Cross document person name disambiguation using entity profiles

Harish Srinivasan and **John Chen** and **Rohini Srihari**
Janya Inc.
1408 Sweet Home Road, Suite 1 Amherst, NY 14228
{hsrinivasan,jchen,rohini}@janyainc.com

Abstract

Given an ambiguous person name as input, a cross-document person name disambiguation system clusters documents so that each cluster contains all and only those documents referring to the same person. In this paper we present our approach to this task. We introduce novel features based on topic models and also document-level entity profiles—sets of information that are collected for each ambiguous person in the entire document. We also introduce a modified term frequency-inverse document frequency (TF-IDF) weighting scheme to represent entities in a vector-space model (VSM). Disambiguation is then performed via single-link hierarchical agglomerative clustering. Experiments show that an average F-measure of 94.03% is achieved using our proposed enhanced VSM model. This is an improvement over previous best results on the same test corpora.

Introduction

Cross-document entity coreference resolution refers to linking entities in one document to the same entities in others. A significant problem for this task is person name disambiguation; the same name may refer to different people. Lately, this has received a lot of interest in the research community. For example, in the 2007 SemEval workshop, a competition for disambiguating names from web search was included (Artiles, Gonzalo, and Sekine 2007). In this work, we also focus on this task. Previous work examined the use of 'context' (surrounding words in which names occur). Here, we examine the use of other contexts such as those gathered from topic models as well as from entity profiles and show that they are useful.

Previous work

The task of person name disambiguation has received attention only in the last decade. (Bagga and Baldwin 1998) were the first to address this problem and used a simple VSM model. (Mann and Yarowsky 2003) as well as most of the other previous work such as (Gooi and Allan 2004), (Malin 2005) and (Chen and Martin 2007) have employed unsupervised learning approaches. (Malin 2005) considers the named-entity disambiguation as a graph problem and constructs a social network graph to learn the similarity matrix.

(Chen and Martin 2007) used a combination of lexical context features and information extraction (IE) results and obtained superior performance to then previous published results. They use the following features in a Vector Space Model(VSM) - (i) **Summary terms**: Each non-stop word appearing within a fixed window around any mention of the entity (Bagga and Baldwin 1998) (ii) **Base Noun Phrases (BNP):** All tokens (unit of words/phrase in the document as processed by an IE engine) that are non-recursive noun phrases in the sentences containing the ambiguous name (or a coreference) and (iii) **Document Entities (DE):** All tokens that are named entities (Person other than the ambiguous name, Organization name, Location etc. as well as their nominals) in the entire document.

Our model builds on top of those features employed by (Chen and Martin 2007). With our enhancements and modifications, we obtain an improved performance over all previously published results on the same corpora.

System Background

Our person name disambiguation system is built on top of our IE system (Srihari et al. 2006). The latter may be described as follows. Documents are processed one at a time. The results of such processing include entities, relations, and events as well as syntactic information including base noun phrases and syntactic and semantic dependencies. Named entity and nominal entity mentions are recognized using maximum entropy Markov models. Subsequently, coreference resolution is performed on these mentions as well as any pronouns in the document according to a pairwise entity coreference resolution module.

In our system, we maintain what could be described as an *entity-oriented* model. The key objects in the model are *entity profiles*, which combine in one place features of the entity, attributes of the entity (links from the entity to a *value*, rather than another entity), relations (to or from another entity), and events that this entity is involved in as a participant. The result of processing a document is a collection of *document-level* entity profiles, which represent all of the information associated with any mention of that entity in the document. In this setting, we characterize the problem of cross document person name disambiguation and informa-

Source	Ambiguous Name	Entity Mention Sentence	Profile	Entities
Document 1	*John Smith*	*John Smith, a U.K. equities strategist at Henry Cooke Lumsden, said the acquisition makes Royal Bank the most likely bank share to make gains in coming days.*	*equities , strategist*	*U.K., Henry Cooke Lumsden, Royal Bank*
Document 2	*John Smith*	*U.K. interest rates are likely to rise once more next year, probably no more than 25 basis points, said John Smith, an equity strategist at Henry Cooke Lumsden.*	*equity , strategist*	*U.K., Henry Cooke Lumsden*

Table 1: Example of two document level entity profiles that need to be merged

tion consolidation as specifically the problem of merging document level entity profiles into corpus level entity profiles. Table 1 shows two document level entity profiles that need to be merged.

