
2. How to choose a suitable value for δ.
3. How to calculate the distance between two clusters C_0 and C_i.

These are implementation details that can vary from system to system. For example, two pages can be considered as linked if both contain a hyperlink to the same page, or both are hyperlinked from one page, or one page can be reached within three hyperlink hops from the other. Different approaches can also be considered, for example, two pages are linked if both mention the same organization, e.g., *University of Massachusetts at Amherst*.

Similarly, the distance measure between two clusters can be different. For example, it can be the cosine similarity or Kullback-Leibler divergence. It can be learned using MaxEnt classification as proposed in [7, 15]. It can also be the distance not between clusters themselves, but between their closest elements. In this case the discrimination function f

would be redefined as

$$f(d,h) = \begin{cases} 1, & \text{if } d \in C_i : \ \exists d_i \in C_i \ \exists d_j \in C_0 \ \|d_i - d_j\| < \delta \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

In our experimental setup we have made the following design choices:

1. **Linked pages.** For this work, we decided to only consider the hyperlink structure of the pages. Since the full URLs of the hyperlinks seem to be too specific, while the URL domains seem to be too general, we define a function $url(d)$ to output the domain of the d's URL with its first directory in case this directory exists. For example. given page d_1 with URL `http://www.cs.umass.edu/~ronb/timeline.html` the function $url(d_1)$ will return `www.cs.umass.edu/~ronb`. Given page d_2 with URL `http://www.cs.umass.edu/` the function $url(d_2)$ will return `www.cs.umass.edu`. For the reminder of the section, by URL we mean the output of $url(d)$.

   We define the set POP to be a set of URLs of extremely popular domains, such as `www.amazon.com`. The popularity of a domain can be determined using operator `:link` of the Google command line. We define the set $TR(D)$ of *trusted URLs* as $\{url(d_i)\} \setminus POP$. We also define the function $links(d)$ that given page d returns a set of URLs that occur in d.

   Definition 3. The *link structure* $LS(d)$ of a page d is defined as $LS(d) = (links(d) \cap TR(D)) \cup url(d)$.

   So, the link structure of a page is its own URL and its hyperlinks given that they appear as URLs of other pages in the dataset. By this we minimize undesirable hazards that can occur if a page contains too many hyperlinks, pretending to be a hub.

   Definition 4. Two pages d_1 and d_2 are *linked to each other* if their link structures intersect, that is $LS(d_1) \cap LS(d_2) \neq \emptyset$.

2. **Distance threshold.** We do not explicitly set the distance threshold δ. Instead, we set it so that one third of the pages in the dataset are within the threshold.

3. **Distance measure between clusters.** We applied cosine similarity with a novel variation of the *tfidf* term weighting function:

$$tfidf(w) = \frac{tf(w)}{\log google\_df(w)}, \quad (3)$$

   where $google\_df(w)$ is the *estimated total results count* of the term w if being provided as a query to Google. This document frequency count seems to be the most adequate measurement of the commonness of the term. The estimated total results counts of words in our dataset were obtained using Google API.[2]

[2]http://www.google.com/apis/

3.2 Agglomerative/Conglomerative Double Clustering (A/CDC) Model

The problem of Web appearance disambiguation can be addressed within the standard clustering framework: the set of Web pages D is split into M clusters, then one of the clusters is considered as containing only relevant pages while all the other clusters are irrelevant. The decision about which one of the M clusters is the relevant one can be made based on either internal or external information. An internal resource might be the measure of interconnectedness of the clusters, in the sense of the discussion in Section 3.1. The most interconnected cluster is then chosen as relevant. An external resource can be, e.g., email messages from all or some people in the group: the distance between the set of messages and each one of the clusters is computed, then the closest cluster is chosen. Since we intend to minimize the background knowledge about the people, we adopt the former technique.

