Unsupervised Personal Name Disambiguation

Gideon S. Mann and David Yarowsky
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218 USA
{gsm,yarowsky}@cs.jhu.edu

Abstract

This paper presents a set of algorithms for distinguishing personal names with multiple real referents in text, based on little or no supervision. The approach utilizes an unsupervised clustering technique over a rich feature space of biographic facts, which are automatically extracted via a language-independent bootstrapping process. The induced clustering of named entities are then partitioned and linked to their real referents via the automatically extracted biographic data. Performance is evaluated based on both a test set of hand-labeled multi-referent personal names and via automatically generated pseudonames.

1 Introduction

One open problem in natural language ambiguity resolution is the task of proper noun disambiguation[1]. While word senses and translation ambiguities may typically have 2-20 alternative meanings that must be resolved through context, a personal name such as "Jim Clark" may potentially refer to hundreds or thousands of distinct individuals. Each different referent typically has some distinct contextual characteristics. These characteristics can help distinguish, resolve and trace the referents when the surface names appear in online documents.

A search of Google shows 76,000 web pages mentioning Jim Clark, of which the first 10 unique referents are:

1. Jim Clark - Race car driver from Scotland
2. Jim Clark - Clockmaker from Colorado
3. Jim Clark - Film Editor
4. Jim Clark - Netscape Founder
5. Jim Clark - Disaster Survivor
6. Jim Clark - Car Salesman in Kansas
7. Jim Clark - Fishing Instructor in Canada
8. Jim Clark - Computer Science student in Hong Kong
9. Jim Clark - Professor at McGill
10. Jim Clark - Gun Dealer in Louisiana

In this paper, we present a method for distinguishing the real world referent of a given name in context. Approaches to this problem include Wacholder et al. (1997), focusing on the variation of surface name for a given referent, and Smith and Crane (2002), resolving geographic name ambiguity. We present preliminary evaluation on *pseudonames*: conflations of multiple personal names, constructed in the same way pseudowords are used for word sense disambiguation (Gale et al., 1992). We then present corroborating evidence from real personal name polysemy to show that this technique works in practice.

	Miles Davis
birth day	**May 26**(5), **May 25**(5)
birth year	**1926**(82), 1967(18), 1969(9)...
occupation	**trumpeter**(38), **artist**(10), **player**(5)...
birth place	**Alton**(7), **Illinois**(6)
	Joerg Haider
birth year	**1950**(6)
occupation	**leader**(198) **politician**(93) **chairman**(6)...
birth place	**Austria**(1)

Table 1: Extracted Biographical Information from 1000 Web Pages

Another topic of recent interest is in producing biographical summaries from corpora (Schiffman et al., 2001). Along with disambiguation, our system simultaneously collects biographic information (Table 1). The relevant biographical attributes are de-

[1]This has been recognized even by the popular press. Reuters (March 13, 2003) observed the problem of name ambiguity to be a major stumbling block in personal name web searches.

picted along with a clustering which shows the distinct referents (Section 4.1).

2 Robust Extraction of Categorical Biographic Data

Past work on this task (e.g. Bagga and Baldwin, 1998) has primarily approached personal name disambiguation using document context profiles or vectors, which recognize and distinguish identical name instances based on partially indicative words in context such as *computer* or *car* in the Clark case. However, in the specialized case of personal names, there is more precise information available.

In particular, information extraction techniques can add high precision, categorical information such as approximate age/date-of-birth, nationality and occupation. This categorical data can support or exclude a candidate name↔referent matches with higher confidence and greater pinpoint accuracy than via simple context vector-style features alone.

Another major source of disambiguation information for proper nouns is the space of associated names. While these names could be used in a undifferentiated vector-based bag-of-words model, further accuracy can be gained by extracting specific types of association, such as familial relationships (e.g. son, wife), employment relationships (e.g. manager_of), and nationality as distinct from simple term co-occurrence in a window. The Jim Clark married to "Vickie Parker-Clark" is likely not the same Jim Clark married to "Patty Clark". Additionally, information about one's associates can help predict information about the person in question. Someone who frequently associates with Egyptians is likely to be Egyptian, or at the very least, has a close connection to Egypt.