Model

We employ a Vector Space Model (VSM) to represent the document level entities. The VSM considers the words (terms) in a given document as a 'bag of words'. (Chen and Martin 2007) employ separate 'bag of words' for each of the three features (Summary terms, Base Noun Phrases and Document Entities) and use a Soft Tf-Idf weighting scheme with cosine similarity to evaluate the similarity between two entities. The similarities computed from each feature are averaged to obtain a final similarity value. Below we describe our modification and enhancements to Chen and Martin's model.

1. **Single bag of words model:** We employ a single bag of words model (rather than separate bag of words used by Chen and Martin). The motivation was that to allow terms from one bag of words (say summary sentence terms) to match the terms from another bag of words (say DE-document entities).

2. **Profile features (PF):** As described in the previous section, our IE system constructs entity profiles, consolidating all of the information that our system finds associated with a particular entity in a particular document. All of the features in a profile are extracted and stored as attribute-value (two tuple) pairs. The value term in the tuple is then appended to the 'bag of phrases and words'. Table 2 given an example of a profile for an entity named 'John Smith' as extracted by our IE engine.

   Because they are extracted from the same input document, there will often be overlap between profile features and features of other types. Consider the input sentence "Captain John Smith first beheld American strawberries in Virginia." Here, the feature "Captain" is both a Summary term and a profile feature. Still, profile features are useful because they highlight critical entity information. In this example, "Captain" is highlighted because it is a person title. In contrast, "strawberries" would be a Summary term feature but not a profile feature.

Attribute	**Value**
PRF_NAM	John Smith
CE_MODIFIERS	Still alive
EVENTS_INVOLVED	Ran into
CE_PER_TITLE	Captain
Ce_Assocication_Entity	Joe Grahame

Table 3: Example of document level entity profile

3. **Topic Model Features (TM):** It was observed that certain pairs of documents had no common terms in their feature space even though, they contained similar terms such as 'island, bay, water, ship' in one document and 'founder, voyage, and captain' in another document. A naive string matching (VSM model) fails to match these terms. Hence, an expansion of the common noun words in a document was attempted using topic modeling (Blei, Ng, and Jordan 2003). Every document is assigned a possible set of topics and every topic is associated with a list of most common words. The following steps were performed to use features from topic model.

   (a) The words that were used to learn the topic model were all the nouns in the document along with the terms in the summary sentence. Hence, for each corpus a different topic model was learned due to the difference in the input (words) to the topic model learning algorithm.

   (b) The number of topics to learn was set at 50. Once the topic model was learned for each document, the top 10 words with highest *joint probability of word in topic and topic in a document* were chosen. This probability corresponds to the joint probability of word and topic in a document. $P(w,t|D) = P(w|t,D) \times P(t|D) = P(w|t) \times P(t|D)$, where w, t and D are word, topic and document respectively. The last equality in the expression is due to conditional independence of the word and the document, given the topic. The numbers of topics (50) and words from topic (10) were chosen by using one corpus as a validation set.

   (c) These 10 topic model words are then appended to the existing bag of words and phrases.

