The role of named entities in Web People Search

Javier Artiles
UNED NLP & IR group
Madrid, Spain
javart@bec.uned.es

Enrique Amigó
UNED NLP & IR group
Madrid, Spain
enrique@lsi.uned.es

Julio Gonzalo
UNED NLP & IR group
Madrid, Spain
julio@lsi.uned.es

Abstract

The ambiguity of person names in the Web has become a new area of interest for NLP researchers. This challenging problem has been formulated as the task of clustering Web search results (returned in response to a person name query) according to the individual they mention. In this paper we compare the coverage, reliability and independence of a number of features that are potential information sources for this clustering task, paying special attention to the role of named entities in the texts to be clustered. Although named entities are used in most approaches, our results show that, independently of the Machine Learning or Clustering algorithm used, named entity recognition and classification per se only make a small contribution to solve the problem.

1 Introduction

Searching the Web for names of people is a highly ambiguous task, because a single name tends to be shared by many people. This ambiguity has recently become an active research topic and, simultaneously, in a relevant application domain for web search services: Zoominfo.com, Spock.com, 123people.com are examples of sites which perform web people search, although with limited disambiguation capabilities.

A study of the query log of the AllTheWeb and Altavista search sites gives an idea of the relevance of the people search task: 11-17% of the queries were composed of a person name with additional terms and 4% were identified as person names (Spink et al., 2004). According to the data available from 1990 U.S. Census Bureau, only 90,000 different names are shared by 100 million people (Artiles et al., 2005). As the amount of information in the WWW grows, more of these people are mentioned in different web pages. Therefore, a query for a common name in the Web will usually produce a list of results where different people are mentioned.

This situation leaves to the user the task of finding the pages relevant to the particular person he is interested in. The user might refine the original query with additional terms, but this risks excluding relevant documents in the process. In some cases, the existence of a predominant person (such as a celebrity or a historical figure) makes it likely to dominate the ranking of search results, complicating the task of finding information about other people sharing her name. The Web People Search task, as defined in the first WePS evaluation campaign (Artiles et al., 2007), consists of grouping search results for a given name according to the different people that share it.

Our goal in this paper is to study which document features can contribute to this task, and in particular to find out which is the role that can be played by named entities (NEs): (i) How reliable is NEs overlap between documents as a source of evidence to cluster pages? (ii) How much recall does it provide? (iii) How unique is this signal? (i.e. is it redundant with other sources of information such as n-gram overlap?); and (iv) How sensitive is this signal to the peculiarities of a given NE recognition system, such as the granularity of its NE classification and the quality of its results?

Our aim is to reach conclusions which are are not tied to a particular choice of Clustering or Machine Learning algorithms. We have taken two decisions in this direction: first, we have focused on the problem of deciding whether two web pages refer to the same individual or not (page coreference task). This is the kind of relatedness measure that most clustering algorithms use, but in this way we can factor out the algorithm and its parameter settings. Second, we have developed a measure, *Maximal Pairwise Accuracy* (PWA) which, given

Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 534–542,
Singapore, 6-7 August 2009.

an information source for the problem, estimates an upper bound for the performance of any Machine Learning algorithm using this information. We have used PWA as the basic metric to study the role of different document features in solving the coreference problem, and then we have checked the predictive power of PWA with a Decision Tree algorithm.

The remainder of the paper is organised as follows. First, we examine the previous work in Section 2. Then we describe the our experimental settings (datasets and features we have used) in Section 3 and our empirical study in Section 4. The paper ends with some conclusions in Section 5.

2 Previous work

In this section we will discuss (i) the state of the art in Web People Search in general, focusing on which features are used to solve the problem; and (ii) lessons learnt from the WePS evaluation campaign where most approaches to the problem have been tested and compared.

The disambiguation of person names in Web results is usually compared to two other Natural Language Processing tasks: Word Sense Disambiguation (WSD) (Agirre and Edmonds, 2006) and Cross-document Coreference (CDC) (Bagga and Baldwin, 1998). Most of early research work on person name ambiguity focuses on the CDC problem or uses methods found in the WSD literature. It is only recently that the web name ambiguity has been approached as a separate problem and defined as an NLP task - *Web People Search* - on its own (Artiles et al., 2005; Artiles et al., 2007).