2.1 Generating Extraction Patterns

One standard method for generating extraction patterns is simply to write them by hand. In this paper, we have experimented with generating patterns automatically from data. This has the advantage of being more flexible, portable and scalable, and potentially having higher precision and recall. It also has the advantage of being applicable to new languages for which no developer with sufficient knowledge of the language is available.

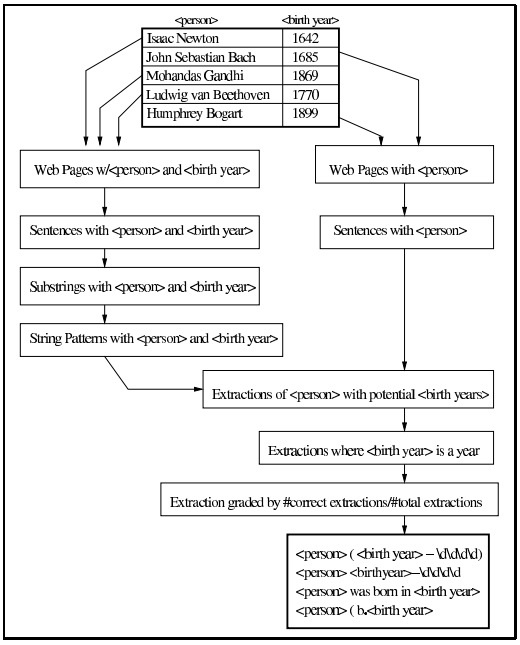

Figure 1: Learning Extraction Patterns from Filled Templates and Web Pages

In the late 90s, there was a substantial body of research on learning information extraction patterns from templates (Huffman, 1995; Brin, 1998; Califf and Mooney, 1998; Freitag and McCallum, 1999; Yangarber et al., 2000; Ravichandran and Hovy, 2002). These techniques provide a way to bootstrap information extraction patterns from a set of example extractions or seed facts, where a tuple with the filled roles for the desired pattern are given. For the task of extracting biographical information, each example would include the personal name and the biographic feature. For example, training data for the pattern *born in* might be ("Wolfgang Amadeus Mozart",1756). Given this set of examples, each method generates patterns differently.

In this paper, we employ and extend the method described by Ravichandran and Hovy (2002) shown in Figure 1. For each seed fact pair for a given template (such as *(Mozart,1756)*), a web query is made which in turn leads to sentences in which the roles are observed in nearby association (e.g. "*Mozart* was born in *1756*"). All substrings from these sentences are then extracted. The substrings are then subject to simple generalization, to produce candidate patterns: Mozart is replaced by <name>, 1756 is replaced by <birth year>, and all digits are replaced by #. These substring templates can

English

Purely Syntactic Patterns		
Pattern Template	Precision	Count
<name> (<birth year> - ####)	1	31
<name> (<birth year> - ####	1	31
- <name> (<birth year>-####)	1	30
- <name> (<birth year>-####	1	30
<name> <birth year>-####	1	27
<name> (<birth year>-####) -	1	26
<name> <name> (<birth year>	1	18

Syntactic & Lexical		
Pattern Template	Precision	Count
<name> *was born in* <birth year>	1	19
<name> *was born in* <birth year> *in*	1	12
by <name> (<birth year>-####)	1	10
by <name> (<birth year>-####	1	10
of <name> (<birth year>-####)	0.933	15
of <name> (<birth year>-####	0.933	15
<name> (<birth year>-####) was	0.833	12

Spanish

Purely Syntactic Patterns		
Pattern Template	Precision	Count
. <name> (<birth year>-	1	62
. <name> (<birth year>-##	1	58
. <name> (<birth year>-####	1	55
. <name> (<birth year>-####)	1	54
<name> (<birth year>-####) :	1	38
<name> <birth year>-#### ,	1	26
<name> , <birth year>-####	1	25

Syntactic & Lexical		
Pattern Template	Precision	Count
a <name> (<birth year>-####	1	30
a <name> (<birth year>-####)	1	29
<birth year> . - *Nace* <name>	1	21
<birth year> . - *Nace* <name> ,	1	17
<name> (<birth year>-####) , *con*	1	15
<name> (<birth year>-####) *se*	1	12
, *de* <name> (<birth year>-####)	1	12

Table 2: Highest Precision Patterns Extracted for English and Spanish using Suffix Tree Methodology

then serve as extraction patterns for previously unknown fact pairs, and their precision in fact extraction can be calculated with respect to a set of currently known facts.