4. **Name as a stop word (Nsw):** The ambiguous name in question was included in the stop word list. This is intuitive since the name itself provides no information in re-

Ambiguous Name / **Total No of Documents**	John Smith(Bagga) 197	James Jones 104	John Smith(Boulder) 112	Michael Johnson 101	Robert Smith 100	Average
No Of Clusters	35	24	54	52	65	
Chen and Martin - Optimal Threshold - S+BNP+DE (Separate bag of words + Soft TF-IDF)	92.02	97.10(28)	91.94(61)	92.55(51)	93.48(78)	93.41
Chen and Martin - Fixed Stop Threshold - S+BNP+DE (Separate bag of words + Soft TF-IDF)	-	*96.64*	*91.31*(dev)	*90.57*(dev)	*86.71*	*91.31*
Baseline - S+BNP+DE (Separate bag of words)	84.20(48)	98.11(25)	85.50(62)	90.79(61)	90.37(79)	89.79
Baseline + Log Transformed	93.96(42)	90.54(33)	86.80(71)	89.52(67)	92.66(73)	90.69
Model (Single bag of words + Log Transformed Tf-Idf)						
S+BNP+DE	92.28(50)	95.48(26)	89.50(69)	91.64(49)	92.42(72)	92.26
S+BNP+DE + PF ($\mathcal{A}$)	91.93(47)	98.14(25)	91.46(65)	90.22(57)	92.54(77)	92.85
$\mathcal{A}$ + Nsw	92.77(49)	98.14(25)	90.56(67)	89.85(62)	93.22(70)	92.90
$\mathcal{A}$ + Nsw + Ptf	92.83(49)	98.14(25)	91.24(68)	93.27(55)	94.27(73)	93.95
$\mathcal{A}$ + Nsw + Ptf + TM	**92.62(42)**	**99.03(26)**	**91.49(67)**	**94.01(56)**	**93.03(76)**	**94.03**
$\mathcal{A}$ + Nsw + Ptf + TM (Fixed Stop Threshold)	*92.42(48)*	*97.28(24)*	*89.3(59)*(dev)	*90.3(62)*(dev)	*92.12(70)*	*92.28*
Model (Separate bag of words with all features)						
Separate - Average	93.01(43)	98.37(27)	81.65(62)	87.34(60)	92.27(69)	90.52
Separate - NN	94.40(43)	99.03 (26)	86.26 (73)	89.19 (66)	90.98 (69)	91.97
Separate - MaxEnt	92.69 (48)	98.14(25)	86.94(65)	88.92 (63)	91.71 (70)	91.68

Table 2: F-measure performance. 'S'-Summary terms, 'PF'-Profile Features, 'BNP'-Base Noun Phrases, 'DE'-Document Entities, $\mathcal{A}$-All features (S+PF+BNP+DE), 'Nsw'-After including the ambiguous Name as a Stop Word, 'Ptf'-Using Prefix matching for calculating Term Frequency, 'TM'-Topic model features. In parenthesis are the number of clusters. In all of the measures, the single linkage hierarchical clustering was employed. To compute the fixed stop threshold, the mean optimal threshold of the 'John Smith (Boulder)' and 'Michael Johnson' were used.

solving the ambiguity as it is present in all the documents.

5. **Prefix matched term frequency (Ptf):** When calculating the term frequency of a particular term in a document, a prefix match was used. e.g. If the term was 'captain', and even if only 'capt' was present in the document, it is counted towards the term frequency. This modification allows for the possibility of correctly matching commonly used abbreviated words with the corresponding non-abbreviated words.
6. **Log-Transformed Tf-Idf weighting:** The Tf-Idf formulation as used by Bagga and Baldwin is given in Equation 1.

$$Sim(S_1, S_2) = \sum_{\text{common terms } t_j} w_{1j} \times w_{2j},$$

$$\text{where} \quad w_{ij} = \frac{\text{tf} \times \ln \frac{N}{\text{df}}}{\sqrt{s_{i1}^2 + s_{i2}^2 + \ldots + s_{in}^2}} \quad (1)$$

where S_1 and S_2 are the term vectors for which the similarity is to be computed. tf is the frequency of the term t_j in the vector. N is the total number of documents. df is the number of documents in the collection that the term t_j occurs in. The denominator is the cosine normalization.

$$Sim(S_1, S_2) = \sum_{\text{common terms } t_j} w_{1j} \times w_{2j},$$

$$\text{where} \quad w_{ij} = \frac{\ln\left(\text{tf} \times \ln \frac{N}{\text{df}}\right)}{\sqrt{s_{i1}^2 + s_{i2}^2 + \ldots + s_{in}^2}} \quad (2)$$

Our modification to this formulation is given in Equation 2. These weights are then used to calculate the similarity values between document pairs. In error analysis it was observed that, several document pairs had low similarity values despite belonging to the same cluster. If one were to use a threshold to decide on the decision to merge clusters, the log transformation will have had no effect (since the transformation is a monotonic function). But in the case of hierarchical agglomerative clustering using single linkage, this transformation helps alleviate the problem by relatively better spacing out those ambiguous document pairs (those that had low similarity scores).

