A Data Driven Approach for Person Name Disambiguation in Web Search Results

Agustín D. Delgado[1], **Raquel Martínez**[1], **Víctor Fresno**[1], **Soto Montalvo**[2]
[1]Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain
[2]Universidad Rey Juan Carlos (URJC), Móstoles, Spain
[1]{agustin.delgado,raquel,vfresno}@lsi.uned.es, [2]soto.montalvo@urjc.es

Abstract

This paper presents an unsupervised approach for the task of clustering the results of a search engine when the query is a person name shared by different individuals. We propose an algorithm that calculates the number of clusters and establishes the groups of web pages according to the different individuals without the need to any training data or predefined thresholds, as the successful state of the art systems do. In addition, most of those systems do not deal with social media web pages and their performance could fail in a real scenario. In this paper we also propose a heuristic method for the treatment of social networking profiles. Our approach is compared with four gold standard collections for this task obtaining really competitive results, comparable to those obtained by some approaches with supervision.

1 Introduction

Resolving the ambiguity of person names in web search results is a challenging problem becoming an area of interest for Natural Language Processing (NLP) and Information Retrieval (IR) communities. This task can be defined informally as follows: given a query of a person name in addition to the results of a search engine for that query, the goal is to cluster the resultant web pages according to the different individuals they refer to. Thus, the challenge of this task is estimating the number of different individuals and grouping the pages of the same individual in the same cluster. The difficulty of this task resides in the fact that a single person name can be shared by many people: according to the U.S. Census Bureau, 90000 different names are shared by 100 million people (Artiles et al., 2007). This problem has had an impact in the Internet and that is why several vertical search engines specialized in web people search have appeared in the last years, e.g. spokeo.com or 123people.com. This task should not be mixed up with *entity linking* (EL), which goal is to link name mentions of entities in a document collection to entities in a reference knowledge base (typically Wikipedia), or to detect new entities.

The main difficulties of clustering web pages referring to the same individual come from their possible heterogeneous nature. For example, some pages may be professional sites, while others may be blogs containing personal information. In addition, the popularity of social networking services makes the search engine usually returns several social profiles belonging to different individuals sharing the same name, as much from the same social networking service as from different services. These social pages often introduce noisy information and make the state of the art algorithms break down (Berendsen et al., 2012). Due to these problems, the users have to refine the queries with additional terms. This task gets harder when the person name is shared by a celebrity or a historical figure, because the results of the search engines are dominated by that individual, making the search of information about other individuals more difficult.

WePS[1] (Web People Search) evaluation campaigns proposed this task in a web searching scenario providing several corpora for evaluating the results of their participants, particularly WePS-1, WePS-2 and WePS-3 campaigns. This framework allows our approach to be compared with the state of the art

[1] http://nlp.uned.es/weps/

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 301–310, Dublin, Ireland, August 23-29 2014.

systems. We also evaluate our system with ECIR2012 corpus[2], a data set that includes social networking profiles, providing a more real scenario for this task.

The most successful state of the art systems have addressed this problem with some kind of supervision. This work proposes a data-driven method for this task with the aim of eliminating the elements of human involvement in the process as much as possible. The main contribution of this work is a new unsupervised approach for resolving person name ambiguity of web search results based on the use of capitalized n-grams. In our approach the decision if two web pages have to be grouped only depends on the information of both pages. In addition, we also propose a heuristic method for the treatment of social media profile web pages in this context.

The paper is organized as follows: in Section 2 we discuss related work; Section 3 details the way we represent the web pages, the algorithm and the heuristic for social pages; in Section 4 we describe the collections used for evaluating our method and we show our results making a comparison with other systems; the paper ends with some conclusions and future work in Section 5.

2 Related Work

Several approaches have been proposed for clustering search results for a person name query. The main differences among all of them are the features they use to represent the web pages and the clustering algorithm. However, the most successful of them have in common that they use some kind of supervision: learning thresholds and/or fixing manually the value of some parameters according to training data.

Regarding the way of representing a web page, the most popular features used by the most successful state of the art approaches are Name Entities (NE) and Bag of Words (BoW) weighted by TF-IDF function. In addition to such features, the systems usually use other kind of information. Top systems from WePS-1 and WePS-2 campaigns, CU_COMSEM (Chen and Martin, 2007) and PolyUHK (Chen et al., 2009), distinguish several kind of tokens according to different schemes (URL tokens, title tokens, ...) and build a feature vector for each sort of tokens, using also information based on the noun phrases appearing in the documents. PolyUHK also adds pattern techniques, attribute extraction and detection when a web page is written in a formal way. A more recent system, HAC_Topic (Liu et al., 2011), also uses BoW of local and global terms weighted by TF-IDF. It adds a topic capturing method to create a Hit List of shared high weighted tokens for each cluster obtaining better results than WePS-1 participants. On the other hand, the WePS-3 best system, YHBJ (Chong and Shi, 2010), uses information extracted manually from Wikipedia adding to BoW and NE weighted by TF-IDF.