We examined a subset of the available and desirable extracted information. We learned patterns for birth year and occupation, and hand-coded patterns for birth location, spouse, birthday, familial relationships, collegiate affiliations and nationality. Other potential patterns currently under investigation include employer/employee and place of residence.

2.2 Multilingual Information Extraction

We adapted the information extraction pattern generation techniques described above to multiple languages. In particular, the methodology proposed by Ravichandran and Hovy (2002) requires no parsing or other language specific resources, so is an ideal candidate for multilingual use. In this paper, we conducted an initial test test of the viability of inducing these information extraction patterns across languages. To test, we constructed a initial database of 5 people and their birthdays, and used this to induce the English patterns. We then increased the database to 50 people and birthdays and induced patterns for Spanish, presenting the results above. Figure 2 shows the top precision patterns extracted for English and for Spanish.

It can be seen that the Spanish patterns are of the same length, with similar estimated precision, as well as similar word and punctuation distribution as the English ones. In fact, the purely syntactic patterns look identical. The only difference being that to generate equivalent Spanish data, a database of training examples an order of magnitude larger was required. This may be because for each database entry more pages were available on English websites than on Spanish websites.

3 Using Unsupervised Clustering to Identify the Referents of Personal Names

This section examines clustering of web pages which containing an ambiguous personal name (with multiple real referents). The cluster method we employed is bottom-up centroid agglomerative clustering. In this method, each document is assigned a vector of automatically extracted features. At each stage of the clustering, the two most similar vectors are merged, to produce a new cluster, with a vector equal to the centroid of the vectors in the cluster. This step is repeated until all documents are clustered.

To generate the vectors for each document, we explored a variety of methods:

1. Baseline : All words (**plain**) or only Proper Nouns (**nnp**)
2. Most Relevant words (**mi** and **tf-idf**)
3. Basic biographical features (**feat**)
4. Extended biographical Features (**extfeat**)

word	weight(**mi**)	weight(**extfeat**)
adderley	5.30	0
snipes	5.16	0
coltrane	5.06	0
montreux	5.01	0
bitches	4.99	0
danson	4.97	0
hemp	4.97	0
mullally	4.95	0
porgy	4.94	0
remastered	4.92	0
actor	3.50	2.40
1926	0	2.20
trumpeter	0	2.20
midland	0	1.39

Table 3: The 10 words with highest mutual information with the document collection and all of extended feature words for DAVIS/HARRELSON pseudoname

3.1 Baseline Models

In our baseline models, we used term vectors composed either of all words (minus a set of closed class "stop" words) or of only proper nouns. To assess similarity between vectors we utilized standard cosine similarity ($cos(a,b) = \frac{a \cdot b}{||a|| \times ||b||}$).

We experimentally determined that the use of proper nouns alone led to more pure clustering. As a result, for the remainder of the experiments, we used only proper nouns in the vectors, except for those common words introduced by the various feature sets.

3.2 Relevant Words (mi and tf-idf)

Selective term weighting has been shown to be highly effective for information retrieval. For this study, we investigated both the use of standard TF-IDF weighting and weighting based on the mutual information, where given a document collection c, for each word w, we calculate $I(w;c) = \frac{p(w|c)}{p(w)}$. From these, we select words which appear more than $\lambda_1 = 20$ times in the collection, and have a $I(w;c)$ greater than $\lambda_2 = 10$. These words are to the document's feature vector with a weight equal to $log(I(w;c))$.

3.3 Extracted Biographical Features (feat)

The next set of models use the features extracted using the methodology described in Section 2. Biographical information such as birth year, and occupation, when found, is quite useful in connecting documents. If a document connects a name with a birth year, and another document connects the same name with the same birth year, typically, those two documents refer to the same person.

Type	Extracted Feature
`birth place`	Midland(4), Texas (3), Alton(1), Illinois(1)
`birth year`	1926 (9), 1967(3), 1973(2), 1947(1), 1958(1), 1969(1)
`occupation`	actor (11), trumpeter(9), heavyweight(2) ...
`spouse`	Demi Moore(1)

Table 4: **feat**: Features Extracted for DAVIS/HARRELSON pseudoname

These extracted features were used to categorically cluster documents in which they appeared. Because of their high degree of precision and specificity, documents which contained similar extracted features are virtually guaranteed to have the same referent. By clustering these documents first, large high quality clusters formed, which then then provided an anchor for the remaining pages. By examining the dendrogram in Figure 3, it is clear that the clusters start with documents with matching features, and then the other documents cluster around this core.