Experiments and Results

For each person name that we want to disambiguate, a hierarchical agglomerative clustering algorithm using single linkage is run across vectors representing documents. Two sets of corpora were used for performing experimental evaluations - (i) Bagga Baldwin corpus (Bagga and Baldwin 1998) containing one ambiguous name and (ii) English Boulder name corpora containing four sub corpora each corresponding to one of four different ambiguous names. These together gave a total of five different corpora each one containing a ambiguous name. The first part of Table 2 summarizes the characteristics of each of the five different corpora.

The second part of Table 2 shows a complete set of results with a breakdown of the contribution of features as they are added into the complete set. First we show a baseline performance that uses the same set of features as that used by Chen and Martin's best model. The baseline model uses a three separate bag of words model, one for each of Summary terms, document entities and base noun phrases. The three resulting similarity values are then combined using plain av-

erage. The difference between the results reported by Chen and Martin and our baseline mode is due to the differences in the IE engine used, the list of stop words, and Chen and Martin's use of Soft TF-IDF weighting scheme.

The remaining rows of Table 2 uses the log transformed tf-idf weighting scheme. Result show that this weighting scheme and also the addition of other features and enhancements contribute significantly to improve the performance from the baseline. Also, our best model outperforms (in average F-measure) that reported by Chen and Martin for both optimal and fixed stop threshold. For the sake of completeness we also present results from learning the separate bag of words model with our complete feature set. We experimented with three different ways of combining the similarities from the individual features (i) Plain average, (ii) Neural network weighting and (iii) Maximum Entropy weighting. The lower performance for these justify the use of a single bag of words model.

Conclusions

Person name disambiguation is critical for consolidating entity information across an entire corpus. The extensions and enhancements to the VSM model described in this paper show an improvement in F-measure over previously published results. These included treatment of different feature types as a single bag of words, use of prefix-matched term frequency, log-transformation of the Tf-Idf score, introduction of entity profile features, addition of features based on topic models, and use of name as a stopword. In the future, we plan to tackle the problem of entity disambiguation in combination with that of alias detection.

Acknowledgments

This work was partly supported by SBIR grant FA8750-08-C-0044 from the Air Force Research Laboratory (AFRL)/RIED, Rome NY.

References

Artiles, J.; Gonzalo, J.; and Sekine, S. 2007. The semeval-2007 weps evaluation: Establishing a benchmark for the web people search task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*.

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

Blei, D.; Ng, A.; and Jordan, M. 2003. Latent dirichlet allocation. In *Journal of machine learning research*, volume 3, 993–1022.

Chen, Y., and Martin, J. 2007. Towards robust unsupervised personal name disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 190–198.

Gooi, C. H., and Allan, J. 2004. Cross-document coreference on a large scale corpus. In *HLT-NAACL*, 9–16.

Malin, B. 2005. Unsupervised name disambiguation via social network similarity. In *Workshop on Link Analysis, Counterterrorism, and Security in conjunction with the SIAM International Conference on Data Mining*, 93–102.

Mann, G. S., and Yarowsky, D. 2003. Unsupervised personal name disambiguation. In Daelemans, W., and Osborne, M., eds., *Proceedings of CoNLL-2003*, 33–40. Edmonton, Canada.

Srihari, R. K.; Li, W.; Cornell, T.; and Niu, C. 2006. Infoxtract: A customizable intermediate level information extraction engine. *Natural Language Engineering* 12.