UC3M_13: Disambiguation of Person Names Based on the Composition of Simple Bags of Typed Terms

David del Valle-Agudo **César de Pablo-Sánchez** **María Teresa Vicente-Díez**
Universidad Carlos III de Madrid
Escuela Politécnica Superior
Av. de la Universidad, 30 – 28911
Leganés (Madrid) Spain
{dvalle, cdepablo, tvicente}@inf.uc3m.es

Abstract

This paper describes a system designed to disambiguate person names in a set of Web pages. In our approach Web documents are represented as different sets of features or terms of different types (bag of words, URLs, names and numbers). We apply Agglomerative Vector Space clustering that uses the similarity between pairs of analogous feature sets. This system achieved a value of 66% for $F_{\alpha=0.2}$ and a value of 48% for $F_{\alpha=0.5}$ in the Web People Search Task at SemEval-2007 (Artiles et al., 2007).

1 Introduction

Name queries account for a substantial part of Web queries in commercial search engines. Name queries often aim at retrieving information about particular persons. Nevertheless, the same query or mention name usually recalls several people and the user is unaware of the potential ambiguity and expects to find the related person after skimming some results.

Similar problems are also common for products, organizations and almost any other named object in real world. A related problem appears for different kinds of objects receiving the same name. For example, Firebird can refer to a car, a guitar, a fiction superhero or a database product among more than twenty different senses collected in Wikipedia. In all these cases, the user could benefit from a structured representation that facilitates browsing results. Other applications like Question Answering would also benefit from name disambiguation and person names disambiguation, in particular. In this work we focus on the task of disambiguating Web pages retrieved for a person name query as proposed in the Web People Search Task at SemEval-2007.

2 Background and Related Research

In recent work in named entity disambiguation, Malin (2005) identifies two different dimensions to classify approaches to the task depending on the information type that is used and whether the method to train the system is supervised or unsupervised. Regarding the information type, Malin (2006) identifies personal information like biographical facts (Bagga and Baldwin, 1998; Mann and Yarowsky, 2003) or relational information (Bekkerman and McCallum, 05), collocations with other entities.

Personal name disambiguation has been studied in relation with citation analysis and record linkage and their use to improve Web search results have attracted more interest recently (Guha and Garg 2004; Bollegala, 2006), but it is evaluated only at a small scale. In contrast Bekkerman and McCallum (2005) have focused on disambiguating complete social networks and not only results for one name.

3 System description

Web People Search proposes a task to find different people sharing the same name referred in a set of Web pages and associate each of these pages to these people. To solve the task we added two simplifying assumptions; each document refers only to one person, and every listed document refers to a person.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 362–365,
Prague, June 2007. ©2007 Association for Computational Linguistics

Our approach is an unsupervised personal name disambiguation system according to the classification proposed by Malin. In this system the method applied to solve ambiguity consists of extracting from each document a set of features, that we called *document context* and afterwards to cluster them according to their similarity

3.1 Document representation

In this task we do not have structured information to estimate similarity. For this reason, the first step of the system consists of extracting features from the documents. Since our goal is to develop techniques that work for large amounts of documents, most of the features are based simply on words, HTML structure and simple patterns that aim to substitute more elaborated features based on information extraction. Features might not have a direct correspondence with facts that help to identify a person like *date of birth* or *telephone* but, in some cases, dealing with them instead of with proper semantic information can be a good approach. On the other hand, some people features, as emails or related URLs, are detected through simple patterns. Other simple patterns like numbers can also provide information about some people features.

All terms identified by the same pattern are represented as a bag of terms. Document context is composed of a set of bags, each containing all the terms of the document that were captured with a fixed pattern.

3.2 Types of Contexts

The bags of terms used in document contexts are the following:

a) emails, b) URLs, c) proper names, d) long numbers (more than four figures), e) short numbers (up to four figures), f) title terms, g) terms of the titles of related documents, h) terms contained in the '*meta*' tag of the documents, i) terms of emphasized text fragments (bold, italic, etc.), j) terms of the document snippet, and k) terms of the related documents snippets.