So, our *Clustering Model* $\mathcal{M}_{CL}$ is a pair $(\mathcal{C}, L(\cdot))$, where $\mathcal{C}$ is the set of clusters of documents in D, and $L(\cdot))$ is the interconnectedness measure of a cluster.

Then the discrimination function f is defined as follows:

$$f(d,h|\mathcal{M}(\mathcal{K})) = \begin{cases} 1, & \text{if } d \in C^* : \ C^* = \arg\max_{i=1..M} L(C_i) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

As our particular clustering method, we apply the A/CDC algorithm—an instance of the new *multi-way distributional clustering* (MDC) method we propose in [3]. The main idea of A/CDC is to employ the fact that similar documents have similar distributions over words, while similar words are similarly distributed over documents. Starting with one cluster containing all words and many clusters with one document each, we iteratively split word clusters and merge document clusters, while conditioning one clustering system on the other, until meaningful clusters are obtained. This method has demonstrated high performance on various datasets including the benchmark 20 Newsgroups.

Multi-way distributional clustering stands in close correspondence with the Multivariate Information Bottleneck (MIB) method. The A/CDC algorithm, while being the simplest MDC application, can also be derived from MIB, which will be shown in this section. We first provide some background on related Information Bottleneck methods, then discuss motivation of the A/CDC approach and overview the A/CDC algorithm.

3.2.1 Background

The *Information Bottleneck (IB)* method [21] is a convenient information-theoretic framework for solving various real-world problems, especially clustering. It has been widely applied in Information Retrieval [20, 4, 18]. The main idea that lies behind the IB clustering is in constructing an assignment of data points X into clusters $\tilde{X}$ that will maximize information about entities Y that are interdependent with X. The information about Y gained from $\tilde{X}$ is represented in terms of *Mutual Information*:

$$I(\tilde{X};Y) = \sum_{\tilde{X},Y} P(\tilde{X},Y) \log \frac{P(\tilde{X},Y)}{P(\tilde{X})P(Y)}. \quad (5)$$

A natural constraint is imposed on the Mutual Information between data instances X and their clusters $\tilde{X}$: it penalizes

Mutual Information $I(\tilde{X};X)$ from being too large because otherwise the clustering will tend to be degenerative (each instance will form a cluster). This constraint is referred to as the *compression constraint*. Thus, the Information Bottleneck problem is stated as

$$\arg\max_{\tilde{X}} I(\tilde{X};Y) - \beta I(\tilde{X};X), \tag{6}$$

where β is a Lagrange multiplier.

Many applications and extensions of the original IB method have been proposed. Some relevant results are listed below. Slonim and Tishby [19] propose a greedy agglomerative algorithm for document clustering based on the Information Bottleneck method, where X stands for documents and Y stands for words in the documents. This simple algorithm achieves surprisingly good results but is computationally expensive. Slonim et al. [18] propose a greedy sequential IB clustering algorithm based on local optimization that demonstrates incredibly high performance and is computationally efficient in practice.

Slonim and Tishby [20] notice that the IB method is symmetric in X and Y. They propose a double clustering technique in which words are first clustered with respect to documents and documents are then clustered with respect to *clusters* of words. El-Yaniv and Souroujon [6] propose an incremental version of this method that significantly improves its performance.

Friedman et al. [8] propose the Multivariate Information Bottleneck (MIB) framework: they consider clustering instances of a set of variables $\mathbf{X} = (X_1, \ldots, X_n)$ into a set of clustering systems $\tilde{\mathbf{X}} = (\tilde{X}_1, \ldots, \tilde{X}_n)$. After generalizing the standard bivariate Mutual Information $I(X;Y)$ to an n-variate Multi-Information $\mathcal{I}(\mathbf{X})$, Friedman et al. reformulate the Information Bottleneck principle as computing

$$\arg\max_{\tilde{\mathbf{X}}} \mathcal{I}^{G_{out}} - \beta \mathcal{I}^{G_{in}}, \tag{7}$$

