Improved Unsupervised Name Discrimination with Very Wide Bigrams and Automatic Cluster Stopping

Ted Pedersen

University of Minnesota, Duluth, MN 55812, USA

Abstract. We cast name discrimination as a problem in clustering short contexts. Each occurrence of an ambiguous name is treated independently, and represented using second–order context vectors. We calibrate our approach using a manually annotated collection of five ambiguous names from the Web, and then apply the learned parameter settings to three held-out sets of pseudo-name data that have been reported on in previous publications. We find that significant improvements in the accuracy of name discrimination can be achieved by using very wide bigrams, which are ordered pairs of words with up to 48 intervening words between them. We also show that recent developments in automatic cluster stopping can be used to predict the number of underlying identities without any significant loss of accuracy as compared to previous approaches which have set these values manually.

1 Introduction

Person name ambiguity is an increasingly common problem as more and more people have online presences via Web pages, social network sites, and blogs. Since many distinct people share the same or similar names, it is often difficult to sort through results returned by search engines and other tools when looking for information about a particular person. There are many examples of identity confusion that have been widely publicized. For example, television talk show host Charlie Rose included his friend George Butler the filmmaker in his list of notable deaths from 2008. The only problem was that this George Butler was still alive, and it was George Butler the recording company executive who had died.

In general the goal of name discrimination is to associate or group the occurrences of person names with their true underlying identities. Our approach is completely unsupervised, and relies purely on the written contexts surrounding the ambiguous name. Our goal is to group these contexts into some (unspecified) number of clusters, where each cluster is associated with a unique individual. We assume that named entity recognition (NER) has already been carried out, so the input consists of text where the occurrences of person names are already identified.

There are various ways to formulate solutions to the problems surrounding name ambiguity or identity confusion, and so this paper tries to clarify exactly

A. Gelbukh (Ed.): CICLing 2009, LNCS 5449, pp. 294–305, 2009.

where this works falls in that spectrum. We make distinctions between our approach of name discrimination versus name *disambiguation*, and between our approach of treating contexts independently versus those that consider them to be dependent (as in cross–document co–reference resolution). We then go on to describe the general methodology of representing contexts with second–order co-occurrences, and then our specific enhancements to that approach that we arrived at via an extensive comparison with previously published results. In particular we focus on using very wide bigrams, which allow for up to 48 intervening words between an ordered pair of words, and on the use of automatic cluster stopping based on the clustering criterion function.

2 Discrimination versus Disambiguation

Name discrimination and name disambiguation are often confused or viewed as the same problem, yet there are important differences.

Name ambiguity can be approached as a problem in word sense disambiguation, where the goal is to assign a meaning to the surface form of a word. In this case, the different forms of a name (e.g., John Smith, Mr. Smith) are surface forms, and the underlying meaning is the unique identity associated with each occurrence of a name. In the case of disambiguation, these identities must be specified in a pre–existing inventory. In word sense disambiguation these inventories take the form of dictionaries, whereas in name disambiguation the inventory may be found in a biographical database or Wikipedia.

The key characteristic of name disambiguation is that the nature and number of possible identities is specified in advance of solving the problem. This means that discovering the number of identities or the nature of those identities has already been accomplished via some means, and is not a part of the problem.

However, in many practical settings a pre–defined inventory of possible identities simply isn't available. In that case, the best that can be hoped for is to carry out *name discrimination*, where we cluster the contexts in which the surface forms of a name occur into groups and thereby discover the number of distinct identities that share the same name. It may also be possible to describe each unique identity by analyzing the contents of each cluster; this is a task we refer to as *cluster labeling* (c.f., [6]). Given the rapidly changing nature of the online world, we believe that in general it's not realistic to assume that a pre–existing identity inventory will be available or can easily be constructed, so our work has focused on discrimination.

For example, in Web search it is very hard to predict what names a user might choose to search for, and the number of new names and identities that appear on the Web changes rapidly. A user searching for a given name (e.g., George Miller) will find some number (often more than they expect) of pages that include this name. The user must then determine which pages belong to the *George Miller* they are interested in, and which ones are about a different *George Miller*. As the user surveys the results, they may ask questions like : "... Did the George Miller that developed WordNet also write *The Magical Number Seven*?"

