AUG: A combined classification and clustering approach for web people disambiguation

Els Lefever and **Véronique Hoste**
LT3 Language and Translation Technology
Ghent University Association
Groot-Brittanniëlaan 45, 9000 Gent
els.lefever@hogent.be
veronique.hoste@hogent.be

Timur Fayruzov
Computational Web Intelligence
Ghent University Association
Krijgslaan 281, 9000 Gent
Timur.Fayruzov@UGent.be

Abstract

This paper presents a combined supervised and unsupervised approach for multi-document person name disambiguation. Based on feature vectors reflecting pairwise comparisons between web pages, a classification algorithm provides linking information about document pairs, which leads to initial clusters. In addition, two different clustering algorithms are fed with matrices of weighted keywords. In a final step the "seed" clusters are combined with the results of the clustering algorithms. Results on the validation data show that a combined classification and clustering approach doesn't always compare favorably to those obtained by the different algorithms separately.

1 Introduction

Finding information about people on the World Wide Web is one of the most popular activities of Internet users. Given the high ambiguity of person names and the increasing amount of information on the web, it becomes very important to organize this large amount of information into meaningful clusters referring each to one single individual.

The problem of resolving name ambiguity on the Internet has been approached from different angles. Mann and Yarowsky (2003) have proposed a Web based clustering technique relying on a feature space combining biographic facts and associated names, whereas Bagga and Baldwin (1998) have looked for coreference chains within each document, take the context of these chains for creating summaries about each entity and convert these summaries into a bag of words. Documents get clustered using the standard vector space model. Other researchers have taken this search for distinctive keywords one step further and tried to come up with "concepts" describing the documents. Fleischman and Hovy (2004) introduce the "maximum entropy model": a binary classifier determines whether two concept-instance pairs refer to the same individual. Pedersen (2006) presented an unsupervised approach using bigrams in the contexts to be clustered, thus aiming at a concept level semantic space instead of a word level feature space.

For the semeval contest, we approached the task from a double supervised and unsupervised perspective. For the supervised classification, the task was redefined in the form of feature vectors containing disambiguating information on pairs of documents. In addition to this, different clustering approaches were applied on matrices of keywords. These results were then merged by taking the classification output as basic "seed" clusters, which were then enhanced by the results from the clustering experiments.

In the remainder of this paper, Section 2 introduces the data sets and describes the construction of the feature vectors and the keyword matrices. The classification and clustering experiments, and the final combination of the different outputs are discussed in Section 3. Section 4 gives an overview of the results on the test data and Section 5 summarizes the main findings of the paper.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 105–108,
Prague, June 2007.

2 Data sets and feature construction

The data we have used for training our system were made available in the framework of the SemEval (task 13: Web People Search) competition (Artiles et al., 2007). As preliminary training corpus (referred to as "trial data" in our article), we used the WePS corpus (Web People Search corpus), available at http://nlp.uned.es/weps. For the real training set, this trial set was expanded in order to cover different degrees of ambiguity (very common names, uncommon names and celebrity names which tend to monopolize search results). The training corpus is composed of 40 sets of 100 web pages, each set corresponding to the first 100 results for a person name query. The documents were manually clustered. Documents that couldn't be clustered properly have been put in a "discarded" section. Test data have been constructed in a similar way (30 sets of 100 web pages).

The content of the web pages has been preprocessed by means of a memory-based shallow parser (MBSP) (Daelemans and van den Bosch, 2005). From the MBSP, we used the regular expression based tokenizer, the part-of-speech tagger and text chunker using the memory-based tagger MBT. On the basis of the preprocessed data we construct a rich feature space that combines biographic facts and distinctive characteristics for a given person, a list of weighted keywords and meta data information about the web page.

2.1 Feature vector construction

The following biographic facts and related named entities were extracted from the preprocessed data. Information on date and place of birth, and on date and place of death were extracted by means of a rule-based component. Furthermore, three named entity features were extracted on the basis of the shallow syntactic information provided by the memory-based shallow parser and additional gazetteer information. Furthermore, a "name" feature was aimed at the extraction of further interesting name information (E.g other surnames, family names) on the person in focus, leading to the extraction of for example "Ann Hill Carter Lee" and "Jo Ann Hill" for the document collection on "Ann Hill". The "location" feature informs on the overlap between all locations named in the different documents. In a similar way, the "NE" feature returns the inter-document overlap between all other named entities.

Starting with the assumption that overlapping URL and email addresses usually point to the same individual, we have also extracted URL, email and domain addresses from the web pages. Therefore we have combined pattern matching rules and markup information (HTML <href> tag). The link of the document itself has been added to the set of URL links. Some filtering on the list has been performed concerning length (to exclude garbage) and content (to exclude non-distinctive URL addresses such as index.html). Pair-wise comparison of documents with respect to overlapping URL, email and domain names resulted in 3 binary features.