Regarding the clustering algorithms, looking at WePS campaigns results, the top ranked systems have in common the use of the Hierarchical Agglomerative Clustering algorithm (HAC) described in (Manning et al., 2008). Different versions of this algorithm were used by (Chen and Martin, 2007; Chen et al., 2009; Elmacioglu et al., 2007; Liu et al., 2011; Balog et al., 2009; Chong and Shi, 2010).

(Berendsen et al., 2012) presented another gold standard for this task, ECIR2012, composed by Dutch person names and social media profile web pages. The system of the authors, UvA, distinguishes the web pages between social ones and non social ones, clusters each group separately and then combines both clustering solutions. They represent each web page as a BoW vector weighted by TF-IDF, and use cosine similarity for comparing web pages. They use HAC algorithm for clustering non social web pages, while use a “one in one” policy for the social ones. Finally, they mix both groups by means of an algorithm which penalizes clusters with social webs or simply taking the union of both clustering solutions. They perform a partial parameter sweep on the WePS-2 data set to fix the clustering thresholds, while explore combinations of other system parameters.

The only system that does not use training data, DAEDALUS (Lana-Serrano et al., 2010), which uses k-Medoids, got poor results in WePS-3 campaign. In short, the successful state of the art systems need some kind of supervised learning using training data or fixing parameters manually. In this paper we explore and propose an approach to address this problem by means of data-driven techniques without the use of any kind of supervision.

[2] http://ilps.science.uva.nl/resources/ecir2012rdwps

3 Proposed Approach

We distinguish two main phases in this clustering task: web page representation (Sections 3.1 and 3.2) and web page grouping (Sections 3.3 and 3.4). In addition, we propose an heuristic to deal with social profiles web pages (Section 3.5).

3.1 Feature Selection

The aim of this phase is to extract relevant information that could identify an individual. We assume the main following hypotheses:

(i) Capitalized n-grams co-occurrence could be a reliable way for deciding when two web pages refer the same individual. Capitalized n-grams usually are Named Entities (organizations and company names, locations or other person names related with the individual) or information not detected by some NE recognizers as for example, the title of books, films, TV shows, and so on. In a previous study with WePS-1 training corpus using the Stanford NER[3] to annotate NE, we detected that only 55.78% of the capitalized tokens were annotated as NE or components of a NE by the NER tool. So the use of capitalized tokens allows increase the number of features compared to the use of only NE. We also compared the n-gram representation with capitalized tokens and with NE. We found that 30.97% of the 3-grams of capitalized tokens were also NE 3-grams, and 25.64% of the 4-grams of capitalized tokens were also NE 4-grams. So even in the case of n-grams the use of capitalized tokens increases the number of features compared to the use of only NE. Table 1 shows the differences in performance when using n-grams representation with NE or with capitalized tokens.

(ii) If two web pages share capitalized n-grams, the higher is the value of n, the more probable the two web pages refer to the same individual. In this case we define "long enough n-grams" as those compose by at least 3 capitalized tokens.

Thus, a web page W is initially represented as the sequence of tokens starting in uppercase, in the order as they appear in the web page. In each step of the algorithm, a web page W will be represented by its long enough n-grams, taking different values for n, as we describe in Section 3.4. Notice that some web pages could not be represented with this proposal because all their content was written in lowercase. In the case of the collections that we describe in Section 4.1, 0.63% of the web pages are not represented for this reason.

3.2 Weighting Functions

We test the well known TF and TF-IDF functions, and z-score (Andrade and Medina, 1998). The z-score of a n-gram a in a web page W_i is defined as follows: $z\text{-}score(a, W_i) = \frac{TF(a,W_i)-\mu}{\sigma}$, where $TF(a, W_i)$ is the frequency of the n-gram a in W_i; μ is the mean frequency of the background set; and σ is the standard deviation of the background set. In this context the background set is the set of web pages that share the person name. This score gives an idea of the distance of the frequency of an n-gram in a web page from the general distribution of this n-gram in the background set.

