Collective Annotation of Wikipedia Entities in Web Text

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti
IIT Bombay
ganesh@cse.iitb.ac.in, soumen@cse.iitb.ac.in

ABSTRACT

To take the first step beyond keyword-based search toward entity-based search, suitable token spans ("spots") on documents must be identified as references to real-world entities from an entity catalog. Several systems have been proposed to link spots on Web pages to entities in Wikipedia. They are largely based on local compatibility between the text around the spot and textual metadata associated with the entity. Two recent systems exploit inter-label dependencies, but in limited ways. We propose a general collective disambiguation approach. Our premise is that coherent documents refer to entities from one or a few related topics or domains. We give formulations for the trade-off between local spot-to-entity compatibility and measures of global coherence between entities. Optimizing the overall entity assignment is NP-hard. We investigate practical solutions based on local hill-climbing, rounding integer linear programs, and pre-clustering entities followed by local optimization within clusters. In experiments involving over a hundred manually-annotated Web pages and tens of thousands of spots, our approaches significantly outperform recently-proposed algorithms.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Information Systems – *Information Storage And Retrieval*
General Terms: Algorithms, Experimentation
Keywords: Entity annotation/disambiguation, Wikipedia, collective inference

1. INTRODUCTION

A critical step in bridging between unstructured Web text and semistructured search and mining applications is to identify textual references (called "spots") to named entities and annotate the spots with unambiguous entity IDs (called "labels") from a catalog. These entity ID annotations enable powerful join operations that can combine information across pages and sites. Named entity recognition and tagging have seen widespread success [17]. Here we are concerned with the second step: entity disambiguation from a given catalog, such as Wikipedia. (The availability of a catalog makes this a supervised setting, unlike unsupervised coreference resolution.)

1.1 Entity catalogs

The success of semantic annotation is greatly determined by widespread adoption of the entity catalog. For common English words, WordNet [14] provides an authoritative lexical network designed by linguists, and widely used for disambiguation of common words [1]. CYC and OpenCYC [12] are partly commercial efforts to maintain entity catalogs, rules and reasoning engines. To understand and maintain TAP [8], WordNet, or OpenCYC, substantial training is needed in knowledge representation and linguistics.

In contrast, the "Web 2.0" trend is to throw open tagging and cataloging of knowledge to the masses. Wikipedia is a stunning example of the success of this strategy: it has over 0.34 million categories and over 2.6 million cataloged entities, and keeps up with world events on an hourly or daily basis. The flip side is that Wikipedia lacks the rigorous "knowledge base" quality of TAP or OpenCYC. There is little by way of schema, quality of authorship is diverse, and the category hierarchy is haphazard. The challenge of Web mining systems is to harness the chaotic "wisdom" of the crowds into relatively clean knowledge.

1.2 Prior work and limitations

Most existing systems annotate only *salient* entity references. In some prototypes, only entities of specific recognized types (most often people and locations) are disambiguated. The goal is to emulate Wikipedia's restrained, informative, editorial links on ordinary Web pages.

SemTag. The first Web-scale entity disambiguation system was SemTag [5]. SemTag annotated about 250 million Web pages with IDs from the Stanford TAP entity catalog [8]. The basic technique was to compare the surrounding context of a spot s with text metadata associated with candidate entity γ in TAP. SemTag preferred high precision over recall, proposing only about 450 million annotations, i.e., fewer than two annotations per page on average.

Wikify!. Wikify! [13] has two components. The first, keyword extraction, decides if a phrase should be linked to Wikipedia. This is based on how often a word or phrase is found to be in the anchor text of some link internal to Wikipedia. The second step is disambiguation. Wikify!, too, is conservative in flagging keywords, so much so that even *random* disambiguation results in an F_1 score of 0.82. Suppose Wikify! is considering linking spot s to entity γ. Wikipedia's page describing γ is explicitly referred from other Wikipedia pages. The context of these known citations is compared with the context of s to decide on a compatibility score. This may be regarded as generalizing SemTag, where known references to γ form part of the metadata of γ. Bunescu and Pasca [3] further improved the compatibility function using SVMs with tree kernels. However, none of these systems attempt collective disambiguation across spots.

M&W. A limited form of collective disambiguation proposed by Milne and Witten [15] yields considerable improvement beyond Wikify!. M&W propose a *relatedness* score $r(\gamma, \gamma')$ between two entities. From the set of all spots S_0, they identify the subset $S_!$ of so-called *context* spots that can refer to exactly one entity each (let this entity set be $\Gamma_!$). They define a notion of *coherence* of a context spot

KDD'09, June 28–July 1, 2009, Paris, France.

$\gamma \in \Gamma_!$ based on its relatedness to other context spots. For an ambiguous spot $s \notin S_!$, the score of a candidate entity $\gamma \notin \Gamma_!$ is strongly influenced by its mention-independent prior probability $\Pr_0(\gamma|s)$, its relatedness to context entities on the page, their coherence, and a measure of overall quality of context entities. M&W also propose a *link detector* (a function similar to keyword extraction in Wikify!) that, like SemTag and Wikify!, sacrifices recall for high precision. For the spots picked by M&W for labeling, even random disambiguation achieves an F_1 score of 0.53.

Cucerzan's algorithm. To our knowledge, Cucerzan [4] was the first to recognize general interdependence between entity labels in the context of Wikipedia annotations. He represents each entity γ as a high-dimensional feature vector $g(\gamma)$, and expressed $r(\gamma, \gamma')$ as the inner product (or cosine, if $\|g(\gamma)\|$ are normalized) $g(\gamma)^\top g(\gamma')$, also written as $g(\gamma) \cdot g(\gamma')$. Let Γ_0 be all possible entity disambiguations for all spots on a page. He precomputes the average vector $g(\Gamma_0) = \sum_{\gamma \in \Gamma_0} g(\gamma)$. The score of candidate γ for spot s depends on two factors. The first, like SemTag or Wikify!, is a local context compatibility score. The second is $g(\gamma)^\top g(\Gamma_0 \setminus \{\gamma\})$, reminiscent of leave-one-out cross validation. Cucerzan also annotates very sparingly: only about 4.5% of all tokens are annotated.

A problem with this approach is that Γ_0 is contaminated with all possible disambiguations of all spots, so this check for "agreement with the majority" may be misleading. Note that both M&W and Cucerzan avoid direct joint optimization of all spot labels, which is precisely what we undertake.

The above line of work has some similarity to identifying mentions of entities in databases (e.g. product catalogs) amidst unstructured text (e.g., blogs) [2], but, in such applications, the "entity catalog" is a clean relational database, and, to our knowledge, no collective labeling is employed.

1.3 Our goals and contributions

Our goal in this paper is *aggressive open-domain annotation* of Web pages with entity IDs from an entity catalog such as Wikipedia. We contrast this with a more restricted disambiguation of entities which achieves high precision by sacrificing recall. The central purpose of our annotation is not direct human consumption, but downstream indexing, search and mining.