Therefore, it is useful to point out some crucial differences between WSD, CRC and WePS:

- WSD typically concentrates in the disambiguation of common words (nouns, verbs, adjectives) for which a relatively small number of senses exist, compared to the hundreds or thousands of people that can share the same name.
- WSD can rely on dictionaries to define the number of possible senses for a word. In the case of name ambiguity no such dictionary is available, even though in theory there is an exact number of people that can be accounted as sharing the same name.
- The objective of CDC is to reconstruct the coreference chain for every mention of a person. In Web person name disambiguation it suffices to group the documents that contain at least one mention to the same person.

Before the first WePS evaluation campaign in 2007 (Artiles et al., 2007), research on the topic was not based on a consistent task definition, and it lacked a standard manually annotated testbed. In the WePS task, systems were given the top web search results produced by a person name query. The expected output was a clustering of these results, where each cluster should contain all and only those documents referring to the same individual.

2.1 Features for Web People Search

Many different features have been used to represent documents where an ambiguous name is mentioned. The most basic is a **Bag of Words** (BoW) representation of the document text. Within-document coreference resolution has been applied to produce summaries of text surrounding occurrences of the name (Bagga and Baldwin, 1998; Gooi and Allan, 2004). Nevertheless, the full document text is present in most systems, sometimes as the only feature (Sugiyama and Okumura, 2007) and sometimes in combination with others - see for instance (Chen and Martin, 2007; Popescu and Magnini, 2007)-. Other representations use the link structure (Malin, 2005) or generate graph representations of the extracted features (Kalashnikov et al., 2007).

Some researchers (Cucerzan, 2007; Nguyen and Cao, 2008) have explored the use of **Wikipedia information** to improve the disambiguation process. Wikipedia provides candidate entities that are linked to specific mentions in a text. The obvious limitation of this approach is that only celebrities and historical figures can be identified in this way. These approaches are yet to be applied to the specific task of grouping search results.

Biographical features are strongly related to NEs and have also been proposed for this task due to its high precision. Mann (2003) extracted these features using lexical patterns to group pages about the same person. Al-Kamha (2004) used a simpler approach, based on hand coded features (e.g. email, zip codes, addresses, etc). In Wan (2005), biographical information (person name, title, organisation, email address and phone number) improves the clustering results when combined with lexical features (words from the doc-

ument) and NE (person, location, organisation).

The most used feature for the Web People Search task, however, are **NEs**. Ravin (1999) introduced a rule-based approach that tackles both variation and ambiguity analysing the structure of names. In most recent research, NEs (person, location and organisations) are extracted from the text and used as a source of evidence to calculate the similarity between documents -see for instance (Blume, 2005; Chen and Martin, 2007; Popescu and Magnini, 2007; Kalashnikov et al., 2007)-. For instance, Blume (2005) uses NEs coocurring with the ambiguous mentions of a name as a key feature for the disambiguation process. Saggion (2008) compared the performace of NEs versus BoW features. In his experiments a only a representation based on Organisation NEs outperformed the word based approach. Furthermore, this result is highly dependent on the choice of metric weighting (NEs achieve high precision at the cost of a low recall and viceversa for BoW).

In summary, the most common document representations for the problem include BoW and NEs, and in some cases biographical features extracted from the text.

2.2 Named entities in the WePS campaign

Among the 16 teams that submitted results for the first WePS campaign, 10 of them[1] used NEs in their document representation. This makes NEs the second most common type of feature; only the BoW feature was more popular. Other features used by the systems include noun phrases (Chen and Martin, 2007), word n-grams (Popescu and Magnini, 2007), emails and URLs (del Valle-Agudo et al., 2007), etc. In 2009, the second WePS campaign showed similar trends regarding the use of NE features (Artiles et al., 2009).

Due to the complexity of systems, the results of the WePS evaluation do not provide a direct answer regarding the advantages of using NEs over other computationally lighter features such as BoW or word n-grams. But the WePS campaigns did provide a useful, standardised resource to perform the type of studies that were not possible before. In the next Section we describe this dataset and how it has been adapted for our purposes.

[1]By team ID: CU-COMSEM, IRST-BP, PSNUS, SHEF, FICO, UNN, AUG, JHU1, DFKI2, UC3M13

3 Experimental settings

3.1 Data

We have used the testbeds from WePS-1 (Artiles et al., 2007)[2] and WePS-2 (Artiles et al., 2009) evaluation campaigns [3].

Each WePS dataset consists of 30 test cases: a random sample of 10 names from the US Census, 10 names from Wikipedia, and 10 names from Programme Committees in the Computer Science domain (ACL and ECDL). Each test case consists of, at most, 100 web pages from the top search results of a web search engine, using a (quoted) person name as query.