3.3 Similarity Functions

To determine the similarity between two web pages we try the cosine distance, a widely measure used in clustering, and the weighted Jaccard coefficient between two bags of n-grams defined as $W.Jaccard(W_i^n, W_j^n) = \frac{\sum_k min(m(t_{k_i}^n,i),m(t_{k_j}^n,j))}{\sum_k max(m(t_{k_i}^n,i),m(t_{k_j}^n,j))}$, where the meaning of $m(t_{k_i}^n, i)$ is explained in Section 3.4. Since weighted Jaccard coefficient needs non-negative entries and we want the cosine similarity of two documents to range from 0 to 1, we translate the values of the z-score so that they are always non-negative.

3.4 Algorithm

The algorithm $UPND$ (Unsupervised Person Name Disambiguator) can be seen in Algorithm 1. The description of this first algorithm does not take into account social profile web pages.

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

$UPND$ algorithm receives as input a set of web documents with a mention to the same person name, let be $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$, and starts assigning a cluster C_i for each document W_i. $UPND$ also receives as input a pair of positive integer values r_1 and r_2, such that $r_2 \geq r_1$, specifying the range of values of n in the n-grams extracted from each web document. In each step of the algorithm we assign to each web page W_i a bag of n-grams $W_i^n = \{(t_1^n, m(t_1^n, i)), (t_2^n, m(t_2^n, i)), \ldots, (t_{k_i}^n, m(t_{k_i}^n, i))\}$, where each t_r^n is a n-gram extracted from W_i and $m(t_r^n, i)$ is the corresponding weight of the n-gram t_r^n in the web page W_i, being $r \in \{1, 2, \ldots, k_i\}$. In Algorithm 1 the function $setNGrams(n, \mathcal{W})$ in line 6 calculates for each web page in the set $\mathcal{W}$ its bag of n-grams representation. $Sim(W_i^n, W_j^n)$ in line 9 refers to the similarity between web pages W_i and W_j.

To decide when two web pages refer the same individual we propose a threshold γ. For each pair of web pages represented as bag of n-grams, let be W_i^n and W_j^n, we compute the threshold as follows: $\gamma(W_i^n, W_j^n) = \frac{min(m,k) - shared(W_i^n, W_j^n)}{max(m,k)}$, where m and k are the number of n-grams of W_i and W_j respectively, and $shared(W_i^n, W_j^n)$ is the number of n-grams shared by those web pages i.e. $shared(W_i^n, W_j^n) = |W_i^n \cap W_j^n|$. Notice that $shared(W_i^n, W_j^n)$ is superiorly limited by $min(m, k)$.

This threshold holds two desirable properties: (i) The more n-grams are shared by W_i and W_j, the lower $\gamma(W_i^n, W_j^n)$ is, so the clustering condition of the algorithm is less strict. (ii) It avoids the penalization due to big differences between the size of the web pages.

Thus, we decide that two web pages W_i and W_j refer to the same person if $Sim(W_i^n, W_j^n) \geq \gamma(W_i^n, W_j^n)$, so $C_i = C_i \cup C_j$ (lines 9, 10 and 11).

We assume that we can get accurate and reliable information for disambiguating with n-grams of at least size 3. Thus, we propose to iterate this process for 3-grams and 4-grams, i.e. $UPND(\mathcal{W}, 3, 4)$. We consider that selecting a value of n grater than 4 could lead to find few n-grams, so that many web pages could be under-represented. On the other hand, previous experiments using also bigrams showed that they are not suitable for this approach. This algorithm is polynomial and has a computational cost in $\mathcal{O}(N^2)$, where N is the number of web pages.

Algorithm 1 $UPND(\mathcal{W}, r_1, r_2)$

```
Require: Set of web pages that shared a person name $\mathcal{W} = \{W_1, W_2, ..., W_N\}$, $r_1, r_2 \geq 1$ such that $r_2 \geq r_1$
Ensure: Set of clusters $\mathcal{C} = \{C_1, C_2, ..., C_l\}$
 1: for $n = 1$ to $N$ do
 2:   $C_i = \{W_i\}$
 3: end for
 4: $\mathcal{C} = \{C_1, C_2, ..., C_N\}$.
 5: for $n = r_1$ to $r_2$ do
 6:   $setNGrams(n, \mathcal{W})$.
 7:   for $i = 1$ to $N$ do
 8:     for $j = i + 1$ to $N$ do
 9:       if $Sim(W_i^n, W_j^n) \geq \gamma(W_i^n, W_j^n)$ then
10:         $C_i = C_i \cup C_j$
11:         $\mathcal{C} = \mathcal{C} \setminus \{C_j\}$
12:       end if
13:     end for
14:   end for
15: end for
16: return $\mathcal{C}$
```