where G_{in} and G_{out} are graphical models over $(\boldsymbol{X}, \tilde{\boldsymbol{X}})$ that best describe the dependencies between $\boldsymbol{X}$ and $\tilde{\boldsymbol{X}}$ in the input space and in the output space respectively. The double clustering problem thus becomes a partial case of MIB and can be derived as

$$\arg\max_{\tilde{X},\tilde{Y}} I(\tilde{X};\tilde{Y}) - \beta \left(I(\tilde{X};X) + I(\tilde{Y};Y) \right), \tag{8}$$

where $I(\tilde{X};X)$ and $I(\tilde{Y};Y)$ are the compression constraints.

3.2.2 Motivation

In the hard clustering variation of the IB method we set the Lagrange multiplier β to zero (see, e.g., [19]). Since we cannot just omit the compression constraints this way, a decent substitute would be to fix the number of (hard) clusters. The double clustering objective is then derived from Equation (8) as

$$\arg\max_{\tilde{X},\tilde{Y}} I(\tilde{X};\tilde{Y}), \text{ subject to } |\tilde{X}| = N_{\tilde{X}}, |\tilde{Y}| = N_{\tilde{Y}}, \tag{9}$$

where $|\tilde{X}|$ and $|\tilde{Y}|$ are sizes of the clustering systems $\tilde{X}$ and $\tilde{Y}$ respectively.

Since determining the "good" number of clusters is a hard problem, we cannot a priori be satisfied with fixed sizes $N_{\tilde{X}}$ and $N_{\tilde{Y}}$. Our intention is to explore different possibilities while employing the hierarchical structure of the clusters. At least two frameworks are ready for this task: agglomerative (bottom-up) and conglomerative (top-down) clustering.

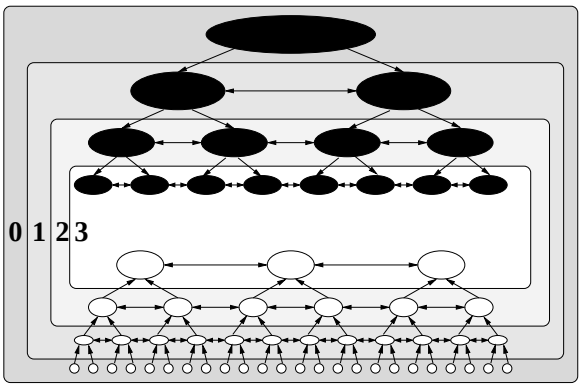

Figure 2: A/CDC procedure. At each iteration black clusters are split and then white clusters are merged.

We basically have three possibilities for performing the double clustering: we can use a top-down clustering scheme for both, we can cluster both by a bottom-up scheme, or we can apply a top-down scheme to one of the two clustering systems, while applying a bottom-up scheme to another one. Two top-down schemes are clearly a bad choice, because in the top-down scheme we start with one cluster that contains all the instances, and if both systems start with one cluster, then conditioning one on the other will lead to a completely random split. Two bottom-up schemes are also a bad choice, because of the computational issues: at the initial stages the two clustering systems are so large that the calculation of the Mutual Information $I(\tilde{X};\tilde{Y})$ can be infeasible.

We are left with top-down clustering in one system and bottom-up clustering in the other. In this case, iterative splits and merges (when one clustering system is conditioned in another) cause the effect that the two clustering systems "bootstrap" each other. Thus, the A/CDC method is the simultaneous clustering of X by a top-down scheme and Y by a bottom-up scheme, while applying the objective function from Equation 9. Figure 2 visualizes the A/CDC procedure.

3.2.3 Overview of algorithm

Following El-Yaniv and Souroujon [6], we break Equation (9) down to two parts:

$$\arg\max_{\tilde{X}} I(\tilde{X};\tilde{Y}), \quad \arg\max_{\tilde{Y}} I(\tilde{X};\tilde{Y}) \tag{10}$$

At each iteration of our algorithm we attempt to first build the best clustering system $\tilde{X}$ and then build the best clustering system $\tilde{Y}$.