For each test case, annotators were asked to organise the web pages in groups where all documents refer to the same person. In cases where a web page refers to more than one person using the same ambiguous name (e.g. a web page with search results from Amazon), the document is assigned to as many groups as necessary. Documents were discarded when they did not contain any useful information about the person being referred.

Both the WePS-1 and WePS-2 testbeds have been used to evaluate clustering systems by WePS task participants, and are now the standard testbed to test Web People Search systems.

3.2 Features

The evaluated features can be grouped in four main groups: token-based, n-grams, phrases and NEs. Wherever possible, we have generated *local* versions of these features that only consider the sentences of the text that mention the ambiguous person name[4]. Token-based features considered include document full text tokens, lemmas (using the OAK analyser, see below), title, snippet (returned in the list of search results) and URL (tokenised using non alphanumeric characters as boundaries) tokens. English stopwords were removed, including Web specific stopwords, as file and domain extensions, etc.

We generated **word n-grams** of length 2 to 5,

[2]The WePS-1 corpus includes data from the Web03 testbed (Mann, 2006) which follows similar annotation guidelines, although the number of document per ambiguous name is more variable.

[3]Both corpora are available from the WePS website http://nlp.uned.es/weps

[4]A very sparse feature might never occur in a sentence with the person name. In that cases there is no *local* version of the feature.

using the sentences found in the document text. Punctuation tokens (commas, dots, etc) were generalised as the same token. N-grams were discarded when they were composed only of stopwords or when they did not contain at least one token formed by alphanumeric characters (e.g. n-grams like "at the" or "# @"). **Noun phrases** (using OAK analyser) were detected in the document and filtered in a similar way.

Named entities were extracted using two different tools: the Stanford NE Recogniser and the OAK System[5].

Stanford NE Recogniser[6] is a high-performance Named Entity Recognition (NER) system based on Machine Learning. It provides a general implementation of linear chain Conditional Random Field sequence models and includes a model trained on data from CoNLL, MUC6, MUC7, and ACE newswire. Three types of entities were extracted: person, location and organisation.

OAK[7] is a rule based English analyser that includes many functionalities (POS tagger, stemmer, chunker, Named Entity (NE) tagger, dependency analyser, parser, etc). It provides a fine grained NE recognition covering 100 different NE types (Sekine, 2008). Given the sparseness of most of these fine-grained NE types, we have merged them in coarser groups: event, facility, location, person, organisation, product, periodx, timex and numex.

We have also used the results of a **baseline NE** recognition for comparison purposes. This method detects sequences of two or more uppercased tokens in the text, and discards those that are found lowercased in the same document or that are composed solely of stopwords.

Other features are: emails, outgoing links found in the web pages and two boolean flags that indicate whether a pair of documents is linked or belongs to the same domain. Because of their low impact in the results these features haven't received an individual analysis, but they are included in the "all features" combination in Figure 7.

[5]From the output of both systems we have discarded person NEs made of only one token (these are often first names that significantly deteriorate the quality of the comparison between documents).

[6]http://nlp.stanford.edu/software/CRF-NER.shtml

[7]http://nlp.cs.nyu.edu/oak . OAK was also used to detect noun phrases and extract lemmas from the text.

4 Experiments and results

4.1 Reformulating WePS as a classification task

As our goal is to study the impact of different features (information sources) in the task, a direct evaluation in terms of clustering has serious disadvantages. Given the output of a clustering system it is not straightforward to assess why a document has been assigned to a particular cluster. There are at least three different factors: the document similarity function, the clustering algorithm and its parameter settings. Features are part of the document similarity function, but its performance in the clustering task depends on the other factors as well. This makes it difficult to perform error analysis in terms of the features used to represent the documents.

Therefore we have decided to transform the clustering problem into a classification problem: deciding whether two documents refer to the same person. Each pair of documents in a name dataset is considered a classification instance. Instances are labelled as coreferent (if they share the same cluster in the gold standard) or non coreferent (if they do not share the same cluster). Then we can evaluate the performance of each feature separately by measuring its ability to rank coreferent pairs higher and non coreferent pairs lower. In the case of feature combinations we can study them by training a classifier or using the maximal pairwise accuracy methods (explained in Section 4.3).