For example, we may gather from one page that m is a mathematician, and from another, that m plays the violin. Such data can be aggregated to explore whether scientists tend to play music significantly more or less often than other people mentioned on the Web.

Our guiding premise is that documents largely refer to topically coherent entities, and this "coherence prior" can be exploited for disambiguation. While *Michael Jordan* and *Stuart Russell* can refer to seven (basketball player, footballer, actor, machine learning researcher, etc.) and three (politician, AI researcher, DJ) persons respectively in Wikipedia (as of early 2009), a page where both *Michael Jordan* and *Stuart Russell* are mentioned is almost certainly about computer science, disambiguating them completely.

We propose a collective optimization problem that precisely models the combination of evidence from local spot-to-entity compatibility ("node potential") and global page-level topical coherence ("clique potential") of the entities chosen to disambiguate all spots. Our optimization is equivalent to searching for the maximum probability annotation configuration (inference) in a probabilistic graphical model where each page is a clique. Inference is NP-hard. We propose practical and effective heuristics based on local hill-climbing and linear program relaxations. Our framework also applies to word sense disambiguation [1], and therefore, may be of independent interest.

We describe our experiments with two data sets. Cucerzan's ground-truth data [4] has annotations only for persons, places and organization and is limited to only 700 spots on non-Wikipedia data. While SemTag was run on 264 million pages and produced 434 million annotations, human judgement was collected on only about 1300 spot labelings (and this data is not publicly available). We built a browser-based annotation UI that six volunteers used to collect over 19,000 spot annotations on more than 100 pages; largest among known prior work[1]. Experiments show that we can significantly push the recall envelope without hurting precision. Our trained node potential alone can improve F_1 accuracy considerably compared to all of Wikify!, Cucerzan and M&W's algorithms. Taking clique potentials into consideration using LP rounding or greedy hill-climbing gives further accuracy gains.

S_0	All candidate spots in a Web page
$S \subseteq S_0$	Arbitrary set of spots
$s \in S$	One spot, including surrounding context
Γ_s	Candidate entity labels for spot s
Γ_0	$\bigcup_{s \in S_0} \Gamma_s$, all candidate labels for page
$\Gamma \subseteq \Gamma_0$	An arbitrary set of entity labels
$\gamma \in \Gamma$	An entity label value, here, a Wikipedia URN
NA	The "no attachment" decision
ρ_{NA}	The "reward" for setting $y_s = \text{NA}$
$N_0 \subseteq S_0$	Spots assigned NA
$A_0 = S_0 \setminus N_0$	Spots assigned some $\gamma \neq \text{NA}$
S_γ	Spots that can disambiguate to γ
S_Γ	$\bigcup_{\gamma \in \Gamma} S_\gamma$

Figure 1: Notation.

2. NOTATION AND PRELIMINARIES

2.1 Spots and entity labels

Because we process one page at a time, we will elide the page in our notation. A *spot* s is a (short) token (sequence) that is potentially a direct reference to an entity in Wikipedia. We do not consider indirect references like pronouns, but do aim to resolve imperfect matches such as *Michael* for *Michael I. Jordan*. The *context* of s is the text in a suitable window around s. An *entity* is expressed as a URN in Wikipedia, and denoted γ. NA is a special label denoting "no attachment", i.e. an algorithm can avoid labeling a spot to increase precision at the cost of recall. $\rho_{\text{NA}} \geq 0$ is a tuned parameter to guide this tradeoff. See Figure 1.

2.2 Compatibility feature vector $f_s(y_s)$

y_s is a *variable* denoting the entity label, taking a *value* from $\Gamma_0 \cup \{\text{NA}\}$. y is the vector of all spot labels. $f_s(\gamma) \in \mathbb{R}^d$ is a feature vector whose elements express various measures of compatibility between s and γ. The *context* of a spot is a bag of words collected from a suitable window around the candidate entity reference.

[1]The data is in the public domain, see `http://soumen.cse.iitb.ac.in/~soumen/doc/CSAW/` or `http://www.cse.iitb.ac.in/~soumen/doc/CSAW/`

Wikipedia is preprocessed so that each page corresponding to an entity γ is represented by four *fields*.

- Text from the first descriptive paragraph of γ.
- Text from the whole page for γ.
- Anchor text within Wikipedia for γ.
- Anchor text and five tokens around it.

Each field is turned into a bag (multiset) of words. Three text match scores are computed between a field of γ and s:

- Dot-product between word count vectors.
- Cosine similarity in TFIDF vector space.
- Jaccard similarity between word sets.

So in all, we get $4 \times 3 = 12$ features.

Sense probability prior. Some $\gamma \in \Gamma_s$ are very obscure and rare; i.e., $\Pr(\gamma|s)$ is very low. E.g., *Intel* is (also) a fictional cartel in a 1961 BBC TV serial, but this sense is much rarer than the semiconductor giant. We can easily count from intra-Wikipedia links the fraction of times a link with *Intel* in the anchor text points to every sense of *Intel*, and use this as a **prior** estimate of $\Pr_0(\gamma|s)$. The last element of $f_s(\gamma)$ is $\log \Pr_0(\gamma|s)$ (the log is explained below). This feature is somewhat different from local compatibility features, so we will often study its effect separately.

2.3 Compatibility score (node potential)

The local compatibility score between s and γ is modeled as $w^\top f_s(\gamma)$ where $w \in \mathbb{R}^d$ is a model vector. For a locally optimal choice, we would pick $\arg\max_{\gamma \in \Gamma_s} w^\top f_s(\gamma)$ as the label for s. If we had to normalize this to a probability, we would use a logistic model $\Pr(\gamma|s) = \exp(w^\top f_s(\gamma))/Z_s$ where $Z_s = \sum_{\gamma' \in \Gamma_s} \exp(w^\top f_s(\gamma'))$, is the partition function (hence the log in the sense probability feature above). We call $\exp(w^\top f_s(\cdot))$ the *node potential* of s, using graphical model [9] terminology.

We train w using a max-margin technique. Given ground truth assignment $\gamma_s^* \in \Gamma_s$, we want $w^\top f_s(\gamma_s^*)$ to be larger than any other $w^\top f_s(\gamma)$, with a margin; this gives us the usual SVM linear constraints (for spots with $\gamma_s^* \neq$ NA only):

$$\forall s, \forall \gamma \neq \gamma_s^* \in \Gamma_s : \quad w^\top f_s(\gamma_s^*) - w^\top f_s(\gamma) \geq 1 - \xi_s$$

and we minimize over $\xi \geq \vec{0}$ and w the objective $\|w\|_2^2 + C \sum_s \xi_s$ where C is the usual balancing parameter. Our training data had tens of thousands of spots, and the Γ_s has more than 10 elements, leading to $\sim 10^6$ constraints. Rather than use $\|w\|_2^2$ and a QP solver, we used $\|w\|_1$ which let us use a more scalable LP solver (Mosek). More notation is summarized in Figure 2.