We initiate the two clustering systems with one cluster $\tilde{x}$ that contains all data points x, and one data point y_i per each cluster $\tilde{y}_i$. We then calculate the initial Mutual Information $I(\tilde{X};\tilde{Y})$. At each iteration of the algorithm, we perform four operations:

1. **Split step.** We split each cluster $\tilde{x}_i$ uniformly at random to two equally sized parts.

2. **Sequential pass.** We utilize the sequential IB algorithm proposed by Slonim et al. [18]: we pick each data point x_j out of its cluster and place it sequentially

into each one of the other clusters, while attempting to maximize $I(\tilde{X};\tilde{Y})$. We finally place the data point x_j into a cluster $\tilde{x}_i$ such that $I(\tilde{X};\tilde{Y})$ is maximal. We perform this procedure twice in order to closer approach the local maximum of our objective.

3. **Merge step.** We uniformly at random select each cluster $\tilde{y}_i$, and find its best mate while applying a criterion for minimizing Bayes classification error that was proposed in [19].

4. **Another sequential pass.** We perform the same sequential pass as in Step 2 over all data points y_j.

Following Slonim et al. [18], in order to get closer to the global maximum of our objective function, at each iteration we perform a number of random restarts of Steps 1-2 and then of Steps 3-4. We also efficiently cache slices of the Mutual Information $I(\tilde{X};\tilde{Y})$ so that it should not be entirely recalculated during the sequential passes. The computational complexity of our algorithm is $O(N_x N_y \log N_y)$, where N_x and N_y are sizes of X and Y respectively.

In the case of Web appearance disambiguation, we use the top-down scheme for clustering words and the bottom-up scheme for clustering documents. We continue the process until we have three document clusters (one of which is then chosen to be the class of relevant pages).

3.3 LS+A/CDC Hybrid Model

Since in both solutions of the Web appearance disambiguation problem (Link Structure method and the A/CDC method) we build one group of relevant Web pages, we can attempt to overlap the groups built by the two methods. At one of the iterations of the A/CDC clustering we choose the most interconnected cluster C^* of the size that is roughly correspondent to the size of the central cluster C_0. Then we compose a new central cluster C_0^* by uniting all the connected components that overlap with C^*:

$$C_0^* = \bigcup_{C_i \cap C^* \neq \emptyset,\ i=0..M} C_i \tag{11}$$

After that, our discrimination function f is very similar to the discrimination function from Equation (1):

$$f(d,h|\mathcal{M}(\mathcal{K})) = \begin{cases} 1, & \text{if } d \in C_i:\ \|C_i - C_0^*\| < \delta,\ i = 0..M \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

This method gives us a larger but still clean central cluster which leads to more accurate choice of clusters within the δ radius from C_0^*.

4. DATASET

For evaluation of our methods, we have gathered and labeled a dataset of 1085 Web pages. In this section we describe the dataset and provide some interesting insights into its structure.

In a collaborative effort to create publicly available email datasets, participants in CALO project [14] are encouraged to collect and folder their correspondence on CALO-related topics. From the Feb 2, 2004 snapshot of this data, we selected one folder from Melinda Gervasio's email directory and extracted 12 person names that appeared in headers of messages found in this folder. The names are primarily of SRI employees and professors from different universities. All of the individuals are likely to be present on the Web.

Personal name	Position	Num of pages	Num of categories	Num of relevant pages
Adam Cheyer	SRI Manag	97	2	96
William Cohen	CMU Prof	88	10	6
Steve Hardt	SRI Eng	81	6	64
David Israel	SRI Manag	92	19	20
Leslie Pack Kaelbling	MIT Prof	89	2	88
Bill Mark	SRI Manag	94	8	11
Andrew McCallum	UMass Prof	94	16	54
Tom Mitchell	CMU Prof	92	37	15
David Mulford	Stanford Undergrad	94	13	1
Andrew Ng	Stanf Prof	87	29	32
Fernando Pereira	UPenn Prof	88	19	32
Lynn Voss	SRI Eng	89	26	1
	OVERALL:	1085	187	420

Table 1: Statistics of our dataset. Categories are different namesakes or *other* in case the page does not refer to any of the namesakes.