Each instance (pair of documents) is represented by the similarity scores obtained using different features and similarity metrics. We have calculated for each feature three similarity metrics: Dice's coefficient, cosine (using standard tf.idf weighting) and a measure that simply counts the size of the intersection set for a given feature between both documents. After testing these metrics we found that Dice provides the best results across different feature types. Differences between Dice and cosine were consistent, although they were not especially large. A possible explanation is that Dice does not take into account the redundancy of an n-gram or NE in the document, and the cosine distance does. This can be a crucial factor, for instance, in the document retrieval by topic; but it doesn't seem to be the case when dealing with name ambiguity.

The resulting classification testbed consists of 293,914 instances with the distribution shown in

Table 1, where each instance is represented by 69 features.

	true	false	total
WePS1	61,290	122,437	183,727
WePS2	54,641	55,546	110,187
WePS1+WePS2	115,931	177,983	293,914

Table 1: Distribution of classification instances

4.2 Analysis of individual features

There are two main aspects related with the usefulness of a feature for WePS task. The first one is its performance. That is, to what extent the similarity between two documents according to a feature implies that both mention the same person. The second aspect is to what extent a feature is orthogonal or redundant with respect to the standard token based similarity.

4.2.1 Feature performance

According to the transformation of WePS clustering problem into a classification task (described in Section 4.1), we follow the next steps to study the performance of individual features. First, we compute the Dice coefficient similarity over each feature for all document pairs. Then we rank the document pair instances according to these similarities. A good feature should rank positive instances on top. If the number of coreferent pairs in the top n pairs is t_n and the total number of coreferent pairs is t, then $P = \frac{t_n}{n}$ and $R = \frac{t_n}{t}$. We plot the obtained precision/recall curves in Figures 1, 2, 3 and 4.

From the figures we can draw the following conclusions:

First, considering subsets of tokens or lemmatised tokens does not outperform the basic token distance (figure 1 compares token-based features). We see that only local and snippet tokens perform slightly better at low recall values, but do not go beyond recall 0.3.

Second, shallow parsing or n-grams longer than 2 do not seem to be effective, but using bi-grams improves the results in comparison with tokens. Figure 2 compares n-grams of different sizes with noun phrases and tokens. Overall, noun phrases have a poor performance, and bi-grams give the best results up to recall 0.7. Four-grams give slightly better precision but only reach 0.3 recall, and three-grams do not give better precision than bi-grams.

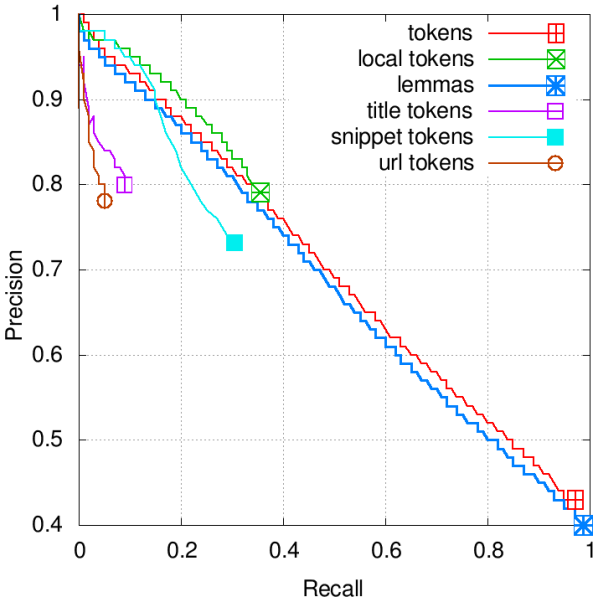

Figure 1: Precision/Recall curve of token-based features

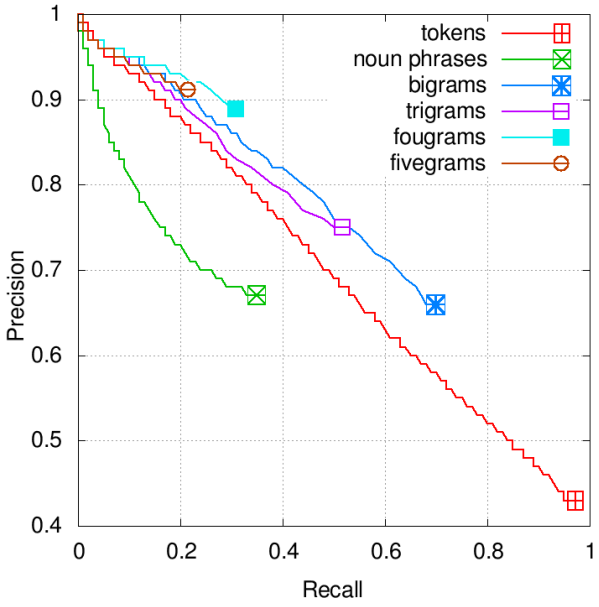

Figure 2: Precision/Recall curve of word n-grams

Third, individual types of NEs do not improve over tokens. Figure 3 and Figure 4 display the results obtained by the Stanford and OAK NER tools respectively. In the best case, Stanford person and organisation named entities obtain results that match the tokens feature, but only at lower levels of recall.