3.5 Social Media Treatment

Social networking services have increased their popularity and number of users in the last years. This fact affects this task mainly in two ways. On one hand, as a result of the success of this kind of platforms, a lot of web pages contain terms related to them (e.g. the name of these platforms: Twitter, Facebook, LinkedIn, etc.). On the other hand, for a person name query in a search engine, it usually returns several profiles of such person name that are as much in the same as in different social networking services. These profiles usually are from different people sharing the same name, so they should be in different clusters. Most of the methods of the state of the art do not take into account this fact, usually taking as features tokens from the URL or the title of each web page, which includes the name of these platforms. This practice could lead to add noise to the representation of the web pages.

(Berendsen et al., 2012) proposed the "one in one" baseline to deal with social platform web pages, which creates a singleton cluster for each social web page. However, its main disadvantage is that it does not consider that a same individual could have accounts in several social platforms. A search engine could also return web pages from a social platform which are not profiles, as for example, a group page of Facebook where a person expounds an opinion, in addition to the profile of the same individual in that social platform. In these cases the "one in one" baseline also fails.

We propose a heuristic method that takes into account the limitations of the "one in one" heuristic, letting group social web pages from different platforms and also cluster social web pages from the same social platform. The algorithm that implements our heuristic is $SUPND$ (Social UPND). This algorithm applies $UPND$ with the following restriction: two web pages assigned to the same social networking service cannot be compared. This policy is taken because when a search engine returns several links from the same social platform, they usually refer to different individuals. However, this does not necessarily imply that two web pages belonging to the same social site cannot belong to the same cluster, because they would be compared to other webs pages separately, possibly ending up in the same cluster in a transitive way. For example, giving two web pages from Facebook, let be FB_1 and FB_2, and a non-social web page W, then FB_1 and FB_2 would not be compared, however each FB_i would be compared with W. If $SUPND$ decides to cluster each FB_i with W, then finally both web pages, from the same platform, would be in the same cluster. To identify the social web pages we obtain a list of social media platforms from Wikipedia[4], so when looking at the URL of a web page, we can detect if it corresponds to any of those social media platforms. If it is the case, we assign to that web page its social media site. The computational cost of $SUPND$ is the same of $UPND$.

4 Experiments

In this section we present the corpora of web pages used, the preprocessing of each web page, the experiments carried out and the obtained results.

4.1 Web People Search Collections

WePS is a competitive evaluation campaign that proposes several tasks including resolution of disambiguation on the Web data. In particular, WePS-1, WePS-2 and WePS-3 campaigns provide an evaluation framework consisting in several annotated data sets composed of English person names.

In these experiments we use WePS-1 (Artiles et al., 2007) test corpus composed by 30 English person names and the top 100 search results from Yahoo! search engine; WePS-2 (Artiles et al., 2009a) containing 30 person names and the top 150 search results from Yahoo! search engine; and WePS-3 (Artiles et al., 2010) containing 300 person names and the top 200 search results from Yahoo! All WePS corpora have few social profile web pages, so the impact of this kind of pages in the results of the algorithms is insignificant. We also use the ECIR2012 corpus, which is composed by 33 Dutch person names selected from query logs of a people search engine. For each person name the web pages set is built retrieving several profiles from social media platforms as Facebook, Twitter or LinkedIn, and results returned by Google, Bing and Yahoo! search engines. This data set gives a more real scenario for this task than the WePS ones, because it includes social network profiles of several person sharing the same name.

[4] en.wikipedia.org/wiki/Category:Social_networking_services

4.2 Corpus Preprocessing

Given a person name and a set of web pages, we first discard web pages that do not mention such name using several patterns that take into account the usual structure of person names.

For each not discarded web page, we delete the name and the surname because they appear in all the remaining documents and are the object of the ambiguity. We also delete stop words.

4.3 Results and Discussion

We present our results for all the corpora comparing them with the state of the art systems. The figures in the tables are macro-averaged, i.e., they are calculated for each person name and then averaged over all test cases. For WePS data sets we get the same results for $UPND$ and $SUPND$ algorithms, because these collections include few social networking profiles. The metrics used in this section are the BCubed metrics defined in (Bagga and Baldwin, 1998): BCubed precision (BP), BCubed recall (BR) and their harmonic mean $F_{0.5}(BP/BR)$. (Artiles, 2009) showed that these metrics are accurate for clustering tasks, particularly for person name disambiguation in the Web. We use the Wilcoxon test (Wilcoxon, 1945) to detect statistical significance in the differences of the results considering a confidence level of 95%. In order to compare our algorithm with the WePS better results using the Wilcoxon test, the samples consist in the pairs of values $F_{\alpha=0.5}(BP/BR)$ of each system for each person name.