These 12 names (taken in quotation marks) were then issued as queries to Google and for each query the first 100 pages were retrieved. We manually filtered the pages, removing pages in non-textual formats, HTTPD error pages and empty pages. We labeled the remaining pages by the occupation of the individuals whose name appeared in the query. In 10 out of 12 cases, the names were heavily ambiguous, thus pages representing 187 different people were retrieved given the 12 names of people in Melinda's social network. In some cases, it was difficult to decide to which of the namesakes the page refers. To determine this, we often performed manual Web investigations. Table 1 shows some statistics of the dataset.

Finally, all the pages were cleaned of their HTML markup and scripts. All the URLs mentioned in the pages were extracted and placed at the end of each page, together with the URL of the page itself. The dataset is publicly available at http://www.cs.umass.edu/∼ronb.

The most ambiguous personal name among the twelve is Tom Mitchell. Although the CMU Professor's pages are prevalent over all the others, 37 different Tom Mitchells can be distinguished in the 100 first Google hits, including professors in different fields, musicians, executive managers, an astrologist, a hacker and a rabbi. Two personal names out of the 12, Adam Cheyer and Leslie Pack Kaelbling, seem to be unique in the Internet. However, for either of them, one page was retrieved that did not contain any part of their names. These two pages were put into respective categories *other*. Two other people, David Mulford and Lynn Voss, seem to have very little Web presence. Only one page out of the 100 was related to any of the two. William Cohen's and David Mulford's namesakes are well known politicians: the former Secretary of Defense William S. Cohen and the current US Ambassador to India David C. Mulford. Naturally,

Method	Precision	Recall	F-measure
Agglomerative	61.7	53.3	57.2
Link Structure	84.2	71.8	77.5
A/CDC	87.3 ± 1.7	71.3 ± 2.5	78.4 ± 0.9
LS+A/CDC	86.9	74.5	80.3

Table 2: Web appearance disambiguation results. A/CDC results are averaged over 4 random restarts.

the distributions of Cohen's and Mulford's pages are heavily biased toward the politicians who are well represented on the Web.

An interesting phenomenon is observed for the names David Israel and Bill Mark. Many of pages that responded to these queries only accidently contain the two words adjacent to each other: Bill Mark's pages often refer to mark-ups of certain bills, or just list people's first names (e.g. "Thanks Bill, Mark!"), while some of David Israel's pages discuss Israeli history and King David. None of these pages were removed from the dataset, despite the fact that they are clearly unrelated to a particular living person.

A real challenge for any Web presence finding system is the pages of Bill Mark and Fernando Pereira. Both researchers have namesakes who are also researchers in Computer Science: another Bill Mark is a UTexas Professor, while another Fernando Pereira is a Professor at Instituto Superior Técnico in Portugal. We term these pairs "doubles". To separate them is an especially difficult task. The opposite problem occurs with Steve Hardt: he appears on the Web not only as an SRI engineer, but also as a creator of an online game. We ourselves are actually unsure whether this is one person or two different people, but most likely this is one person.

5. RESULTS AND DISCUSSION

Given the class of relevant documents obtained by one of our models, our evaluation method computes precision and recall of the class with respect to the true labels in our dataset. We then compute the F-measure by averaging precision and recall. Our goal is to maximize the F-measure; however, we also consider separate precision and recall measures, because various real-world scenarios may prefer one over another.

As our baseline method, we implemented greedy agglomerative clustering (as applied in the related work [2, 13, 7]), based on the cosine similarity measure between clusters and the augmented *tfidf* weighting function from Equation (3). We did not measure interconnectedness of the clusters, we simply chose the cluster whose F-measure was the highest among all the clusters. The motivation for this choice was that we would like to show that our methods overcome the best possible results of the baseline method.