Finally, using different NER systems clearly leads to different results. Surprisingly, the baseline NE system yields better results in a one to one comparison, although it must be noted that this baseline agglomerates different types of en-

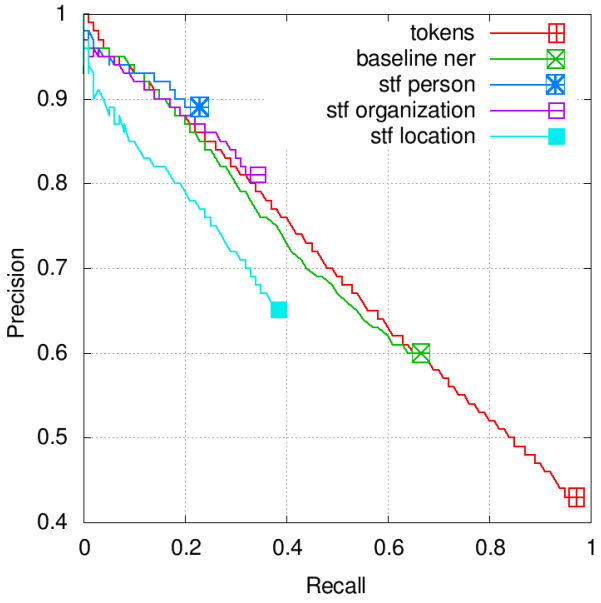

Figure 3: Precision/Recall curve of NEs obtained with the Stanford NER tool

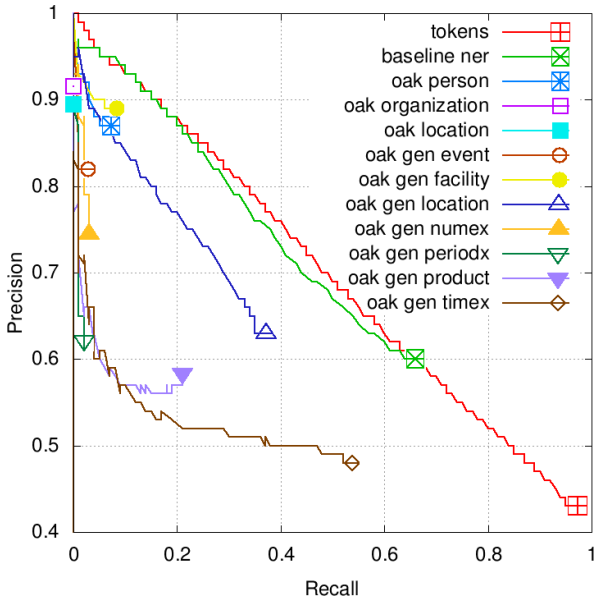

Figure 4: Precision/Recall curve of NEs obtained with the OAK NER tool

tities that are separated in the case of Stanford and OAK, and this has a direct impact on its recall. The OAK results are below the tokens and NE baseline, possibly due to the sparseness of its very fine grained features. In NE types, cases such as person and organisation results are still lower than obtained with Stanford.

4.2.2 Redundancy

In addition to performance, named entities (as well as other features) are potentially useful for the task only if they provide information that complements (i.e. that does not substantially overlap) the basic token based metric. To estimate this redundancy, let us consider all document tuples of size four $< a, b, c, d >$. In 99% of the cases, token similarity is different for $< a, b >$ than for $< c, d >$. We take combinations such that $< a, b >$ are more similar to each other than $< c, d >$ according to tokens. That is:

$$\text{sim}_{\text{token}}(a, b) > \text{sim}_{\text{token}}(c, d)$$

Then for any other feature similarity $sim_x(a, b)$, we will talk about *redundant* samples when $sim_x(a, b) > sim_x(c, d)$, *non redundant* samples when $sim_x(a, b) < sim_x(c, d)$, and *non informative* samples when $sim_x(a, b) = sim_x(c, d)$. If all samples are redundant or non informative, then sim_x does not provide additional information for the classification task.