First, Table 1 shows the results of $UPND$ using n-grams of capitalized tokens and n-grams of NE with WePS-1 training corpus. Experiments include the three weighting functions and the two similarity functions. The results of using n-grams of NE rank below those obtained with n-grams of capitalized tokens in all cases. The Wilcoxon test comparing the results of both representations shows that there are significant differences between them, except TF and TF-IDF with cosine. So we can conclude that in our approach using n-grams of capitalized tokens outperforms the use of n-grams of NE, what confirms our hypothesis.

	TF		z-score		TF-IDF	
Representation	W. Jaccard	Cosine	W. Jaccard	Cosine	W. Jaccard	Cosine
Capitalized n-gram	**0.82**	**0.69**	**0.83**	**0.78**	**0.81**	**0.63**
NE (Stanford NER)	0.77	0.6	0.77	0.72	0.76	0.6

Table 1: $F_{0.5}(BP/BR)$ results of $UPND$ algorithm comparing capitalized n-gram and NE n-gram representations with WePS-1 training corpus.

In Table 2 we show the results of $UPND$ for all WePS test data sets with the three weighting functions and the two similarity measures.

		WePS-1			WePS-2			WePS-3		
		BP	BR	$F_{0.5}$(BP/BR)	BP	BR	$F_{0.5}$(BP/BR)	BP	BR	$F_{0.5}(BP/BR)$
W. Jaccard	TF	0.73	0.77	0.74	0.82	0.82	**0.81**	0.46	0.70	0.50
	z-score	0.70	0.78	0.72	0.80	0.84	**0.81**	0.44	0.72	0.50
	TF-IDF	0.73	0.77	0.73	0.82	0.82	**0.81**	0.46	0.70	0.50
Cosine	TF	0.92	0.61	0.72	0.95	0.61	0.73	0.75	0.45	0.51
	z-score	0.85	0.69	**0.76**	0.91	0.73	**0.81**	0.62	0.56	**0.53**
	TF-IDF	0.94	0.57	0.7	0.96	0.52	0.65	0.79	0.40	0.49

Table 2: Results of $UPND$ algorithm for WePS test data sets.

The combination of z-score with cosine gets the best balance between the values of BP and BR, reaching the highest results of $F_{\alpha=0.5}$ for the three WePS corpora. The combination of TF-IDF with cosine gets the best BP results, but BR results are the lowest. On the other hand, the combination of z-score and Jaccard gets the best BR results, but the BP results are the lowest.

Regarding the significance of the differences between the best results, the improvement between z-score with cosine and z-score with Jaccard is significant in WePS-1 and WePS-3, but not in WePS-2. The improvement between z-score with cosine and Jaccard with TF is significant only in WePS-3.

Thus, we select the combination of z-score as weight function and cosine as similarity function as the most suitable combination for our algorithm. Therefore we use it in the following experiments.

Table 3 shows the results of $UPND$ with WePS-1 test, WePS-2 and WePS-3 corpora in addition to the top ranking systems of the campaigns, and also the results obtained by HAC_Topic system in the case of WePS-1. We include the results obtained by three unsupervised baselines called ALL_IN_ONE, ONE_IN_ONE and Fast AP. ALL_IN_ONE provides a clustering solution where all the documents are assigned to a single cluster, ONE_IN_ONE returns a clustering solution where every document is assigned to a different cluster, and Fast AP applies a fast version of Affinity Propagation described in (Fujiwara et al., 2011) using the function TF-IDF to weight the tokens of each web page, and the cosine distance to compute the similarity.