(Yes). "... Did Professor George Miller of Princeton direct the movie *Mad Max*?" (No). Performing this type of discrimination task manually can be difficult and error prone (as the Charlie Rose example above possibly suggests). Automatic methods that help organize the different contexts in which a name occurs into different clusters could be useful in identifying the underlying identities more conveniently and reliably.

The problem of name discrimination has drawn increasing attention, including the recent Web People Search Task (WePS) at SemEval-2007 [1], which included 16 participating teams. There is a second Web People Search Task taking place in late 2008 and early 2009 as well. The goal of the WePS tasks is to cluster Web pages based on the underlying identity of a given name. This is a variant of the name discrimination task where the context in which the name occurs is an entire web page. This task resulted in the release of a manually disambiguated corpus of names. It consists of 4,722 contexts (web pages) in which 79 names occur. These names represent a total of 1,954 entities. This corpus has a average of 59.8 pages per name, and 24.7 entities per name.

3 Relation to Cross–Document Co–reference Resolution

Our approach to name discrimination treats each context as an independent occurrence of a name, and clusters each occurrence without regard to the document from which it originally came (or its position within that document).

It is also possible to take the view that all the occurrences of a name in a document collectively form a single context, and simply assume that all of the surface forms of a name in that document refer to the same underlying identity. This is a slight variation on the one sense (of a word) per discourse hypothesis of Gale, Church and Yarowsky [3] for word sense disambiguation.

This representation of context is characteristic of cross-document co–reference resolution, where the problem is to determine if the surface forms of a particular name found in multiple documents refer to the same identity or different ones. The sequence of named entities in a document are referred to as chains, and the goal is to link together chains across documents that refer to the same person.

While this might appear to be a somewhat different problem, it is in fact very similar to name discrimination. The only real difference is the size of the context, which now includes all the occurrences of a name in a document, and either the sentences in which they occur or some fixed size window of surrounding context. As a result, the methods we describe in this paper can be used without modification to solve cross–document co–reference resolution simply by allowing contexts to include multiple occurrences of a name. If there is only one occurrence of a name in a document, then it is exactly equivalent to the name discrimination problem.

In fact, linking chains of references found across documents is nearly equivalent to assigning the contexts in which a name occurs to a cluster. The only difference is that the order of the documents is preserved when linking rather

than clustering. However, such orderings are relatively easy to recover from a cluster, especially if the document names indicate the order.

An early approach to cross–document co–reference resolution is described by Bagga and Baldwin [2]. They identify co–reference chains within a document between the multiple occurrences of a person name. Then they create a first order bag of words feature set from the sentences that occur around the person names in the chain in order to create a context vector that will represent the person name in that document. Thus, there is one vector per person name per document, and these vectors are clustered to determine the number of underlying individuals that share that name. They make pairwise comparisons between these vectors and those that are sufficiently similar (according to a predetermined threshold) are judged to refer to the same underlying identity and are placed in the same cluster. They did experiments using the John Smith Corpus, which includes 197 articles from the 1996-1997 New York Times that include the surface form John Smith (or various variations). There were 11 different entities mentioned more than one time in 173 articles, and 23 singleton entities, leading to 197 total articles in the corpus.

Gooi and Allan [4] evaluated several statistical methods on the John Smith corpus, and on their Person-X corpus, which is made up of name conflations or pseudo–names, where multiple named entities are disguised by replacing their surface form with Person-X. They created this corpus by searching for different subject domains in TREC volumes, which yielded 34,404 documents. Then, each document was processed with a named entity recognizer. A single person name was randomly selected from each document, and disguised throughout all the documents with Person-X. There were 14,767 distinct entities that were disguised.

Each occurrence of Person-X is represented by a first order bag of words created from a 55 word snippet of text that surrounds the entity. The authors experimented with incremental and agglomerative vector space models, as well as Kulback-Liebler divergence. They report that agglomerative clustering of the vectors fares better than the incremental clustering methods of Bagga and Baldwin [2]. In this case the number of clusters was not specified apriori, rather a threshold was set that determined if one chains was sufficiently similar to another to be grouped together.