The bags b, f, g, j, and k have been extracted from the data files provided (snippets, rank, etc.), whereas a, c, d, e, h and i have been directly extracted from result pages.

From all the bags of terms, we finally selected to compound the contexts b, c, d, e, f, g and j as in the training set they contributed to obtain the best result.

3.3 Term normalization and filtering

Each extracted term is normalized, filtered and weighted before being added to a bag of terms. A filter for stopwords is applied to every bag of words and they are represented in lowercase. Spurious HTML tags and terms under three characters are also considered stopwords. Bag of numbers are normalized by removing blanks, hyphens and parenthesis.

In addition to stopwords, terms with low frequency, lower than 0.2 times the frequency of the more frequent term of each bag of words, are not considered. Finally the *tf-idf* value of every term is associated.

Proper names are extracted with a robust rule based name recognizer based on surface feature and some trigger words. It should be emphasized that over the bag of proper names, a filtering is implemented to make the detection of co-referents proper names easier when comparing different arrays. In this way, a similarity measure among proper names is considered (Camps and Daudé, 2003) more flexible than the simple comparison of their strings of characters. This approach tolerates the omission, substitution or inclusion of words in the proper name, the alteration in the order of the words, or the substitution of words with initials, as well as the omission, substitution or inclusion of characters. First, all proper names that are in the set of documents are identified, and all similar proper names according to these relaxed rules are grouped by the same common term. In this way, arrays of proper names are rewritten, referencing each proper name through its common term and recalculating its frequency.

3.4 Clustering algorithm

Our system uses *Agglomerative Vector Space Clustering* to group and disambiguate pages. Given the nature of the problem, it does not need to indicate the number of classes to be obtained in advance. To determine if two documents should be assigned to the same cluster, we evaluate the similarity between each pair of bags of terms and, later, it is determined how many of these pairs have a similarity over a threshold. For a document to be in the same cluster we require a minimum number of similar pairs.

In order to allow finer adjustments in the number of similar pairs needed, instead of requiring N similar pairs, the pairs are arranged in a decreasing order according to the obtained similarity and it is checked if the similarity of the nth pair is above or below the threshold. In this case, interpolation can be applied, so the number of necessary similar pairs is not limited to the natural numbers. The developed system uses linear interpolation to calculate this function.

We use the cosine vector similarity as similarity measurement.

4 Results and Evaluation

For the evaluation the system has been adjusted with a threshold of similarity of 0.001, 2.5 pairs of bags of terms above the threshold required for including two documents in the same cluster and the following bags of terms: bags of URLs, proper names, long and short numbers, terms of titles, terms of the titles of the related documents and terms of the document snippets.

With this adjustment it is noticed that some problems affect the results of the evaluation. The most important of these problems is the small number of clusters in which pages are classified. For instance, Mark Johnson refers to 70 different people in key, but our system classified his pages in only 14 clusters. Due to this small number of clusters, each contains more than one person to search, but with a good recall of pages for each person. Table 1 shows the results obtained for the test set, where P is the purity, R is the inverse purity, $F_{\alpha=0.5}$ represents the harmonic mean of purity and inverse purity, and $F_{\alpha=0.2}$ is the measure of F that considers more important inverse purity than purity.

Although at a first sight set 1 shows better results than set 2 and 3, once we discard the people names 'Sharon Goldwater' and 'Dekang Lin' (whose results are above the mean), results are very similar for all groups. We consider that our system behaves in a homogenous way regardless of the proportion of the different types of names the sets are composed of: less frequent names (with lower ambiguity) and 'celebrity' names (with people that dominate the set of pages).

In the other hand, the assumptions considered to solve the problem (each page references at least one and only one person) were definitely too naïve, as there is a lot of discarded pages (in some cases more than 50% of the pages are not taken into account) and some pages refer to several people. These facts also contribute to make lower purity.