2.4 Label embedding and relatedness

The last and most important piece in our model is an embedding of labels γ in a suitable (usually high-dimensional) feature space: $g : \Gamma_0 \to \mathbb{R}^c$. This embedding is used to define relatedness between two entities.

2.4.1 Cucerzan's category-based $g(\gamma)$ and $r(\gamma, \gamma')$

The Wikipedia page for γ lists a set of *categories* that γ belongs to. E.g., $\gamma =$ `Michael_jordan` is a *Sportspeople of multiple sports* while $\gamma =$ `Michael_I._Jordan` is a *Machine learning researcher*. If there are c categories in Wikipedia, the categories that γ belongs to can be represented by a c-long bit vector, which is designated as $g(\gamma)$. Cucerzan defined the relatedness between two entities γ, γ' as

$$r(\gamma, \gamma') = \frac{g(\gamma)^\top g(\gamma')}{\sqrt{g(\gamma)^\top g(\gamma)}\sqrt{g(\gamma')^\top g(\gamma')}},$$

a standard cosine measure.

Wikipedia's categorization is organic and uncontrolled: $\gamma =$ `Michael_I._Jordan` also belongs to categories called *Living people* and *Year of birth missing*, which are not topical. We tried to mitigate this with various weighting schemes, but Cucerzan's algorithm nevertheless performed worse than each of our algorithms, which used the relatedness definition described next.

y_s	Variable denoting label assigned to spot s
y	Vector of all label assignment for page
y^*	Vector of ground-truth labels
Γ^*	Set of ground-truth labels in y^*
$f_s(y_s) \in \mathbb{R}^d$	Compatibility feature vector between s and y_s
$w \in \mathbb{R}^d$	Compatibility weight vector
$\mathrm{NP}_s(y_s)$	$\exp(w^\top f_s(y_s))$, spot-to-label compatibility
$g(\gamma) \in \mathbb{R}^c$	An embedding of γ in suitable space (see text)
$r(\gamma, \gamma')$	Topical relatedness between γ, γ'
$\mathrm{CP}(y)$	Topical coherence among all labels for page

Figure 2: More notation.

2.4.2 M&W's inlink-based $g(\gamma)$ and $r(\gamma, \gamma')$

Cocitation has been used to detect relatedness for a long time [11]. Milne and Witten [15] represented $g(\gamma)$ as the set of Wikipedia pages that link to γ, with size $|g(\gamma)|$. Let c be the total number of Wikipedia pages. M&W defined a relatedness measure (larger value implies more related) as:

$$r(\gamma, \gamma') = \frac{\log |g(\gamma) \cap g(\gamma')| - \log \max\{|g(\gamma)|, |g(\gamma')|\}}{\log c - \log \min\{|g(\gamma)|, |g(\gamma')|\}}$$

The numerator is a slight variation on Jaccard similarity, and the denominator is inversely related to $\min\{|g(\gamma)|, |g(\gamma')|\}$. Unless otherwise specified this is the measure we use.

2.5 Range compression

To robustly balance between local and global signals having diverse dynamic ranges, we apply a *range compressor function* $R(\cdot)$ to all elements of vectors $f_s(\cdot)$ and $g(\cdot)$. Specifically,

$$R(t) = \begin{cases} \log(1+t), & t \geq 0 \\ -\log(1-t), & t < 0. \end{cases}$$

This limits the numeric output range without clipping it at any value. Henceforth, when we write "$f_s(y_s)$" or "$g(\gamma)^\top g(\gamma')$", we will mean $R(f_s(y_s))$ or $R(g(\gamma))^\top R(g(\gamma'))$ respectively. To keep notation uncluttered, we will hide R and this preprocessing step. Note that $R(\cdot)$ is not applied to ρ_{NA}, the reward for assigning label NA to a spot.

3. THE DOMINANT TOPIC MODEL

We now describe our main model and inference approaches. The key is to define, over and above node potentials, a collective score based on pairwise topical coherence of all γs used for labeling.

3.1 Coherence score (clique potential)

For the moment, disallow $y_s =$ NA. Consider Figure 3. If γ, γ' are used as labels for s, s', their agreement is defined as $r(\gamma, \gamma')$. For the whole page, the overall agreement is

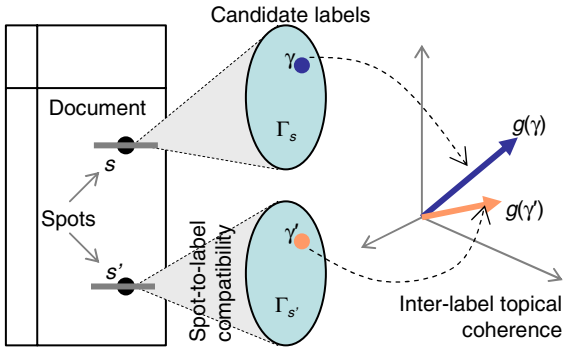

Figure 3: Labels $\gamma \in \Gamma_s, \gamma' \in \Gamma_{s'}$ have to be chosen for spots s, s' to maximize a combination of spot-to-label compatibility scores $\mathrm{NP}_s(\gamma), \mathrm{NP}_{s'}(\gamma')$ as well as topical similarity between γ and γ', say, $g(\gamma)^\top g(\gamma')$.

aggregated as $\sum_{s \neq s' \in S_0} r(y_s, y_{s'})$. In keeping with standard graphical models style [9], we can turn this into a clique potential

$$\mathrm{CP}(y) = \exp\left(\textstyle\sum_{s \neq s' \in S_0} r(y_s, y_{s'})\right), \tag{1}$$

and the overall probability of a label assignment y is written as $\Pr(y) = (1/Z)\,\mathrm{CP}(y) \prod_{s \in S_0} \mathrm{NP}_s(y_s)$, where $Z = \sum_{y'} \mathrm{CP}(y') \prod_{s \in S_0} \mathrm{NP}_s(y'_s)$ is a scale factor that makes the probabilities add up to 1 over all possible y.

Evaluating Z is difficult because an exponential number of terms need to be added up. For predicting the most likely label vector, finding Z is not needed; we just need

$$\begin{aligned}
\arg\max_y \Pr(y) &= \arg\max_y \mathrm{CP}(y) \prod_s \mathrm{NP}_s(y_s) \\
&= \arg\max_y \log \mathrm{CP}(y) + \sum_s \log \mathrm{NP}_s(y_s) \\
&= \arg\max_y \sum_{s \neq s' \in S_0} r(y_s, y_{s'}) + \sum_{s \in S_0} w^\top f_s(y_s).
\end{aligned}$$

The two sums have different number of terms, which also vary from page to page. To be able to use a single consistent w across all pages, we need to scale the two parts to a compatible magnitude. So our objective, barring NAs, is

$$\frac{1}{\binom{|S_0|}{2}} \sum_{s \neq s' \in S_0} r(y_s, y_{s'}) + \frac{1}{|S_0|} \sum_{s \in S_0} w^\top f_s(y_s).$$

3.2 Recall-precision balance

Almost a third of spots in our ground truth data are marked "NA" by volunteers, meaning that no suitable entity was found in Wikipedia. This is a reality on the open-domain Web, and many systems [13, 3, 15] can back off from annotation (indeed, back off aggressively).