In addition to improving disambiguation performance, these extracted features help distinguish the different clusters, and provide information about the different people.

3.4 Extended Biographical Features (extfeat)

Another method for using these extracted features is to give higher weight to words which have ever been seen as filling a pattern. For example, if 1756 is extracted as a birth year from a syntactic-based pattern for the polysemous name, then whenever 1756 is observed anywhere in context (outside an extraction pattern), it is given a higher weighting and added to the document vector as a potential biographic feature. In our experiments, we did this only for words which appeared as values for a feature more than a threshold of 4 times. Then, whenever the word was seen in a document, it was given a weight equal to the log of the number of times the word was seen as an extracted feature.

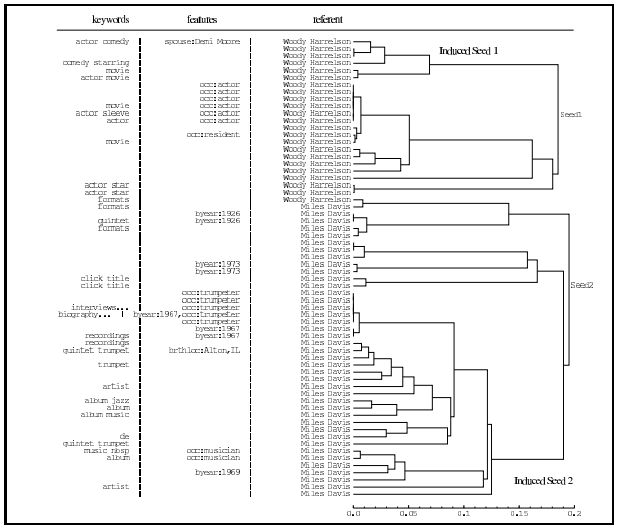

Figure 2: **nnp+feat+extfeat+mi** Clustering Visualization for DAVIS/HARRELSON pseudoname

3.5 Cluster Refactoring

Ideally, the raw unsupervised clustering would yield a top level distinction between the different referents. However, this is rarely the case. With this type of agglomerative clustering, the most similar pages are clustered first, and outliers are assigned as stragglers at the top levels of the cluster tree. This typically leads to a full clustering where the top-level clusters are significantly less discriminative than those at the roots. In order to compensate for this effect, we performed a type of tree refactoring, which attempted to pick out and utilize **seed** clusters from within the entire clustering.

In the refactoring, the clustering is stopped before it runs to completion, based on the percentage of documents clustered and the relative size of the clusters achieved. At this intermediate stage, relatively large and high-precision clusters are found (e.g. Figure 2). These automatically-induced clusters are then used as seeds for the next stage, where the unclustered documents are assigned to the seed with the closest distance measure (Figure 3).

An alternative to this form of cluster refactoring would be to initially cluster only pages with extracted features. This would yield a set of cluster seeds, divided by features, which could then be used for further clustering. However, this method relies on having a number of pages with extracted features that overlap from each referent. This can only be assured when the feature set is rich, or a large document space is assumed.