In our earlier work we drew inspiration from cross–document co–reference resolution research, and in particular utilized first order context vectors to represent the context surrounding ambiguous names. While this sometimes works quite well, we often found that we did not have large enough numbers or sizes of context to obtain sufficient feature coverage to discriminate names reliably.

4 Name Discrimination Methodology

Over the last few years, we have developed an unsupervised approach to name discrimination, where our input is N contexts in which a single surface form of a given name occurs. Our goal is to group these contexts into k clusters, where the

value of k is automatically determined. Each discovered cluster is made up of contexts associated with a unique person, and the number of clusters indicates the number of distinct individuals that share the given name. Since this is name discrimination, each context is treated independently and there is no pre–existing sense inventory.

The evolution of this approach has been described in a series of publications that began at CICLing-2005 [14] and continued at IICAI-2005 [6] and CICLing-2006 [13]. In those papers, we explored different ways of constructing second–order context vectors, and the effect of Singular Value Decomposition (SVD) on the context representation prior to clustering. At this stage in the development of this work, we would manually set the number of clusters to be discovered to the proper value, or some arbitrary value.

This work continued to evolve, with an emphasis on comparing first and second–order feature representations, and studying the effect of SVD. In addition, we developed numerous methods to automatically identify the number of clusters/identities in a data set (c.f., [5], [10]). This work was reported on at an IJCAI-2007 workshop [11] and CICLing-2007 [12].

It must be said that it is impossible to provide a short summary of the different methods used in these five previous papers, since in each paper many different settings were employed for each word, and the best results per word were reported. Thus, there is no single method that emerged from those earlier papers as dominant, rather these were explorations of the capabilities of the SenseClusters system, and the range of possibilities for solving this problem in general. Our goal in this paper is to try and arrive at a more generic method of name discrimination which can hopefully be applied to a wide range of data with good effect.

One of the dominant themes in our work has been the use of second–order context representations. In this framework, the contexts in which ambiguous names occur are approximately 50 words wide. All of the bigrams in the contexts to be clustered that have a log–likelihood ratio or pointwise mutual information (PMI) score above a pre–determined threshold are identified as features. These bigrams are then used to construct a word by word matrix, where the first words in the bigrams represent the rows, and the second words in the bigrams represent the columns. Then, each context with an ambiguous name to be discriminated is re–processed, such that every word in that context that has a row entry in this word by word matrix is replaced by the vector formed by that row. This replacement step creates the second–order representation, where the words in the context are now represented not by the words that occur in that context, but by the words that occur with them in the contexts to be clustered. After all possible substitutions are made, the vectors are averaged and then the resulting vector becomes the representation of the context. Rather than bigrams, co–occurrences can be employed exactly as described above. The only difference is that since co–occurrences are not order dependent, the resulting word by word matrix that is created is symmetric.

We have also used first–order contexts, where unigram features are identified via frequency counts, and then the contexts are are represented simply by indicating which words have occurred surrounding the ambiguous name.

After the context representation has been created (whether it is first or second order), modified forms of k–means clustering are performed using the PK2 or the Adapted Gap Statistic method of cluster stopping. In either case clustering is performed on a range of values of k in order to determine which best fits the data, and automatically determine the number of underlying identities.

Each ambiguous name is is processed separately, and evaluation is done via the F-Measure, which finds the maximal agreement between the discovered clusters and the actual identities of the names. Note that this style of evaluation results in stiff penalties if the method predicts the wrong number of clusters.

5 Development Experiments

Our most recent work is reported on in papers at CICLing-2007 [12] and an IJCAI-07 workshop [11]. In those papers we experimented on the Kulkarni Name Corpus[1]. This consists of 1,375 manually disambiguated contexts, each of which are approximately 100 words wide. The center of each context includes one of five different ambiguous names as retrieved from the Web in May 2006. Over the five names a total of 14 different identities are represented, which results in an average of 2.8 identities per name. Note that there was is some variation in the surface forms of names in this data, where first initials or titles may also be used (e.g., Professor Miller, G. A. Miller, etc.)

The names, the number of distinct identities (I), the number of contexts (N) and the percentage of the most common identity per name (the Majority class) are shown in Table 1. Note that the Majority class corresponds to the F-measure that would be obtained by a simple baseline method that simply assigns all the contexts for a given name to a single cluster (in effect assuming that there is no ambiguity in the name).