To implement a recall-precision balance, we use one tuned parameter $\rho_{\mathrm{NA}} \geq 0$, the reward for not assigning a spot any label. Let $N_0 \subseteq S_0$ be the spots assigned NA, and $A_0 = S_0 \setminus N_0$ the remaining spots. We thus get our final objective:

$$\max_y \frac{1}{|S_0|} \left(\sum_{s \in N_0} \rho_{\mathrm{NA}} + \sum_{s \in A_0} w^\top f_s(y_s) \right) \tag{NP}$$

$$+ \frac{1}{\binom{|S_0|}{2}} \sum_{s \neq s' \in A_0} r(y_s, y_{s'}) \tag{CP1}$$

The reader may demand that $\binom{|S_0|}{2}$ be replaced by $\binom{|A_0|}{2}$ in (CP1). This creates difficulty for at least one of our inference approaches, because A_0 depends on y and the resulting optimization can no longer be written as an integer linear program. One possible rationalization is that NA has zero topical coherence with any other label, including another instance of NA:

$$r(\mathrm{NA}, \cdot) = r(\cdot, \mathrm{NA}) = r(\mathrm{NA}, \mathrm{NA}) = 0; \tag{2}$$

therefore, the edge potential sum can be rewritten over $s \neq s' \in S_0$, not $s \neq s' \in A_0$, so that the $\binom{|S_0|}{2}$ denominator is acceptable.

A reasonable way to tune ρ_{NA} would be to first compute the typical value of $w^\top f_s(y_s)$ across all pages and spots in the training set, then sweep ρ_{NA} between $0.1\times$ to $10\times$ of that typical value. We use this approach in our experiments.

3.3 Complexity of inference

Figure 3, (NP) and (CP1) get to the heart of the collective disambiguation problem, so it is of interest to understand the complexity of inference.

Proposition 1. *Inference problem* $\max_y$ (NP) + (CP1) *is NP-hard, even when* $\rho_{\mathrm{NA}} = -\infty$ *and therefore* $A_0 = S_0$.

The reduction is from the maximal clique problem [7]. We also note that other natural definitions of CP do not make the problem easier.

Proposition 2. *The inference problem remains NP-hard with the following alternative definitions of* CP*:*

$$\mathrm{CP}(y) = \exp\left(-\sum_{i \neq j} \|g(y_i) - g(y_j)\|_2^2 \right) \tag{3}$$

$$\mathrm{CP}(y) = \exp\left(-\max_{i \neq j} \|g(y_i) - g(y_j)\|_\infty \right) \tag{4}$$

Hardness using (3) is shown using a reduction from exact cover by 3-sets [7]. Hardness using (4) is shown using a reduction from 3SAT. Proofs are omitted to save space.

3.4 LP rounding approach

Guided by approaches to Quadratic Assignment Problems (QAPs) [16] we can turn our optimization into a 0/1 integer linear program, and then relax it to an LP. First disallow $y_s = \mathrm{NA}$. The ILP is designed with up to $|\Gamma_0| + |\Gamma_0|^2$ variables

$$z_{s\gamma} = [\![\text{spot } s \text{ is assigned label } \gamma \in \Gamma_s]\!]$$
$$u_{\gamma\gamma'} = [\![\text{both } \gamma, \gamma' \text{ assigned to spots}]\!]$$

The node potential part is written as

$$\frac{1}{|S_0|} \sum_{s \in S_0} \sum_{\gamma \in \Gamma_s} z_{s\gamma} w^\top f_s(\gamma) \tag{NP$'$}$$

and the clique potential part is written as

$$\frac{1}{\binom{|S_0|}{2}} \sum_{s \neq s' \in S_0} \sum_{\gamma \in \Gamma_s, \gamma' \in \Gamma_{s'}} u_{\gamma\gamma'} r(\gamma, \gamma') \tag{CP1$'$}$$

where we assume (2). So the goal is to

$$\max_{\{z_{s\gamma}, u_{\gamma\gamma'}\}} (\mathrm{NP}') + (\mathrm{CP1}') \quad \text{s.t.}$$

$$\forall s, \gamma : z_{s\gamma} \in \{0,1\}, \quad \forall \gamma, \gamma' : u_{\gamma\gamma'} \in \{0,1\} \tag{5}$$

$$\forall s, \gamma, \gamma' : u_{\gamma\gamma'} \leq z_{s\gamma} \quad \text{and} \quad u_{\gamma\gamma'} \leq z_{s\gamma'} \tag{6}$$

$$\forall s : \textstyle\sum_\gamma z_{s\gamma} = 1. \tag{7}$$

Constraints (6) enforce what we need, because, if $z_{s\gamma} = z_{s\gamma'} = 1$, the objective will push $u_{\gamma\gamma'} = 1$. The formulation generalizes readily to the NA case using one more variable $z_{s\text{NA}}$ per spot, changing constraint (7) to

$$\forall s: \; z_{s\text{NA}} + \sum_{\gamma} z_{s\gamma} = 1$$

and adding a term $\frac{1}{|S_0|}\sum_{s\in S_0} \rho_{\text{NA}} z_{s\text{NA}}$ to the objective.

3.4.1 Integrality gap

The relaxed LPs replace constraints (5) with $0 \le z_{s\gamma} \le 1$ and $0 \le u_{\gamma\gamma'} \le 1$. The optimal LP objective will be an upper bound on the optimal ILP objective. To understand how loose the upper bound can be in the worst case, consider the following "butterfly graph" example. (Disallow NA using $\rho_{\text{NA}} = -\infty$.) There are two spots s_1, s_2, with $\Gamma_{s_1} = \{\gamma_1, \gamma_2\}$ and $\Gamma_{s_2} = \{\gamma_3, \gamma_4\}$. Assume all node potentials are zero, and all $r(\gamma, \gamma') = 1$. The optimal integral solution can have at most one $u_{\gamma\gamma'} = 1$, leading to an objective value of $1/\binom{2}{2} = 1$. The fractional solution will find it best to assign all $z_{s\gamma} = u_{\gamma,\gamma'} = 1/2$, with an objective of $4 \times 0.5 = 2$. The gap can be increased arbitrarily by increasing the bipartite clique size, i.e., $|\Gamma_s|$.