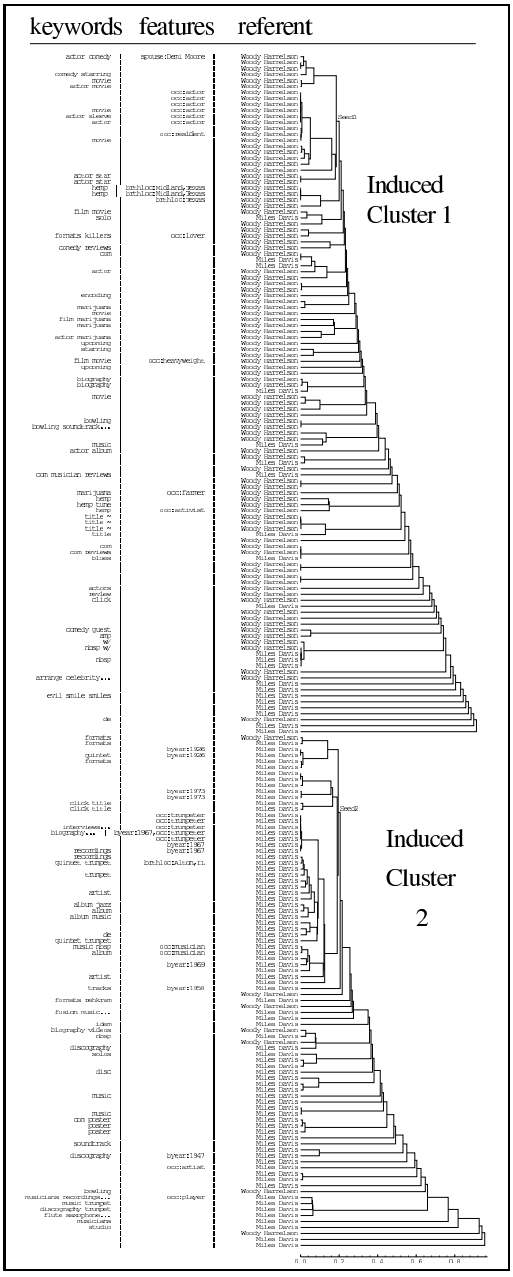

Figure 3: **nnp+feat+extfeat+mi** Clustering Visualization for DAVIS/HARRELSON pseudoname

4 Experiments

To test these clustering methods, we collected web pages by making requests to the Google website for a set of target personal names (up to a maximum of 1000 pages per name). There was no require-

ment that the web page be focused on that name, nor was there a minimum number of name occurrences. As a result, some pages clustered only mentioned the name in passing, or in a specialized, commercial context (e.g. Amazon sales product).

The pseudonames were created as follows. The retrieval results from two different randomly-selected people were taken, and all references to either name (in full or part) replaced by a unique, shared pseudoname. The resulting collection then consisted of documents which were ambiguous as to whom they talked about. The aim of the clustering was then to distinguish this artificially conflated pseudoname. In addition, a test set of four naturally occurring polysemous names (such as Jim Clark), containing an average of 60 instances each, was manually annotated with distinguishing nameID numbers and used for a parallel evaluation.

The experiments consist of two parts. The first output is the clustering visualizations whose utility can be judged by inspection. The second is a quantitative analysis of the different methodologies. Both are conducted over test sets of pseudonames and naturally occurring ambiguities.

4.1 Clustering Visualizations

Figures 2/3 and 4 each have two subfigures. The left/top figure shows the extracted seed sets. The right/bottom figure shows the final clustering of the entire document collection. In each figure, there are three columns of information before the dendrogram. The first column contains high weighted document content words. The second column contains the extracted features from the document. The third column indicates the real referent. This is either the real name of the conflated pseudoname (e.g. Woody Harrelson or Miles Davis), or a number indicating the referent (e.g. 1 - 20 in the case of Jim Clark). This presentation allows a quick scan of the clustering to reveal correlations.

In general, the visualizations are informative. Occasionally, the extractions err. One time when the patterns themselves cannot be syntactically faulted comes in the case where Woody Harrelson's wife is extracted as Demi Moore. The information was extracted from the sentence: "Architect Woody Harrelson and his wife realtor Demi Moore ..." which appears as a plot description for the movie "Indecent Proposal". Here, untangling of synecdoche is needed. For Miles Davis, the incorrectly extracted birth years refer to record release dates, which take the same surface form as birth years in some genres.