As we reviewed our results from these 2007 papers, we noticed that some of the Web contexts were relatively impoverished and had very little text. Our features in these papers were based on using unigrams and first–order contexts, or adjacent bigrams and second–order contexts. We realized that for this data, first order representations of contexts might have little chance of success, since there is such a small number of features. While second–order features are sometimes seen as a solution to small data problems, there are limits to have effective they can be with very sparse or noisy feature sets.

Thus, we decided to try and increase the amount of context by using bigrams and co–occurrence features with very wide windows. This means that rather than requiring that a pair of words occur together to form a bigram (in order) or a co-occurrence (in either order), we would allow up to 48 intervening words to occur between them. This will increase the number of bigram or co–occurrence features available for second order methods, and we felt that might be a promising method

[1] http://www.d.umn.edu/~tpederse/namedata.html

Table 1. Development Results with Kulkarni Name Corpus

Name	I	N	Maj.	Best	(k)	New-Coc	(k)	New-Big	(k)
IJCAI-2007, CICLing-2007:									
Sarah Connor	2	150	72.7	90.0	(2)	79.0	(3)	**100.0**	(2)
Richard Alston	2	247	71.3	**99.6**	(2)	**99.6**	(2)	98.8	(2)
George Miller	3	286	**75.9**	**75.9**	(1)	61.2	(4)	61.9	(3)
Ted Pedersen	3	333	**76.6**	**76.6**	(1)	61.3	(3)	62.2	(3)
Michael Collins	4	359	74.9	93.0	(3)	88.9	(4)	**94.4**	(3)

of improving performance. In addition, we were concerned that our methods were a bit brittle. For example, we used the log–likelihood ration or Pointwise Mutual Information (PMI) to identify the bigram and co–occurrence features. However, in both cases the values that we use for determining which bigrams and co–occurrences are interesting will vary depending on the sample size, and so there isn't a clear way to make this process fully automatic. Finally, we also observed that the Adapted Gap Statistic, which we used for cluster stopping in the 2007 experiments, had a tendency to simply find one cluster, which resulted in F-measures that sometimes converged on the Majority class percentage.

Thus, we made a few modifications to our approach for this most recent round of experiments. First, we allowed the words in the bigram or co-occurrence to be separated by up to 48 intervening words, which greatly increases the number of bigrams that are considered as possible features. Second, we switched to using Fisher's Exact Test for identifying significant bigrams [9], which is relatively robust in the face of changing sample sizes, and allows for a single p–value (0.99) to be used for assessing significance. We also began to use the PK2 method of cluster stopping as our default method. We conducted an extensive series of experiments on the Kulkarni Name Corpus using these new ideas, and in general found some improvement in results. However, we wanted to make sure that we were not tuning the results too closely to this one particular data set, so after arriving at what appeared to be a robust and effective set of parameter settings[2] we evaluated those on data sets we had used in earlier name discrimination experiments.

Our final choices for the parameter settings based on both our intuitions and observed results on the Kulkarni Name Corpus as are follows:

1. A context of 50 words to the left and right of the ambiguous name is used both for feature selection and context representation.
2. Bigrams or co–occurrences may have up to 48 intervening words between them (a 50 word window), and are selected based on Fisher's Exact test (left-sided) with a p–value of 0.99.
3. Any bigram or co–occurrence that includes at least one stop word from the standard Ngram Statistics Package (version 1.09) stop list is discarded, and any bigram or co–occurrence that occurs less than 2 times is discarded.
4. SVD is not employed.

[2] These parameter settings refer to option values given to the SenseClusters package, which is the tool we have developed and used in all of our name discrimination work.

5. Cluster stopping is done with the PK2 method. This means that clustering is done with k–means for successive values of k, and the criterion function for clustering indicates at what value of k we should stop.

In our experiments with the Kulkarni Name Corpus we noticed fairly significant advantages to using bigrams rather the co-occurrences. This surprised us since the only difference between them is that bigrams are ordered pairs of words while co–occurrences occur in either order. Even though we observed better performance for bigrams, we decided to continue to study the difference between these two kinds of features (where everything else is held steady) in the held–out data as well.