3.4.2 Rounding policy

In our experiments, we found about 70% of pages to give completely integral (hence, optimal) solutions. The obvious rounding strategy for fractional solutions is $\arg\max_{\gamma\in\Gamma_s\cup\text{NA}} z_{s\gamma}$. We found that this tended to label NA as some $\gamma \neq$ NA. Insisting that $z_{s\gamma} > 1/2$ was more reticent and gave slightly better F_1.

```
1: initialize some assignment y^(0)
2: for k = 1, 2, ... do
3:    select a small spot set S_Δ
4:    for each s ∈ S_Δ do
5:       find new γ that improves objective
6:       change y_s^(k−1) to y_s^(k) = γ greedily
7:    if objective could not be improved then
8:       return latest solution y^(k)
```

Figure 4: Dominant cluster hill-climbing (Hill1)

3.5 Hill-climbing approach

Another approach is to avoid math programming and deploy a direct greedy hill-climbing approach. Hill climbing has the advantage that it can be easily stopped and interpreted at any time, and may achieve acceptable accuracy faster than solving and rounding an LP. The generic template is shown in Figure 4. It remains to specify the initialization, and how to make label modifications.

3.5.1 Initialization

Some initializations suggest themselves:

- Initialize all $y_s =$ NA
- Initialize all $y_s \neq$ NA as per node potential alone, i.e., $\arg\max_{\gamma\in\Gamma_s} w^\top f_s(\gamma)$ (we use this option)

In our experiments we did not find significant differences between accuracies obtained using the above initializations.

3.5.2 Label updates

We tried perturbing sets S_Δ of sizes 1 and 2. The rationale for trying to perturb a pair of spots was that any single spot perturbation may appear unattractive while at a local optimum. However, $|S_\Delta| = 2$ was already too slow to improve upon LP speeds. So we concentrate on single moves. If the label of s is changed from γ_1 to γ_2, node score (NP) changes by

$$\frac{-\left\{\begin{array}{ll}\rho_{\text{NA}} & (\gamma_1 = \text{NA})\\ w^\top f_s(\gamma_1) & (\text{o.w.})\end{array}\right\} + \left\{\begin{array}{ll}\rho_{\text{NA}} & (\gamma_2 = \text{NA})\\ w^\top f_s(\gamma_2) & (\text{o.w.})\end{array}\right\}}{|S_0|}$$

and edge score (CP1) changes by

$$\frac{1}{\binom{|S_0|}{2}} \sum_{s'\neq s} \big(r(y_{s'}, \gamma_2) - r(y_{s'}, \gamma_1)\big)$$

4. EXPERIMENTS

4.1 Testbed

4.1.1 Preprocessing Wikipedia

We downloaded the August 2008 version of Wikipedia, and prepared a dictionary of entity IDs, their labels and mentions, as follows:

- A set of 5.15 million entity IDs, including titles, redirections, disambiguations, and category names was first collected from the dump.
- A subset of these entity IDs was filtered out. An entity ID was filtered out either if it was composed purely of verbs, adverbs, conjunctions or prepositions or if it conformed to certain lexical patterns (e.g., fewer than three characters). The former rules pruned about 15,000 entity IDs; the latter pruned about 16,100 (of a total of 2.6 million).
- To enable efficient lookups of entity labels at runtime, a trie (prefix tree) of the filtered Wikipedia entity IDs was constructed using the Webgraph[2] framework.
- Spots are identified by first tokenizing the document (based on punctuation and white space as delimiters) and then identifying token sequences that maximally match an entity ID in the trie. Consequently, any candidate spot that happens to be a substring of another candidate spot will be subsumed into the latter.
- For each identified spot indexed i, all entity IDs found to have the same surface forms are associated with the spot to yield a set Γ_i of its possible disambiguations.

4.1.2 Preparing corpora and annotations

Earlier work has used Wikipedia text itself as ground truth annotations. This is not suited to our aggressive recall target, so we looked for other data. SemTag collected only about 1300 manually labeled spots for quality checking, and these are not publicly available. Data used by Bunescu and Pasca [3] was not publicly available. Cucerzan's data [4] (which we abbreviate to 'CZ') is available and we do use it, but annotations are sparse and limited to a few entity types. Several URN labels in CZ data no longer exist in Wikipedia. Moreover, there is no NA annotation.

Therefore we undertook to build a ground truth collection (which we call "IITB") using a browser-based annotation system. Documents for manual annotation were collected from the links within homepages of popular sites belonging to a

[2] `http://webgraph.dsi.unimi.it/`

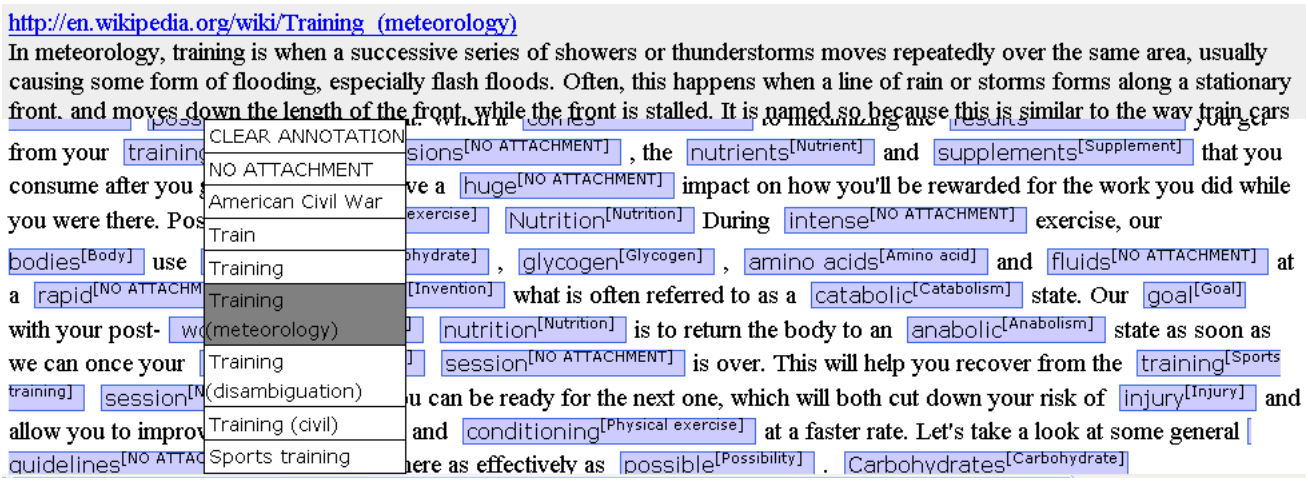

Figure 5: Browser-based annotation GUI. For each s, trainers choose γ_s^* from a pulldown menu showing Γ_s.

handful of domains that included sports, entertainment, science and technology, and health (sources: `http://news.google.com/` and `http://www.espnstar.com/`). Figure 6 summarizes some important corpus statistics. The annotations are available in the public domain. Both IITB and CZ data have high average ambiguity. CZ's is higher because the spots are limited to common person and place names. Obviously, random assignment would get very poor accuracy, unlike M&W.

4.1.3 Browser-based annotation GUI

CZ data came pre-annotated, but for the IITB corpus, we built a browser-based annotation tool. As illustrated in Figure 5, candidate spots are highlighted to differentiate between pending and already annotated spots. Clicking on a spot drops down a list of possible disambiguations. Hovering on a specific Wikipedia label shows an excerpt from the definition paragraph of the corresponding entity.