Table 1. Test results (in percentages)

		P	R	$F_{\alpha=.5}$	$F_{\alpha=.2}$
Set 1	**Mark Johnson**	20	98	33	54
	Sharon Goldwater	99	99	99	99
	Robert Moore	26	94	40	61
	Leon Barrett	34	97	50	70
	Dekang Lin	100	98	99	98
	Stephen Clark	21	98	34	56
	Frank Keller	25	90	39	59
	Jerry Hobbs	52	92	67	80
	James Curran	24	98	39	61
	Chris Brockett	68	97	80	89
Set 2	**Thomas Fraser**	33	96	49	70
	John Nelson	24	96	38	60
	James Hamilton	19	99	32	54
	William Dickson	20	99	33	55
	James Morehead	26	96	41	62
	Patrick Killen	55	99	71	86
	George Foster	35	94	51	70
	James Davidson	25	98	39	61
	Arthur Morgan	54	98	70	84
	Thomas Kirk	11	98	20	39
Set 3	**Harry Hughes**	36	79	50	64
	Jude Brown	25	91	39	59
	Stephan Johnson	57	92	70	82
	Marcy Jackson	32	95	48	68
	Karen Peterson	12	99	21	40
	Neil Clark	46	98	62	80
	Jonathan Brooks	21	95	35	56
	Violet Howard	15	88	26	45
	Martha Edwards	11	96	20	38
	Alvin Cooper	34	95	50	70
	Set 1 Average	47	96	58	73
	Set 2 Average	30	97	44	64
	Set 3 Average	29	93	42	60
	Global Average	**35**	**95**	**48**	**66**

5 Conclusions and future works

This system obtains a good result for inverse purity to the detriment of purity. This causes a difference of almost twenty points in the measures of $F_{\alpha=0.5}$ and $F_{\alpha=0.2}$. In order to correct this weakness, in the future we will consider that any person can be mentioned in different pages, and that not all pages reference to any of the people to search.

Also we will perform additional experiments regarding parameter tuning. Although the number of similar contexts considered in these experiments

is 1.5 (value that maximizes the measure of F), results show that this value causes larger groups than those found in search results. Probably a smaller value for this parameter will divide pages in more clusters, improving the purity of the result.

Finally, we would like to consider different methods to select relevant terms.

References

A. Bagga and B. Baldwin. 1998. *Entity-based cross-document coreferencing using the vector space model.* In Proc 36th Annual Meeting of the Association for Computational Linguistics. San Francisco, CA.; 79-85.

Artiles, J., Gonzalo, J. and Sekine, S. (2007). *Establishing a benchmark for the Web People Search Task: The Semeval 2007 WePS Track*. In Proceedings of Semeval 2007, Association for Computational Linguistics.

Bradley Malin. 2005. *Unsupervised name disambiguation via social network similarity*. In Proceedings of the Workshop on Link Analysis, Counterterrorism, and Security, in conjunction with the SIAM International Conference on Data Mining. Newport Beach, CA; 93-102.

Camps, R., Daudé, J. 2003. *Improving the efficacy of aproximate personal name matching.* NLDB'03. 8th International Conference on Applications of Natural langage to Informations Systems.

Danushka Bollegala, Yutaka Matsuo and Mitsuru Ishizuka. 2006. *Disambiguating Personal Names on the Web using Automatically Extracted Key Phrases*. Proceedings of the European Community of Artificial Intelligence (ECAI 2006), Italy

G. Mann and D. Yarowsky. 2003. *Unsupervised personal name disambiguation*. In Proc 7th Conference on Computational Natural Language Learning. Edmonton, Canada.

Ramanathan V. Guha and A. Garg. 2004. *Disambiguating people in search*. In WWW2004.

Ron Bekkerman, Andrew McCallum. 2005. *Disambiguating Web appearances of people in a social network*. Proceedings of the 14th international conference on World Wide Web 2005. Pages 463 - 470.