The results of our experiments on the Kulkarni Name Corpus with the settings above are shown in Table 1. The high F-Measure in the 2007 papers for a given name is shown in the column labeled Best, and it's important to understand that this is the best result selected from a large number of different methods reported on in those two papers. Our goal in this paper is to arrive at a single set of parameter settings that will result in more consistent and accurate performance across a range of names. After the Best column we show the number of clusters (k) discovered by the Adapted Gap Statistic. We then show the F-Measure obtained using co–occurrence features (New-Coc) and bigram features (New-Big). Both of these values are followed by the number of clusters predicted by PK2.

In general we were pleased with the results on the development data set. For three of the names the results of the new settings are as good or better than the previous results (which are the best of over 60 different experimental settings for each word). However, the results for *Ted Pedersen* and *George Miller* were disappointing. In the case of TP this was because one of the identities was associated with e–commerce sites that had little textual content, and were essentially unrepresented by textual features. A similar effect was observed for GM, where the movie director identity had quite a number of low text contexts, and again it was difficult to represent that identity. However, in our development experiments we achieved an F-measure of 88.29 on TP using SVD, a window size of 10 for the bigrams, and a much smaller window of 5 words to the left and right of TP (rather than 50) when building our second order representation of the context. However, this combination of settings seemed to be uniquely effective with TP. In general GM very rarely rose above the value of the Majority class even in our development experiments.

6 Evaluation Data

After arriving at the New-Big and New-Coc parameter settings described above, we evaluated them on the datasets used our papers from CICLing-2005 [14], IICAI-2005 [6] and CICLing-2006 [13]. These datasets were kept out of the development process completely, and were only processed with our new sets of parameter settings (New-Big and New-Coc).

The data from 2005 and 2006 is not manually disambiguated, but rather is a more artificial form of data that is based on creating ambiguous names from relatively unambiguous names found in newspaper text. This is very much like

pseudo-words have been created for word sense disambiguation experiments, where for example all occurrences of the words *banana* and *door* are combined together to create the newly ambiguous word *banana–door*.

In creating the pseudo–name data, we tried to select names to conflate together that might have some relation to each other, in order to avoid obviously easy cases. Discriminating between two soccer players, for example, is probably more difficult than discriminating between a soccer player and an investment banker (due to the distinct contexts in which these names will occur).

Once the names to be conflated are determined, we select some number of contexts containing one of these names from the English GigaWord Corpus, which consists of newspaper text from the 1990's and 2000's. Then all the occurrences of the names to be conflated are replaced by a pseudo–name which is now ambiguous. For example, we identified all occurrences of the different forms of *Bill Clinton* and *Tony Blair*, and conflated those together into a newly ambiguous name that could mean the former US President or the former British Prime Minister. In creating the pseudo–names, we also controlled the frequency distribution of the individual names to provide a variety different experimental scenarios. The creation of the pseudo–name data was carried out with our *nameconflate* program[3].

The pseudo-names we used in previous studies are shown in Table 2. These are referred to via their abbreviations as used in the original papers, except for the IICAI-2005 data where we only referred to the original names. In that case we have introduced abbreviations. This table also shows the number of identities (I), the total number of contexts (N), and the percentage of the most common underlying identity (Majority class).

The details surrounding the creation of this data and the underlying identities can be found in the original publications, but but briefly the CICLing-2005 data includes pseudo-names with 2 underlying identities. These include two soccer players (Robek : David Beckham and Ronaldo), an ethnic group and a diplomat (JikRol : Tajik and Rolf Ekeus), two high–tech companies (MSIBM : Microsoft and IBM), two political leaders (MonSlo : Shimon Peres and Slobodon Milosovic), a nation and a nationality (JorGypt : Jordan and Egyptian), and two countries (JapAnce : Japan and France). Note that these are generally unambiguous, although *Jordan* no doubt includes some contexts referring to the basketball player *Michael Jordan* and *Ronaldo* is a relatively common name outside of professional soccer.