In the IITB data, we collected a total of about **19,000** annotations by 6 volunteers. Unlike in previous work, *volunteers were told to be as exhaustive as possible and tag all possible segments, even if to mark them as* NA. The number of distinct Wikipedia entities that were linked to was about 3,800. About 40% of the spots was labeled NA, highlighting the importance of backoffs. However, this also says that 60% of the spots *were* attached by volunteers, which by far exceeds the token rate of attachment in earlier work. While its absolute scale is impressive, SemTag produced only 434 million annotations from 264 million Web pages, or fewer than two per page. From Figure 6, we see that the CZ data identifies only about 15 spots per page. We thus highlight that *we are in a completely different recall regime.*

The annotation module allows each document to be tagged by two volunteers. Figure 7 summarizes some statistics on inter-annotator agreement. Clearly, a considerable number of disagreements are over NA vs. "not-NA".

4.1.4 Evaluation measures

Accuracy. A simple option would be to count the fraction of spots s (that have manually associated labels γ_s^*) which get assigned $y_s = \gamma_s^*$, over N_0^* and A_0^* alike. However, typical applications will be asymmetric in how they react to these labels. E.g., an indexing engine that incorporates object IDs will simply ignore NA labels. Therefore, we need to also focus on A_0^* separately.

Recall, precision, F_1. Suppose, in ground truth, the set of spots marked NA is N_0^*, and $A_0^* = S_0 \setminus N_0^*$ is the set of spots marked some label other than NA. We will be largely concerned about the precision, recall, and F_1 scores of spots in A_0^*. The fate of such a spot can be one of the following:

$A \rightarrow A$: Correctly labeled
$A \nrightarrow A$: Algorithm picks wrong label $\gamma \neq$ NA
$A \nrightarrow$ NA: Algorithm picks $\gamma =$ NA

$$precision = \frac{|\{A \rightarrow A\}|}{|\{A \rightarrow A\}| + |\{A \nrightarrow A\}| + |\{A \nrightarrow \text{NA}\}|}$$

$$recall = \frac{|\{A \rightarrow A\}|}{|A_0^*|}$$

Precision and recall are (macro-) averaged across documents and overall F_1 computed from average precision and recall. Note that the presence of NA makes these definitions different from what Cucerzan and M&W measured as spot labeling accuracy *after* spot detection.

All parameters were tuned using 2-fold cross validation.

4.2 Local NP optimization

As a first step, we are interested in evaluating the effect of training w, isolated from the influence of clique potentials. For this, we ran a very simple system that we will call LOCAL. LOCAL used the trained w to choose a label for each spot independent of others, without any collective information:

	IITB	CZ
Number of documents	107	19
Total number of spots	17,200	288
Spot per 100 tokens	30	4.48
Average ambiguity per Spot	5.3	18

Figure 6: Corpus statistics.

#Spots tagged by more than one person	1390
#NA among these spots	524
#Spots with disagreement	278
#Spots with disagreement involving NA	218

Figure 7: Inter-annotator agreement.

1: $\gamma_0 \leftarrow \arg\max_{\gamma \in \Gamma_s} w^\top f_s(\gamma)$
2: **if** $w^\top f_s(\gamma_0) > \rho_{\text{NA}}$ **then return** γ_0 **else return** NA

If $f_s(\cdot)$ does not include the sense probability prior, we call the above strategy LOCAL, otherwise we call it LOCAL+Prior.

4.3 Effect of learning node potentials

M&W use two important signals, relatedness and commonness, in their disambiguator. In Figure 8 we present ablation studies showing the relative effectiveness of various features, together with the benefits of using *all* features with a learnt model w.

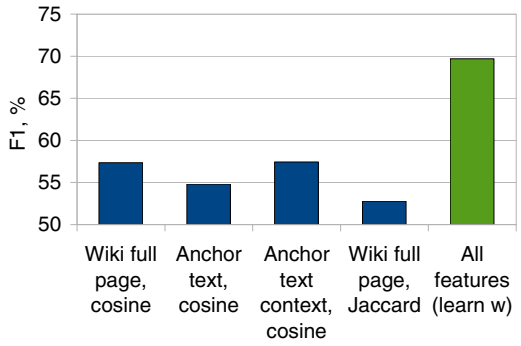

Figure 8: Training a model w is better than using any single spot-label compatibility feature.

4.4 Comparison with earlier algorithms

Rather surprisingly, LOCAL already produced significantly better F_1 scores than the two state-of-the-art annotations systems by M&W and Cucerzan.

The M&W algorithm can be directly executed on any page text using a Web service API[3]. The API includes a knob to control the recall-precision balance. We implemented Cucerzan's algorithm locally. Cucerzan's algorithm does not have a recall-precision knob. In LOCAL, we used ρ_{NA} as the knob.

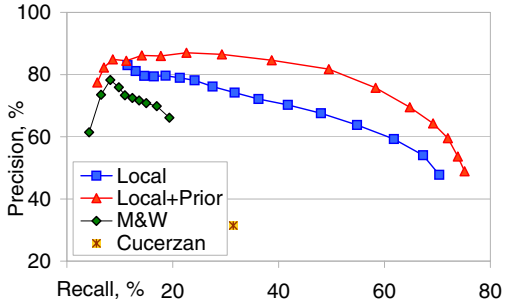

Figure 9: Even a non-collective Local approach that only uses trained node potential dominates both Cucerzan and M&W's algorithms wrt both recall and precision (IITB data).

Figure 9 shows recall-precision plots. Cucerzan's algorithm is shown by a single point. M&W's precision is very high, consistent with their claims. However, the R/P knob cannot increase recall beyond 20%. Meanwhile, the ρ_{NA} knob in LOCAL can be used to push it to 70% recall while remaining comparable to M&W precision. If we dial down our recall to levels comparable with M&W, our precision becomes visibly larger than M&W. Cucerzan's recall and precision are both dominated by LOCAL, like M&W. LOCAL+Prior is substantially better than LOCAL, and is a formidable F_1 level to beat.

Cucerzan did not learn the node potential but hardwired it. We gave Cucerzan's algorithm the benefit of our learned node potentials. The F_1 score improved to 51.8%, which was still short of LOCAL and far short of LOCAL+Prior.

[3] `http://www.nzdl.org/pohutukawa/wikifier/index.jsp`

4.5 HILL1 update trajectories

In Figure 10 we consider the trajectory of several documents (one line per document) as HILL1 optimizes their labels. Specifically, we plot the objective minus the ρ_{NA} contribution on the x-axis, and correspondingly, the F_1 score for spots that are marked some non-NA label in ground truth on the y-axis. Although there are occasional expected setbacks and oscillations, increasing the objective is generally good for F_1 too. This lends credibility to our basic dominant-cluster model.