Figure 5 shows the proportion of redundant, non redundant and non informative samples for several similarity criteria, as compared to token-based similarity. In most cases NE based similarities give little additional information: the baseline NE recogniser, which has the largest independent contribution, gives additional information in less than 20% of cases.

In summary, analysing individual features, the NEs do not outperform BoW in terms of the classification task. In addition, NEs tend to be redundant regarding BoW. However, if we are able to combine optimally the contributions of the different features, the BoW approach could be improved. We address this issue in the next section.

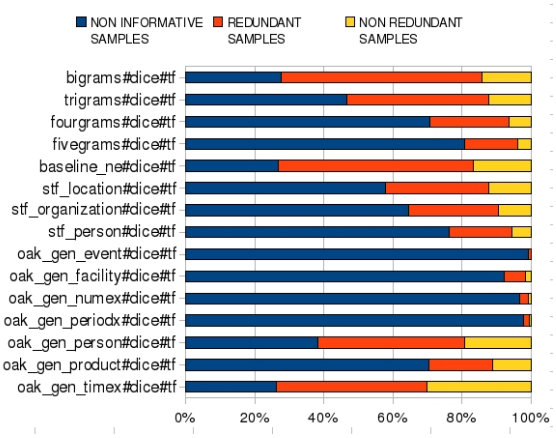

Figure 5: Independence of similarity criteria with respect to the token based feature

4.3 Analysis of feature combinations

Up to now we have analysed the usefulness of individual features for the WePS Task. However, this begs to ask to what extent the NE features can contribute to the task when they are combined together and with token and n-gram based features. First, we use each feature combinations as the input for a Machine Learning algorithm. In particular, we use a Decision Tree algorithm and WePS-1 data for training and WePS-2 data for testing. The Decision Tree algorithm was chosen because we have a small set of features to train (similarity metrics) and some of these features output Boolean values.

Results obtained with this setup, however, can be dependent on the choice of the ML approach. To overcome this problem, in addition to the results of a Decision Tree Machine Learning algorithm, we introduce a *Maximal Pairwise Accuracy* (MPA) measure that provides an upper bound for any machine learning algorithm using a feature combination.

We can estimate the performance of an individual similarity feature x in terms of accuracy. It is considered a correct answer when the similarity $x(a, a')$ between two pages referring to the same person is higher than the similarity $x(b, c)$ between two pages referring to different people. Let us call this estimation *Pairwise Accuracy*. In terms of probability it can be defined as:

$$PWA = \text{Prob}(x(a, a') > x(c, d))$$

PWA is defined over a single feature (similarity metric). When considering more than one similarity measure, the results depend on how measures are weighted. In that case we assume that the best possible weighting is applied. When combining a set of features $X = \{x_1 \dots x_n\}$, a perfect Machine Learning algorithm would learn to always "listen" to the features giving correct information and ignore the features giving erroneous information. In other words, if at least one feature gives correct information, then the perfect algorithm would produce a correct output. This is what we call the *Maximal Pairwise Accuracy* estimation of an upper bound for any ML system using the set of features X:

$$\text{MaxPWA}(X) =$$
$$\text{Prob}(\exists x \in X . x(a, a') > x(c, d))$$

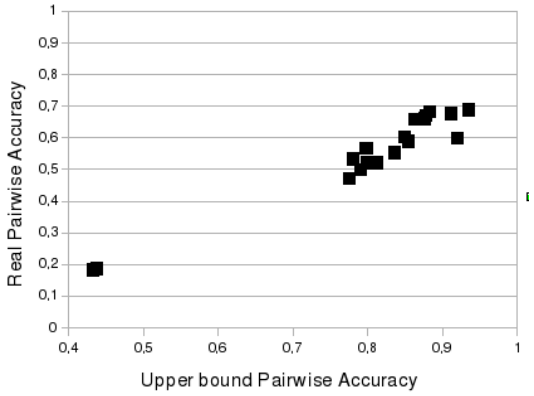

Figure 6: Estimated PWA upper bound versus the real PWA of decision trees trained with feature combinations