Another binary feature we have extracted is the location, based on our simple supposition that if two documents are hosted in the same city, they most probably refer to the same person (but not vice versa). For converting IP-addresses to city locations, we have used MaxMind GeoIP(tm) open source database[2], which was sufficient for our needs.

2.2 A bag of weighted keywords

The input source for extracting our distinctive keywords is double: both the entire (preprocessed) content of the web pages as well as snippets and titles of documents are used. Keywords extracted from snippets and titles get a predefined -rather high- score, as we consider them quite important. For determining the keyword relevance of the words extracted from the content of the web pages, we have applied Term Frequency Inverse Document Frequency (TF-IDF) (Berger et al., 2000).

Once all scores are calculated, all weighted keywords get stored in a matrix, which serve as input for the clustering experiments. The calculated keyword weight is also used, in case of overlapping keywords, as a feature in our pairwise comparison vector. In case two keywords occurring in two different documents are identical or recognized as synonyms (information we obtain by using WordNet[3]), we sum up the different weights of these keywords and store this value in the feature vector.

[2] http://www.maxmind.com/app/geolitecity

[3] http://wordnet.princeton.edu/

3 Classification and Clustering algorithms

3.1 Classification

For the classification experiments, we used the eager RIPPER rule learner (Cohen, 1995) which induces a set of easily understandable if-then classification rules for the minority class and a default rule for the remaining class. The ruler learner was trained and validated on the trial and training data. Given the completely different class distribution of the trial and training data, viz. 10.6% positive instances in the trial data versus 66.7% in the training data, we decided to omit the trial data and optimize the learner on the basis of the more balanced training data set. There was an optimization of the class ordering parameter, the two-valued negative tests parameter, the hypothesis simplification parameter, the example coverage parameter, the parameter expressing the number of optimization passes and the loss ratio parameter. The predicted positive pairwise classifications were then combined using a for coreference resolution developed counting mechanism (Hoste, 2005).

3.2 Clustering Algorithms

We experimented with several clustering algorithms and settings on the trial and training data to decide on our list of parameter settings. We validated the following three clustering algorithms. First, we compared output from k-means and hierarchical clustering algorithms. Next to that, we have run experiments for agglomerative clustering[4]. with different parameter combinations (2 similarity measures and 5 clustering functions). All clustering experiments take the weighted keywords matrix as input. Based on the validation experiments, hierarchical and agglomerative clustering were further evaluated to find out the optimal parameter settings. For hierarchical clustering, this led to the choice of the cosine distance metric, single-link hierarchical clustering and a 50% cluster size. For agglomerative clustering, clustering accuracy was very dependent on the structure of the document set. This has made us use different strategies for clustering sets containing "famous" and "non famous" people. As a distinction criterion we have chosen the presence/non-presence of the person in Wikipedia. We started with the assumption that sets containing famous people (found in Wikipedia) most probably contain a small amount of bigger clusters than sets describing "ordinary" persons. According to this assumption, two different parameter sets were used for clustering. For Wikipedia people we have used the correlation coefficient and g1 clustering type, for ordinary people we have used the cosine similarity measure and single link clustering. For both categories the number of target output clusters equals (number of RIPPER output clusters + the number of documents*0.2).

Although the clustering results with the best settings for hierarchical and agglomerative clustering were very close with regard to F-score (combining purity and inverse purity, see (Artiles et al., 2007) for a more detailed description), manual inspection of the content of the clusters has revealed big differences between the two approaches. Clusters that are output by our hierarchical algorithm look more homogeneous (higher purity), whereas inverse purity seems better for the agglomerative clustering. Therefor we have decided to take the best of two worlds and combined resulting clusters of both algorithms.

3.3 Merging of clustering results

Classification and clustering with optimal settings resulted in three sets of clusters, one based on pairwise similarity vectors and two based on keyword matrices. Since the former set tends to have better precision, which seems logical because more evident features are used for classification, we used this set as "seed" clusters. The two remaining sets were used to improve recall.