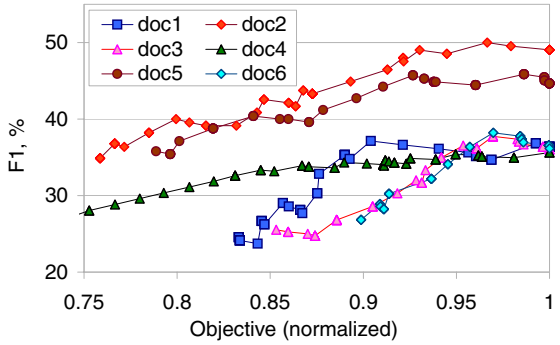

Figure 10: As Hill1 improves our proposed objective, it usually improves F_1 as well (IITB data).

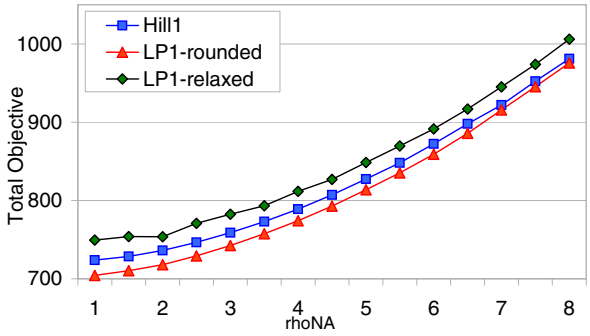

Figure 11: Hill1 can attain objectives comparable to relaxed LP1 (IITB data).

4.6 HILL1 vs. LP1

For over 70% of the documents, LP1 gives fully integral solutions, which are therefore optimal for our integer programs. Even otherwise, LP1 gives an efficiently computable, yet reliable upper bound to the objective that HILL1 is trying to attain. Figure 11 shows that in practice, the integrality gap is small, that HILL1 gets reasonably close to the upper

bound, and that rounding makes LP1 slightly *worse* than Hill1.

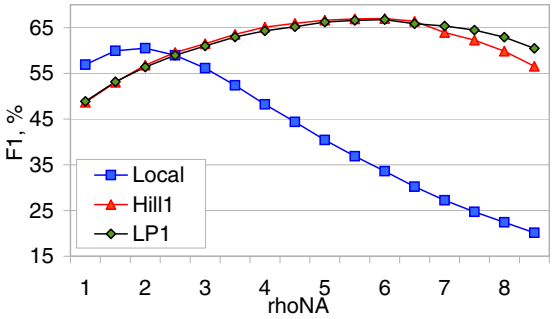

Figure 12: Hill1 attains almost the same F_1 score as LP1; both are better than Local (IITB data).

More directly useful is Figure 12, which compares F_1 scores of Hill1 and LP1 (rounded). They are very close, but Hill1 is slightly better at high recall levels of our interest. Both Hill1 and LP1 are robust to ρ_{NA}, whereas Local suffers if ρ_{NA} is chosen poorly.

Hill1 and LP1 scale mildly quadratically wrt $|S_0|$, as shown in Figure 13. For most documents, Hill1 takes about 2–3 seconds and LP1 takes around 4–6 seconds, much of which is fixed overhead.

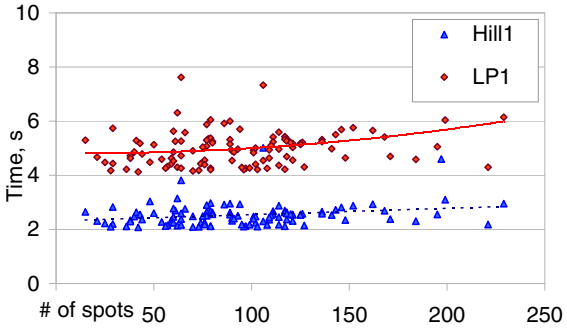

Figure 13: Scalability of Hill1 and LP1, IITB data.

4.7 Comparing Local, Hill1, LP1

Having established that Local alone can significantly improve upon prior work, we investigate whether collective inference gives additional accuracy gains compared to Local. In Figures 14 and 15 we plot precision against recall for the Local, Hill1, and LP1, for our two data sets.

In case of the IITB data set, we see that collective inference has distinct precision advantage (almost 9%), especially as we push recall aggressively beyond 70–75%. Summarized below are F_1 scores obtained by 2-fold cross-validation of ρ_{NA}:

	Local	Hill1	LP1
no Prior	63.45%	64.87%	67.02%
+Prior	68.75%	67.46%	69.69%

From Figure 6, we see that the CZ data is much smaller, sparse in ground truth annotations, but has more potential ambiguity. Here LP1 led to more fractional solutions and overall worse accuracy than Hill1, which still beat M&W's $F_1 = 63\%$ with our score of 69%, although M&W attained larger precision than us at lower recall.

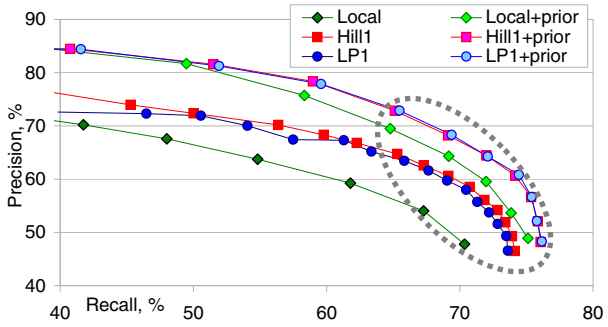

Figure 14: Recall/precision on IITB data.

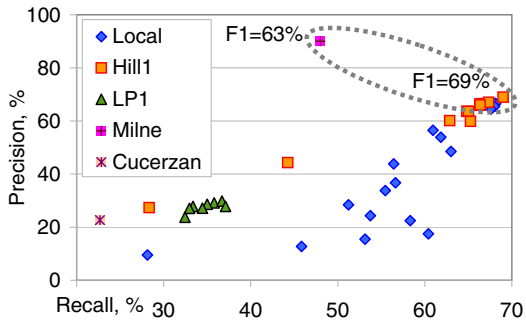

Figure 15: Recall/precision on Cucerzan's data.

5. MULTI-CLUSTER MODELS

Our clique potentials (expressions 1, 3, 4) implicitly encourage a single cluster model, because the clique potentials are largest when all $g(\gamma)$ are close to each other. Let $\Gamma^* = \bigcup_s y_s^*$ be the entity labels used on the optimal assignment. Is it true that a clustering of Γ^* in $g(\cdot)$-space will show one giant component cluster?

Figure 16 shows a dendrogram formed by agglomeratively clustering Γ^*, the ground truth entities on a page. The "single cluster hypothesis" is only somewhat true. There is a cluster corresponding to the broad topic of the page, but this typically covers fewer than a third of the spots. The rest belong to smaller clusters, or are singletons, in which case they bear no collective information.