In the IICAI-2005 data there are 2 and 3-way ambiguities. The identities are indicated via the initials of a person name of the first few letters of a single entity name. The 2-identity pseudo-words include people occupying similar roles in the world : Tony Blair and Vladimir Putin; Serena Williams and Tiger Woods; and Sonia Gandhi and Leonid Kuchma. It also includes pairs of entities of the same type : Mexico and Uganda (countries) plus Microsoft and Compaq (high–tech companies). The three identity conflations are based on mixing names found in the 2–way distinctions: Tony Blair, Vladimir Putin, and Saddam Hussein; Mexico, Uganda, and India; and Microsoft, Compaq and Serena Williams.

[3] http://www.d.umn.edu/~tpederse/tools.html

Table 2. Evaluation Data Results

Name	I	N	Maj.	Best	New-Coc	(k)	New-Big	(k)
CICLing-2005:								
Robek	2	2,542	69.3	**85.9**	46.5	(13)	38.5	(14)
JikRol	2	4,073	73.7	96.2	99.0	(2)	**99.4**	(2)
MSIBM	2	5,807	58.6	**68.0**	57.4	(3)	58.0	(3)
MonSlo	2	13,734	56.0	96.6	**99.5**	(2)	**99.5**	(2)
JorGypt	2	46,431	53.9	**62.2**	58.8	(3)	54.8	(2)
JapAnce	2	231,069	51.4	51.1	51.4	(2)	**51.5**	(2)
IICAI-2005:								
SG-LK	2	222	50.5	91.0	83.4	(3)	**97.8**	(2)
SW-TW	2	599	51.4	69.0	**83.8**	(3)	69.0	(2)
Mi-Co	2	760	50.0	70.3	80.8	(3)	**83.0**	(3)
Me-Ug	2	2,512	50.0	60.1	62.8	(3)	**63.5**	(3)
TB-VP	2	3,224	55.5	96.2	**96.7**	(2)	**96.7**	(2)
Mi-Co-SW	3	1,140	33.3	56.6	**94.2**	(3)	66.3	(2)
Me-Ug-In	3	3,768	33.3	46.4	64.1	(4)	**64.6**	(4)
TB-VP-SH	3	4,272	41.9	75.7	**94.9**	(3)	72.4	(2)
CICLing-2006:								
Me-Ug	2	2,512	50.0	59.2	62.8	(3)	**63.5**	(3)
BC-TB	2	3,800	50.0	**81.0**	50.0	(3)	75.0	(3)
IBM-Mi	2	5,807	58.6	**63.7**	57.4	(3)	58.0	(3)
BC-TB-EB	3	5,700	33.3	47.9	55.7	(3)	**56.5**	(3)
Me-In-Ca-Pe	4	6,000	25.0	**28.8**	26.4	(2)	27.9	(3)

For the CICLing-2006 data we used similar identities to create 2, 3 and 4-way ambiguities, including : Bill Clinton and Tony Blair; Microsoft and IBM; Mexico and Uganda; Bill Clinton, Tony Blair, and Ehud; and Mexico, India, California, Peru.

7 Discussion and Future Work

In Table 2 we show the Best results from the original publications from 2005 or 2006, and then the results with our new settings of New-Coc and New-Big. Note that there is a very significant difference in the old and new methods; the old results (Best) required that the number of clusters be manually set to the known value, whereas in the new method this has been automatically predicted. Thus, in the (Best) results the value of I was given to the clustering algorithm and it simply found that number of clusters. In the new approach no such information is given, and the value of k is automatically discovered by the PK2 cluster stopping method. Thus, the new methods are in fact solving a somewhat more difficult problem, and are doing so with more success than the previous best methods.

For the 19 names shown in Table 2, the previous Best methods from 2005 and 2006 remain the most accurate for only six names. Only in the case of *Robek* did the automatic cluster stopping with PK2 go badly wrong. That may be due to the fact that *Ronaldo* was more ambiguous than originally anticipated (referring

to many more people than simply the well known soccer player who goes by that one name). In all other cases the automatic cluster stopping performed quite well, and predicted the number of identities exactly correctly or was off by at most one identity. This is a significant improvement over the previous results, and makes the methods much easier to apply on a wider range of data.

Of the 13 names that were most accurately discriminated by the very wide bigrams or co–occurrences, eight were best handled by the bigrams, three by the co–occurrences, and there were two ties between the bigrams and co–occurrences. While this doesn't provide overwhelming evidence that bigrams are always superior, it is an intriguing result that the order of the words matters even with very wide bigrams.