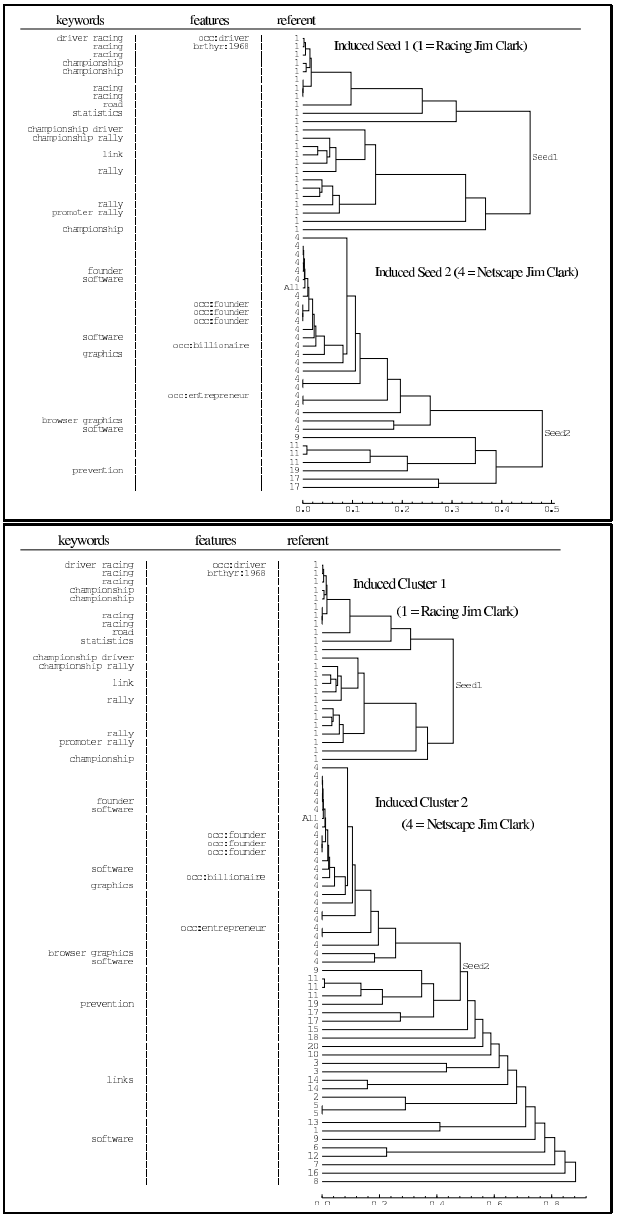

Figure 4: **nnp+feat+extfeat+mi** Clustering Visualization of Jim Clark Pages: "1"=Race Car Driver, "4"=Netscape Founder, "A"=multiple referents

Figure 4 shows a clustering for a naturally occuring name ambiguity, in particular that of web pages which refer to "Jim Clark". The set was constructed by retrieving 100 web pages, and then labeling the pages with respect to their referent. As can be seen,

the clusterings are highly coherent. All of the relevant pages are included in the seed set, and few inappropriate pages are added. This type of clustering would be useful to someone searching for a specific individual named Jim Clark. Once the clustering had been performed, a user could scan the output, and identity the "Jim Clark" of interest, based both on extracted features and key words.

4.2 Evaluation on Pseudonames

For automated pseudoname evaluation purposes, we selected a set of 8 different people for conflation, who we presumed had one vastly predominant sense. We selected these people giving room for historical figures, figures from pop culture and modern media culture, as well as "ordinary" people. We added people with similar backgrounds (born close to each other, or having the same profession). The full list was composed of these 8 individuals:

> *Haifa Al-Faisal, William Blake, Tom Cruise, Woody Harrelson, Hermann Hesse, Wolfgang Amadeus Mozart, Anna Shusterman, Bryon Tosoff*

For each, we submitted Google queries, and retrieved up to 1000 pages each. We then took these hit returns, and subsampled to a maximum of 100 pages per person. The person with the smallest representation was Anna Shusterman with 26 pages. We subsampled by taking the first 100 as ordered lexically. This may have biased the results somewhat towards unreliable web pages, since pages with numeric addresses tend to be newer and more transient.

We evaluated two guanularities of feature extraction. The small feature set uses high precision rules to extract occupation (occ), birthday (brthyr), spouse, birth location (brthloc), and school. The large feature set adds higher recall (and therefore noisier) patterns for the previous relationships and as well as parent/child relationships.

As can be seen from the table, the highest performing system combines proper nouns, relevant words, and the high precision extracted features (**nnp+feat+mi** and **nnp+feat+tfidf**). The extended features (**nnp+feat+extfeat+mi**) do not give additional benefit to this combination. As can be seen from the table, the large feature set yields better overall performance than the smaller feature set.

Clustering Method	Disambiguation Accuracy	
no extracted features		
majority sense	62.5	
plain	74.5	
tfidf	76.7	
nnp	79.7	
nnp+tfidf	79.7	
nnp+mi	**82.9**	
w/ extracted features	feature set size	
	small	large
nnp+feat	82.5	85.1
nnp+feat+extfeat	82.0	84.6
nnp+feat+mi	85.6	85.2
nnp+feat+tfidf	82.9	**86.4**

Table 5: Disambiguation Accuracy of different Clustering Methods over 28 pseudonames

This suggests that the increased coverage outweighs the introduced noise.