The summary of the results is in Table 2. As it can be seen from the table, the results of the three proposed methods are quite close to each other, while the hybrid method nicely improves the recall (and then the F-measure). The relatively high deviation in precision and recall of the A/CDC method is caused by the fact that it never ends up with clusters of the exactly same size. Interestingly, this almost does not affect the F-measure: the precision trades off quite well against the recall.

Name	Found correct	Not found	Found wrong
Adam Cheyer	62	34	0
William Cohen	6	0	4
Steve Hardt	16	48	2
David Israel	19	1	4
Leslie Pack Kaelbling	84	4	1
Bill Mark	6	5	9
Andrew McCallum	54	0	2
Tom Mitchell	14	1	5
David Mulford	1	0	0
Andrew Ng	30	2	6
Fernando Pereira	21	11	14
Lynn Voss	0	1	0
OVERALL:	313	107	47

Table 3: Results by person of the LS+A/CDC hybrid model.

Table 3 collates the results by person, as achieved by the hybrid model. For quite a few people both precision and recall are amazingly high, e.g. for David Israel, Leslie Pack Kaelbling, Andrew McCallum, Andrew Ng. It is also noticeable that the only relevant page of David Mulford (the Stanford student) is found. As could be anticipated, the worst precision is for Bill Mark and and Fernando Pereira, because both of them have "doubles". However, only 9 of 23 pages that refer to Bill Mark the UTexas Professor appear in the category of relevant pages. The worst recall is for Steve Hardt and Adam Cheyer. This can be easily explained for Steve: most of his pages refer to an online game he created—relevance of these pages would be too difficult to determine. As for Adam, the low result is a bit surprising, but it still makes sense: Adam's name often appears in an industrial context, while the language of most correctly-found pages is purely academic—many of Adam's pages fall too far from the central cluster. Unfortunately, the single relevant page about Lynn Voss was not found, probably for the same reason: it uses an industrial vocabulary.

The problem of disambiguating the "doubles"—the two Bill Marks and two Fernando Pereiras who all work in Computer Science—can in fact be handled within the A/CDC framework. At some intermediate stages during the course of the A/CDC algorithm the most interconnected cluster is relatively small but extremely clean. Figure 3 shows the precision/recall curve for one run of the A/CDC algorithm. It can be seen in the graph that when the recall of the relevant cluster is around 45% (there are five clusters overall), the precision is very high (above 98%).[3] This cluster contains two pages of Bill Mark the SRI Manager and none of the pages of Bill Mark the UTexas Professor; it also contains 15 pages of Fernando Pereira the UPenn Professor and only one page of Fernando Pereira the Professor of Instituto Superior Técnico.

This result shows that when our algorithm is stopped with 5, 9 or 17 clusters, rather than with three clusters, its performance is still very reasonable, at least in terms of precision. Constructing clustering systems with all possible granularity levels is an important feature of the A/CDC algorithm.

[3]Notably, when the recall is around 15% (17 clusters overall), we obtain 100% precision.

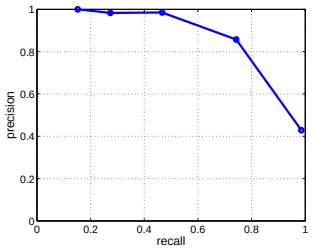

Figure 3: Precision/recall curve of the A/CDC algorithm. Points correspond to consequent iterations of the algorithm (merges of Web page clusters).

Another interesting result is that our methods can also be applied to the problem of homepage finding. Among the 12 people in our dataset ten have homepages (David Mulford and Lynn Voss do not maintain homepages), and nine of the ten homepages are inside the class of relevant documents found by the LS+A/CDC hybrid method. The only homepage the system does not find is the homepage of Steve Hardt.