Merging was done in the following way: first we compare the initial set with the result of the agglomerative clustering by trying to find the biggest intersection. We remove the intersection from the smallest cluster and add both clusters to the final set. The resulting set of clusters is further improved by using the result of the hierarchical clustering. Here we apply another combining strategy: if two documents form one cluster in the initial set, but are in separate clusters in the other set, we merge these two clusters. Table 1 lists all results of the separate clustering algorithms as well as the final clustering results for the Wikipedia person names. Second half of the ta-

[4] http://glaros.dtc.umn.edu/gkhome/views/cluto

Person Name Wikipedia	Ripper	agglom.	hierarch.	merged
Alexander Macomb	.69/.63	.64/.56	.57/.47	.79/.80
David Lodge	.69/.65	.69/.64	.43/.33	.79/.85
George Clinton	.65/.62	.64/.59	.54/.45	.75/.80
John Kennedy	.67/.62	.70/.66	.49/.39	.76/.80
Michael Howard	.56/.54	.63/.62	.65/.58	.62/.75
Paul Collins	.54/.57	.64/.62	.63/.56	.55/.62
Tony Abbott	.63/.59	.67/.63	.62/.54	.77/.83
Average Scores all Training Data	.73/.76	.67/.72	.62/.60	.66/.75

Table 1: Results on Training Data

Person Name ACL	Ripper	agglom.	hierarch.	merged
Chris Brockett	.49/.39	.74/.69	.70/.61	.79/.80
Dekang Lin	.69/.58	.76/.67	.59/.47	.93/.89
Frank Keller	.48/.41	.68/.75	.64/.62	.56/.71
James Curran	.53/.50	.64/.77	.75/.78	.54/.72
Jerry Hobbs	.50/.39	.02/.01	.58/.47	.74/.70
Leon Barrett	.47/.40	.67/.74	.65/.66	.57/.73
Mark Johnson	.45/.42	.55/.70	.65/.77	.44/.65
Robert Moore	.39/.37	.60/.71	.66/.68	.46/.65
Sharon Goldwater	.60/.49	.72/.61	.40/.29	.91/.86
Stephen Clark	.41/.42	.53/.67	.68/.75	.46/.67
Average Scores all Test Data	.49/.45	.58/.63	.69/.69	.61/.74

Table 2: Results on Test Data

ble shows the average results for the separate and combined algorithms. The first score always refers to $F_\alpha = 0.5$, the second score refers to $F_\alpha = 0.2$.

The average scores, that were calculated on the complete training set, show that RIPPER outperforms the combined clusters.

4 Results on the test data

4.1 Final settings

For our classification algorithm, we have finally not kept the best settings for the training data, as this led to an alarming over-assignment of the positive class, thus linking nearly every document to each other. Therefore, we were forced to define a more strict rule set. For the clustering algorithms, we have used the optimal parameter settings as described in Section 3.

4.2 Test results

Table 2 lists the results for the separate and merged clustering for SET 1 in the test data (participants in the ACL conference) and the average for all algorithms. The average score, that has been calculated on the complete test set, shows that the combined clusters outperform the separate algorithms for $F_\alpha = 0.2$, but the hierarchical algorithm outperforms the others for $F_\alpha = 0.5$. Table 3 lists the average results for purity, inverse purity and the F-measures.

Test set	Purity	Inverse Purity	$F = \alpha = 0.5$	$F = \alpha = 0.2$
Set1	.57	.85	.64	.73
Set2	.45	.91	.58	.73
Set3	.48	.89	.60	.73
Global	.50	.88	.60	.73

Table 3: Purity/Inverse Purity Results on Test Data

5 Conclusions

We proposed and validated a combined classification and clustering approach for resolving web people ambiguity. In future work we plan to experiment with clustering algorithms that don't require a predefined number of clusters, as our tests revealed a big impact of the cluster size on our results. We will also experiment with meta-learning, other merging techniques and evaluation metrics. Furthermore, we will investigate the impact of intra-document and inter-document coreference resolution on web people disambiguation.

6 References

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

A. Bagga and B. Baldwin. 1998. *Entity-based cross-document co-referencing using the vector space model, Proceedings of the 17th international conference on Computational linguistics*, 75–85.

A. Berger and R. Caruana and D. Cohn and D. Freitag and V. Mittal. 2000. *Bridging the Lexical Chasm: Statistical Approaches to Answer Finding, Proc. Int. Conf. Reasearch and Development in Information Retrieval*, 192–199.

William W. Cohen. 1995. *Fast Effective Rule Induction, Proceedings of the 12th International Conference on Machine Learning*, 115–123. Tahoe City, CA.

Walter Daelemans and Antal van den Bosch. 2005. *Memory-Based Language Processing*. Cambridge University Press.

Veronique Hoste. 2005. *Optimization Issues in Machine Learning of Coreference Resolution*. Phd dissertation, Antwerp University.

M.B. Fleischman and E. Hovy. 2004. *Multi-document person name resolution, Proceedings of 42nd Annual Meeting of the Association for Computational Linguistics (ACL), Reference Resolution Workshop*.

G. Mann and D. Yarowsky. 2003. *Unsupervised personal name disambiguation, Proceedings of CoNLL-2003*, 33–40. Edmonton, Canada.

T. Pedersen and A. Purandare and A. Kulkarni. 2006. *Name Discrimination by Clustering Similar Contexts, Proceedings of the World Wide Web Conference (WWW)*.