Might our implicitly single-cluster model losing out on recall by not modeling and covering multiple clusters? Here is how this could happen. Say Hill1 is considering changing y_s from NA to γ, where s is a member of a non-dominant cluster. Therefore, $(1/\binom{|S_0|}{2}) \sum_{s' \neq s} r(y_{s'}, \gamma)$ may be too small to overcome the ρ_{NA} barrier. The large $\binom{|S_0|}{2}$ denominator could share the blame. The end result is that a small but tight cluster of such spots cannot "secede" from the dominant cluster.

Accordingly, we retained the node potential (NP), but demanded that an inference algorithm produce not only a label vector y but also a partitioning $C = \Gamma^1, \ldots, \Gamma^K$ of the labels used. We modified the clique potential to

$$\frac{1}{|C|} \sum_{\Gamma^k \in C} \frac{1}{\binom{\Gamma^k}{2}} \sum_{s,s':\, y_s, y_{s'} \in \Gamma^k} r(y_s, y_{s'}). \qquad \text{(CPK)}$$

By using denominator $\binom{\Gamma^k}{2}$ instead of $\binom{S_0}{2}$, this objective rewards smaller coherent clusters as desired, but it is no longer possible to express a simple linear objective as in

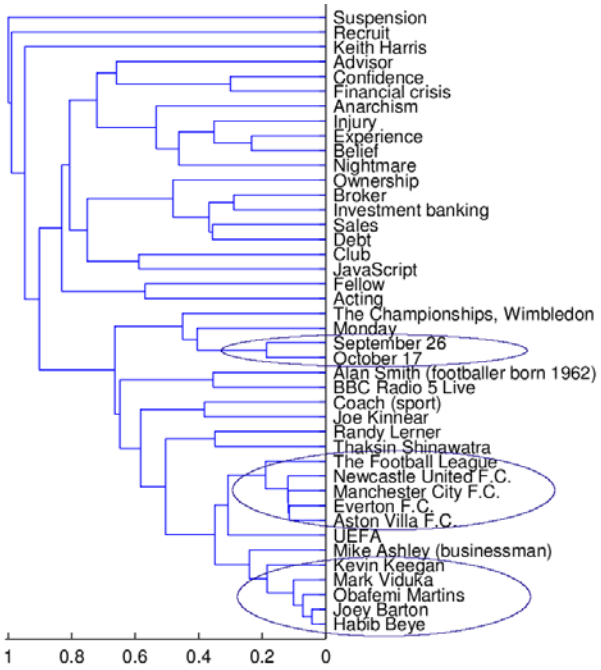

Figure 16: Hierarchical clustering of Γ^* using the g embedding shows more than one clusters.

(CP1′), because $|\Gamma^k|$ themselves depend on y and C.

We extended the LP1 framework to optimize (NP)+(CPK) approximately. In experiments, (CPK) did not perform significantly better than LP1, giving less than 0.5% F_1 boost. We conjecture that this is because of the extreme sparsity of $r(\gamma, \gamma')$, which had only 5% fill. Basically, if $r(\gamma, \gamma')$ were used as edges in a graph, the graph is easy to partition, and LP1 finds it easy to make correct decisions within each partition, even if the LP1 objective tries to account for cross-cluster edges. However, this may change with denser sources of relatedness information.

6. CONCLUSION AND OUTLOOK

We proposed new models and algorithms for a highly motivated problem: annotating unstructured (Web) text with entity IDs from an entity catalog (Wikipedia). Unlike prior work that is biased toward specific entity types like persons and places, with low recall and high precision, our intention is aggressive, high-recall open-domain annotation for indexing and mining tasks downstream.

Our main contribution is a formulation that captures a tradeoff between local spot-to-label compatibility and a global, document-level topical coherence between entity labels. Inference in this model is intractable in theory, but we show that LP relaxations often give optimal integral solutions or achieve close to the optimal objective. We also give a simple local hill-climbing algorithm that is comparable in speed and quality to LP relaxation. Both these algorithms are significantly better than two recently-proposed annotation algorithms.

In continuing work, we are trying to cast the annotation problem as special cases of quadratic assignment that can be approximated well [6, 16] or show that even approximation is difficult [10, 16]. We are trying to combine the generally low-recall, high-precision nature of M&W's $r(\gamma, \gamma')$ based on inlinks with the converse properties of Cucerzan's $r(\gamma, \gamma')$ based on categories. This involves extending the training process from NP to the whole objective. We are also investigating why the multi-cluster extensions of our model obtained no significant accuracy gains. Finally, we are considering collective decisions beyond page boundaries.

7. REFERENCES

[1] E. Agirre and G. Rigau. Word sense disambiguation using conceptual density. In *Computational Linguistics*, pages 16–22. Association for Computational Linguistics, 1996.

[2] S. Agrawal, K. Chakrabarti, S. Chaudhuri, V. Ganti, A. C. König, and D. Xin. Exploiting web search engines to search structured databases. In *WWW Conference*, pages 501–510, 2009.

[3] R. Bunescu and M. Pasca. Using encyclopedic knowledge for named entity disambiguation. In *EACL*, pages 9–16, 2006.

[4] S. Cucerzan. Large-scale named entity disambiguation based on Wikipedia data. In *EMNLP Conference*, pages 708–716, 2007.

[5] S. Dill et al. SemTag and Seeker: Bootstrapping the semantic Web via automated semantic annotation. In *WWW Conference*, 2003.

[6] U. Feige, D. Peleg, and G. Kortsaz. The dense k-subgraph problem. *Algorithmica*, 29(3):410–421, Dec. 2001.

[7] M. Garey and D. Johnson. Computers and intractability: A guide to the theory of NP-completeness, 1979.

[8] R. V. Guha and R. McCool. TAP: A semantic web test-bed. *Journal of Web Semantics*, 1(1):81–87, 2003.

[9] M. I. Jordan, editor. *Learning in Graphical Models*. MIT Press, 1999.

[10] S. Khot. Ruling out PTAS for graph min-bisection, densest subgraph and bipartite clique. In *FOCS Conference*, pages 136–145, 2004.

[11] R. Larson. Bibliometrics of the world wide web: An exploratory analysis of the intellectual structure of cyberspace. In *Annual Meeting of the American Society for Information Science*, 1996. Online at `http://sherlock.berkeley.edu/asis96/asis96.html`.

[12] D. B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):32–38, nov 1995. Also see `http://www.cyc.com/` and `http://www.opencyc.org/`.

[13] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM*, pages 233–242, 2007.

[14] G. Miller, R. Beckwith, C. FellBaum, D. Gross, K. Miller, and R. Tengi. Five papers on WordNet. Princeton University, Aug. 1993.

[15] D. Milne and I. H. Witten. Learning to link with Wikipedia. In *CIKM*, 2008.

[16] V. Nagarajan and M. Sviridenko. On the maximum quadratic assignment problem. In *SODA*, pages 516–524. Society for Industrial and Applied Mathematics, 2009.

[17] S. Sarawagi. Information extraction. *FnT Databases*, 1(3), 2008.