6. CONCLUSIONS AND FUTURE WORK

This paper is the first attempt to approach the problem of finding Web appearances of a group of people. We have proposed two relatively straightforward statistical methods for solving this problem. Both methods are purely unsupervised—they involve minimum of prior knowledge about the people. Essentially, only the affiliation of a person with the group is all the information required.

For evaluation purposes we built a large annotated dataset that is publicly available to the scientific community. Both methods demonstrate high performance on this dataset. The methods are general enough to allow large variety of implementations and extensions.

We are now working on more sophisticated probabilistic models for solving this problem that would capture the relational structure of the class of relevant pages. For example, pages that are retrieved by queries with *pairs* of names can significantly enrich the model. The problem of Web appearance disambiguation is novel and poses a lot of exciting challenges.

7. ACKNOWLEDGMENTS

We thank Ran El-Yaniv for many fruitful discussions. This work was supported in part by the Center for Intelligent Information Retrieval and in part by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under contract number NBCHD030010. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] V. N. Anh and A. Moffat. Homepage finding and topic distillation using a common retrieval strategy. In *Proceedings of TREC-11*, 2002.

[2] A. Bagga and B. Baldwin. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of COLING-ACL-17*, pages 79–85, 1998.

[3] R. Bekkerman, R. El-Yaniv, and A. McCallum. Multi-way distributional clustering via pairwise interactions. Submitted.

[4] R. Bekkerman, R. El-Yaniv, N. Tishby, and Y. Winter. On feature distributional clustering for text categorization. In *Proceedings of SIGIR-24*, pages 146–153, 2001.

[5] A. Culotta, R. Bekkerman, and A. McCallum. Extracting social networks and contact information from email and the web. In *Proceedings of CEAS-1*, 2004.

[6] R. El-Yaniv and O. Souroujon. Iterative double clustering for unsupervised and semi-supervised learning. In *Proceedings of NIPS-14*, 2002.

[7] M. B. Fleischman and E. Hovy. Multi-document person name resolution. In *Proceedings of ACL-42, Reference Resolution Workshop*, 2004.

[8] N. Friedman, O. Mosenzon, N. Slonim, and N. Tishby. Multivariate information bottleneck. In *Proceedings of UAI-17*, 2001.

[9] C. H. Gooi and J. Allan. Cross-document coreference on a large scale corpus. In *Proceedings of HLT/NAACL*, 2004.

[10] H. Han, L. Giles, H. Zha, C. Li, and K. Tsioutsiouliklis. Two supervised learning approaches for name disambiguation in author citations. In *Proceedings of JCDL-4*, 2004.

[11] T. Joachims. *Learning to Classify Text using Support Vector Machines*. Kluwer Academic Publishers, Dordrecht, NL, 2002.

[12] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML-18*, pages 282–289, 2001.

[13] G. S. Mann and D. Yarowsky. Unsupervised personal name disambiguation. In *Proceedings of CoNLL-7*, pages 33–40, 2003.

[14] W. Mark and R. Perrault. CALO: a cognitive agent that learns and organizes. https://www.calo.sri.com.

[15] A. McCallum and B. Wellner. Conditional models of identity uncertainty with application to noun coreference. In *Proceedings of NIPS-17*, 2005.

[16] V. Ng and C. Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of ACL-40*, 2002.

[17] J. Shakes, M. Langheinrich, and O. Etzioni. Dynamic reference sifting: a case study in the homepage domain. In *Proceedings of WWW-6*, 1997.

[18] N. Slonim, N. Friedman, and N. Tishby. Unsupervised document classification using sequential information maximization. In *Proceedings of SIGIR-25*, 2002.

[19] N. Slonim and N. Tishby. Agglomerative information bottleneck. In *Proceedings of NIPS-12*, 2000.

[20] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. In *Proceedings of SIGIR-23*, pages 208–215, 2000.

[21] N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method, 1999. Invited paper to the 37th annual Allerton Conference.