For the **feat+tfidf** system, accuracy at the two-class disambiguation was above 80% for 25 out of the 28 pairs. Without these pairs, the average two-class disambiguation performance over the remaining pairs is 90%. In two of the problematic cases, the contexts of the names are easily confusable, as the individuals share the same profession and many of the same keywords. More complete biographic profiles and different clustering biases would be helpful in fully partitioning these cases. However, in practice these pseudoname pair situations may be more difficult than expected for naturally occurring name pairs. In many occupations that are typically newsworthy (such as actors, authors, musicians, politicians, etc.), there may be a tendency for individuals to avoid using identical names (or entering the field entirely) to minimize confusion. When people with identical names do indeed share the same field one would expect a greater effort to providing disambiguating contextual features to distinguish them.

We have made some preliminary investigations into selecting pages according to the number of mentions, as opposed to by random. The results have not been conclusive, and continuing work is investigating the cause.

4.3 Evaluation on Naturally Ambiguous Names

The above results have utilized pseudoname test sets where high accuracy ground truth is automatically available in large quantities [O(1000) examples per name] to better distinguish model performance. Table 6 shows the performance on the four O(60) example hand-labeled test sets for naturally occurring polysemous person names. Given that this is an n-ary classification task, for consistency with the above experiments the data were assigned to one of 3 clusters, corresponding to the 2 automatically derived first-pass majority seed sets and the residual "other-use" classification, but evaluated strictly on performance for the two major senses. While additional analyses could be accomplished on the residual sets, this is difficult given their small size (remaining personal exemplars were mostly singletons) and lack of evidence on many single-mention web pages. Thus the task of accurately partitioning the two most common uses and clustering the residual examples for visual exploration may be a natural and practical use for these classification and visualization technologies.

Weighting Method	Precision	Recall
TF-IDF	.81	.70
Mutual Information	.88	.73

Table 6: Classification performance for naturally occurring name ambiguities on 3-way classification task (Majority-Use, Secondary-Use, Other-Use).

5 Conclusion

In this paper we have presented a set of algorithms for finding the real referents for ambiguous personal names in text using unsupervised clustering and feature extraction methods. In particular, we have shown how to learn and use automatically extracted biographic information to improve clustering results, and have demonstrated this improvement by evaluating on pseudonames. We have presented initial results on learning these patterns to extract biographic information for multiple languages, and intend to use these techniques for large-scale multilingual polysemous name clustering.

The results presented here support the automatic clustering of polysemous personal name referents and visualization of these induced clusters and their motivating features. These distinct referents can be verified by inspection both of extracted features and of the high weighted terms for each document. These clusterings may be useful in two ways. First as a useful visualization tool themselves, and second as seeds for disambiguating further entities.

References

A. Bagga and B. Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In Christian Boitet and Pete Whitelock, editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 79–85, San Francisco, California. Morgan Kaufmann Publishers.

S. Brin. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98*.

M. E. Califf and R. J. Mooney. 1998. Relational learning of pattern-match rules for information extraction. In *Working Notes of AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*, pages 6–11, Menlo Park, CA. AAAI Press.

D. Freitag and A. McCallum. 1999. Information extraction with hmms and shrinkage. In *Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction*.

B. Gale, K. Church, and D. Yarowsky. 1992. Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language Processing*, pages 54–60, Cambridge, MA.

S. B. Huffman. 1995. Learning information extraction patterns from examples. In *Learning for Natural Language Processing*, pages 246–260.

D. Ravichandran and E. Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

B. Schiffman, I. Mani, and K. J. Concepcion. 2001. Producing biographical summaries: Combining linguistic knowledge with corpus statistics. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*.

D. A. Smith and G. Crane. 2002. Disambiguating geographic names in a historic digital library. In *Proceedings of ECDL*, pages 127–136.

N. Wacholder, Y. Ravin, and M. Choi. 1997. Disambiguation of proper names in text. In *Proceedings of Fifth Conference on Applied Natural Language Processing*, pages 202–208.

R. Yangarber, R. Grishman, P. Tapanainen, and S. Huttunen. 2000. Unsupervised discovery of scenario-level patterns for information extraction. In *Proceedings of the Sixth Conference on Applied Natural Language Processing, (ANLP-NAACL 2000)*, pages 282–289.