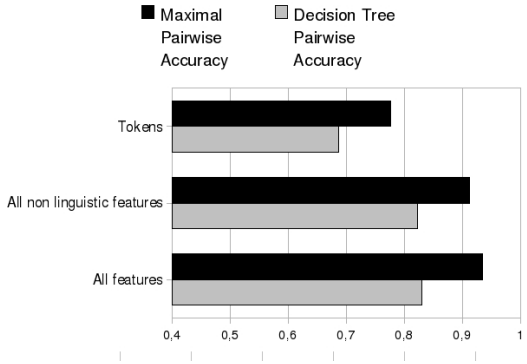

Figure 7: Maximal Pairwise Accuracy vs. results of a Decision Tree

The upper bound (MaxPWA) of feature combinations happens to be highly correlated with the PWA obtained by the Decision Tree algorithm (using its confidence values as a similarity metric). Figure 6 shows this correlation for several features combinations. This is an indication that the Decision Tree is effectively using the information in the feature set.

Figure 7 shows the PWA upper bound estimation and the actual PWA performance of a Decision Tree ML algorithm for three combinations: (i) all features; (ii) non linguistic features, i.e., features which can be extracted without natural language processing machinery: tokens, url, title, snippet, local tokens, n-grams and local n-grams; and (iii) just tokens. The results show that according to both the Decision Tree results and the upper-bound (MaxPWA), adding new features to tokens improves the classification. However, taking non-linguistic features obtains similar results than taking all features. Our conclusion is that NE features are useful for the task, but do not seem to offer a

competitive advantage when compared with non-linguistic features, and are more computationally expensive. Note that we are using NE features in a direct way: our results do not exclude the possibility of effectively exploiting NEs in more sophisticated ways, such as, for instance, exploiting the underlying social network relationships between NEs in the texts.

4.3.1 Results on the clustering task

In order to validate our results, we have tested whether the classifiers learned with our feature sets lead to competitive systems for the full clustering task. In order to do so, we use the output of the classifiers as similarity metrics for a particular clustering algorithm, using WePS-1 to train the classifiers and WePS-2 for testing.

We have used a Hierarchical Agglomerative Clustering algorithm (HAC) with single linkage, using the classifier's confidence value in the negative answer for each instance as a distance metric[8] between document pairs. HAC is the algorithm used by some of the best performing systems in the WePS-2 evaluation. The distance threshold was trained using the WePS-1 data. We report results with the official WePS-2 evaluation metrics: extended B-Cubed Precision and Recall (Amigó et al., 2008).

Two Decision Tree models were evaluated: (i) *ML-ALL* is a model trained using all the available features (which obtains 0.76 accuracy in the classification task) (ii) *ML-NON_LING* was trained with all the features except for OAK and Stanford NEs, noun phrases, lemmas and gazetteer features (which obtains 0.75 accuracy in the classification task). These are the same classifiers considered in Figure 7.

Table 2 shows the results obtained in the clustering task by the two DT models, together with the four top scoring WePS-2 systems and the average values for all WePS-2 systems. We found that a ML based clustering using only non linguistic information slightly outperforms the best participant in WePS-2. Surprisingly, adding linguistic information (NEs, noun phrases, etc.) has a small negative impact on the results (0.81 versus 0.83), although the classifier with linguistic information was a bit better than the non-linguistic one. This seems to be another indication that the use of noun phrases and other linguistic features to improve the task is non-obvious to say the least.

		B-Cubed	
run	F-α $=_{0.5}$	Pre.	Rec.
ML-NON_LING	.83	.91	.77
S-1	.82	.87	.79
ML- ALL	.81	.89	.76
S-2	.81	.85	.80
S-3	.81	.93	.73
S-4	.72	.82	.66
WePS-2 systems aver.	.61	.74	.63

Table 2: Evaluation on the WePS-2 clustering task

5 Conclusions

We have presented an empirical study that tries to determine the potential role of several sources of information to solve the Web People Search clustering problem, with a particular focus on studying the role of named entities in the task.

To abstract the study from the particular choice of a clustering algorithm and a parameter setting, we have reformulated the problem as a coreference classification task: deciding whether two pages refer to the same person or not. We have also proposed the *Maximal Pairwise Accuracy* estimation that establish an upper bound for the results obtained by any Machine Learning algorithm using a particular set of features.

Our results indicate that (i) NEs do not provide a substantial competitive advantage in the clustering process when compared to a rich combination of simpler features that do not require linguistic processing (local, global and snippet tokens, n-grams, etc.); (ii) results are sensitive to the NER system used: when using all NE features for training, the richer number of features provided by OAK seems to have an advantage over the simpler types in Stanford NER and the baseline NER system.