While we did not employ SVD in these experiments, it is clear from the *Ted Pedersen* results that there remain some cases where dimensionality reduction will be helpful. One of the main goals of our future work is to be able to automatically identify situations where SVD should or should not be applied.

SenseClusters also provides support for Latent Semantic Analysis [7], where we represent words in a context relative to the contexts in which they occur (rather than relative to the other words with which they occur, as is the case in this paper). In fact we included the LSA mode in our development experiments, and in general it did not fare well. However, we believe that there are still possible ways to formulate LSA for that name discrimination, as was shown by Levin, et al. [8].

Finally, our experiments have thus far been limited to small numbers of underlying identities. We plan to experiment with data like the John Smith data, where there are many individuals sharing that name.

8 Conclusions

We find that using very wide bigrams improves the results of unsupervised name discrimination, and that automatic cluster stopping can be employed to accurately identify the number of underlying identities.

Acknowledgments

All of the experiments in this paper were carried out with version 1.01 of the SenseClusters package, freely available from http://senseclusters.sourceforge.net. All of the data used in this paper is freely available from the author.

References

1. Artiles, J., Gonzalo, J., Sekine, S.: The SemEval-2007 WePS evaluation: Establishing a benchmark for the web people search task. In: Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval 2007), Prague, Czech Republic, June 2007, pp. 64–69. Association for Computational Linguistics (2007)
2. Bagga, A., Baldwin, B.: Entity–based cross–document co–referencing using the vector space model. In: Proceedings of the 17th International Conference on Computational Linguistics, pp. 79–85 (1998)

3. Gale, W., Church, K., Yarowsky, D.: One sense per discourse. In: Proceedings of the Fourth DARPA Speech and Natural Language Workshop (1992)
4. Gooi, C.H., Allan, J.: Cross-document coreference on a large scale corpus. In: HLT-NAACL 2004: Main Proceedings, Boston, Massachusetts, USA, May 2 - 7, pp. 9–16 (2004)
5. Kulkarni, A.: Unsupervised context discrimination and automatic cluster stopping. Master's thesis, University of Minnesota (July 2006)
6. Kulkarni, A., Pedersen, T.: Name discrimination and email clustering using unsupervised clustering and labeling of similar contexts. In: Proceedings of the Second Indian International Conference on Artificial Intelligence, Pune, India, December 2005, pp. 703–722 (2005)
7. Landauer, T., Dumais, S.: A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition, induction and representation of knowledge. Psychological Review 104, 211–240 (1997)
8. Levin, E., Sharifi, M., Ball, J.: Evaluation of utility of LSA for word sense discrimination. In: Proceedings of the Human Language Technology Conference of the NAACL, New York City, June 2006, pp. 77–80 (2006)
9. Pedersen, T., Kayaalp, M., Bruce, R.: Significant lexical relationships. In: Proceedings of the Thirteenth National Conference on Artificial Intelligence, Portland, OR, August 1996, pp. 455–460 (1996)
10. Pedersen, T., Kulkarni, A.: Selecting the right number of senses based on clustering criterion functions. In: Proceedings of the Posters and Demo Program of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics, Trento, Italy, April 2006, pp. 111–114 (2006)
11. Pedersen, T., Kulkarni, A.: Discovering identities in web contexts with unsupervised clustering. In: Proceedings of the IJCAI 2007 Workshop on Analytics for Noisy Unstructed Text Data, Hyderabad, India, January 2007, pp. 23–30 (2007)
12. Pedersen, T., Kulkarni, A.: Unsupervised discrimination of person names in web contexts. In: Proceedings of the Eighth International Conference on Intelligent Text Processing and Computational Linguistics, February 2007, pp. 299–310 (2007)
13. Pedersen, T., Kulkarni, A., Angheluta, R., Kozareva, Z., Solorio, T.: An unsupervised language independent method of name discrimination using second order co-occurrence features. In: Proceedings of the Seventh International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, February 2006, pp. 208–222 (2006)
14. Pedersen, T., Purandare, A., Kulkarni, A.: Name discrimination by clustering similar contexts. In: Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, February 2005, pp. 220–231 (2005)