This is not exactly a prescription against the use of NEs for Web People Search, because linguistic knowledge can be useful for other aspects of the problem, such as visualisation of results and description of the persons/clusters obtained: for example, from a user point of view a network of the connections of a person with other persons and organisations (which can only be done with NER) can be part of a person's profile and may help as a summary of the cluster contents. But from the perspective of the clustering problem per se, a direct use of NEs and other linguistic features does not seem to pay off.

[8] The DT classifier output consists of two confidence values, one for the positive and one for the negative answer, that add up to 1.0 .

Acknowledgments

This work has been partially supported by the Regional Government of Madrid, project MAVIR S0505-TIC0267.

References

Eneko Agirre and Philip Edmonds, editors. 2006. *Word Sense Disambiguation: Algorithms and Applications*. Springer.

Reema Al-Kamha and David W. Embley. 2004. Grouping search-engine returned citations for person-name queries. In *WIDM '04: Proceedings of the 6th annual ACM international workshop on Web information and data management*. ACM Press.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2008. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*.

Javier Artiles, Julio Gonzalo, and Felisa Verdejo. 2005. A testbed for people searching strategies in the www. In *SIGIR*.

Javier Artiles, Julio Gonzalo, and Satoshi Sekine. 2007. 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)*. ACL.

Javier Artiles, Julio Gonzalo, and Satoshi Sekine. 2009. Weps 2 evaluation campaign: overview of the web people search clustering task. In *WePS 2 Evaluation Workshop. WWW Conference 2009*.

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th international conference on Computational linguistics*. ACL.

Matthias Blume. 2005. Automatic entity disambiguation: Benefits to ner, relation extraction, link analysis, and inference. In *International Conference on Intelligence Analysis*.

Ying Chen and James H. Martin. 2007. Cu-comsem: Exploring rich features for unsupervised web personal name disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations*. ACL.

Silviu Cucerzan. 2007. Large scale named entity disambiguation based on wikipedia data. In *The EMNLP-CoNLL-2007*.

David del Valle-Agudo, César de Pablo-Sánchez, and María Teresa Vicente-Díez. 2007. Uc3m-13: Disambiguation of person names based on the composition of simple bags of typed terms. In *Proceedings of the Fourth International Workshop on Semantic Evaluations*. ACL.

Chung Heong Gooi and James Allan. 2004. Cross-document coreference on a large scale corpus. In *HLT-NAACL*.

Dmitri V. Kalashnikov, Stella Chen, Rabia Nuray, Sharad Mehrotra, and Naveen Ashish. 2007. Disambiguation algorithm for people search on the web. In *Proc. of IEEE International Conference on Data Engineering (IEEE ICDE)*.

Bradley Malin. 2005. Unsupervised name disambiguation via social network similarity. In *Workshop on Link Analysis, Counterterrorism, and Security*.

Gideon S. Mann and David Yarowsky. 2003. Unsupervised personal name disambiguation. In *Proceedings of the seventh conference on Natural Language Learning (CoNLL) at HLT-NAACL 2003*. ACL.

Gideon S. Mann. 2006. *Multi-Document Statistical Fact Extraction and Fusion*. Ph.D. thesis, Johns Hopkins University.

Hien T. Nguyen and Tru H. Cao, 2008. *Named Entity Disambiguation: A Hybrid Statistical and Rule-Based Incremental Approach*. Springer.

Octavian Popescu and Bernardo Magnini. 2007. Irst-bp: Web people search using name entities. In *Proceedings of the Fourth International Workshop on Semantic Evaluations*. ACL.

Y. Ravin and Z. Kazi. 1999. Is hillary rodham clinton the president? disambiguating names across documents. In *Proceedings of the ACL '99 Workshop on Coreference and its Applications Association for Computational Linguistics*.

Horacio Saggion. 2008. Experiments on semantic-based clustering for cross-document coreference. In *International Joint Conference on Natural language Processing.*

Satoshi Sekine. 2008. Extended named entity ontology with attribute information. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*.

Amanda Spink, Bernard Jansen, and Jan Pedersen. 2004. Searching for people on web search engines. *Journal of Documentation*, 60:266 – 278.

Kazunari Sugiyama and Manabu Okumura. 2007. Titpi: Web people search task using semi-supervised clustering approach. In *Proceedings of the Fourth International Workshop on Semantic Evaluations*. ACL.

Xiaojun Wan, Jianfeng Gao, Mu Li, and Binggong Ding. 2005. Person resolution in person search results: Webhawk. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM Press.