	System	BP	BR	$F_{0.5}(BP/BR)$
WePS-1	(+) HAC_Topic	0.79	0.85	**0.81** †
	(-) ***UPND***	0.85	0.69	0.76 ●
	(+)(*) CU_COMSEM	0.61	0.83	0.70 †
	(+)(*) PSNUS	0.68	0.73	0.70 †
	(+)(*) IRST-BP	0.68	0.71	0.69 †
	(+)(*) UVA	0.79	0.50	0.61 †
	(+)(*) SHEF	0.54	0.74	0.62 †
	(-) ONE_IN_ONE	**1.00**	0.43	0.57 ●
	(-) Fast AP	0.69	0.55	0.56 †
	(-) ALL_IN_ONE	0.18	**0.98**	0.25 ●
WePS-2	(+) ORACLE_1	0.89	0.83	**0.85** ●
	(+) ORACLE_2	0.91	0.81	**0.85** ●
	(+)(*) PolyUHK	0.87	0.79	0.82
	(+)(*) ITC-UT_1	0.93	0.73	0.81
	(-) ***UPND***	0.91	0.73	0.81 ●
	(+)(*) UVA_1	0.85	0.80	0.81
	(+)(*) XMEDIA_3	0.82	0.66	0.72 †
	(+)(*) UCI_2	0.66	0.84	0.71 †
	(-) ALL_IN_ONE	0.43	**1.00**	0.53 ●
	(-) Fast AP	0.80	0.33	0.41 †
	(-) ONE_IN_ONE	**1.00**	0.24	0.34 ●
WePS-3	(+)(*) YHBJ_2	0.61	0.60	**0.55**
	(-) ***UPND***	0.62	0.56	0.53 ●
	(+)(*) AXIS_2	0.69	0.46	0.50 †
	(+)(*) TALP_5	0.40	0.66	0.44 †
	(+)(*) RGAI_AE_1	0.38	0.61	0.40 †
	(+)(*) WOLVES_1	0.31	0.80	0.40 †
	(-)(*) DAEDALUS_3	0.29	0.84	0.39 †
	(-) Fast AP	0.73	0.30	0.38 †
	(-) ONE_IN_ONE	**1.00**	0.23	0.35 ●
	(-) ALL_IN_ONE	0.22	**1.00**	0.32 ●

Table 3: Results of $UPND$ and the top state of the art systems with WePS corpora: (+) means system with supervision; (-) without supervision and (*) campaign participant. Significant differences between $UPND$ and other systems are denoted by (†); (●) means that in this case the statistical significance is not evaluated.

Our method $UPND$ outperforms WePS-1 participants and all the unsupervised baselines described before. HAC_Topic also outperforms the WePS-1 top participant systems and our algorithm. This system uses several parameters obtained by training with the WePS-2 data set: token weight according to the kind of token (terms from URL, title, snippets, ...) and thresholds used in the clustering process. Note that WePS-1 participants used the training corpus provided to the campaign, the WePS-1 training data, so in this case the best performance of HAC_Topic could be not only because of the different approach, but also because of the different training data set.

Our algorithm obtains significative better results than the WePS-1 top participant results, and HAC_Topic obtains significative better results than it according to the Wilcoxon test. $UPND$ obtains significative better results than IRST-BP system (the third in the WePS-1 ranking), also based on the co-ocurrence of n-grams.

Regarding WePS-2 we add in Table 3 two oracle systems provided by the organizers. These systems use BoW of tokens (ORACLE_1) or bigrams (ORACLE_2) weighted by TF-IDF, deleting previously stop words, and later apply HAC with single linkage with the best thresholds for each person name. We do not include the results of the HAC_Topic system since it uses this data set for training their algorithm.

The significance test shows that the top WePS-2 systems PolyUHK, UVA_1 and ITC-UT_1 obtain similar results than $UPND$, however they use some kind of supervision. The results of all these systems are the closest to the oracle systems provided by the organizers, which know the best thresholds for each person name.

In the case of WePS-3, the organizers did not take into account the whole clustering solution provided by the systems like in previous editions, but only checks the accuracy of the clusters corresponding to two selected individuals per person name. In this case, the first two systems YHBJ_2 and $UPND$ do not have significant difference in their results. Notice that YHBJ_2 system makes use of concepts extracted manually from Wikipedia. Note that $UPND$ also obtains significative better results than DAEDALUS_3, the only one participant that does not use training data.

Regarding the experiments with the ECIR2012 corpus, which contains social profiles, Table 4 shows the results of the two versions of our algorithm and the results of the system of the University of Amsterdam (UvA). As far as we know, no other systems have been tested with this gold standard. $SUPND$ obtains significative better results than $UPND$ due to its special treatment for social web pages. The UvA system outperforms our algorithm $SUPND$ and this improvement is significative. Note that the heuristic for social pages in $SUPND$ outperforms $UPND$ using the "one in one" heuristic.

System	BP	BR	$F_{0.5}(BP/BR)$
(+) UvA (best perf.)	0.90	0.80	**0.83** †
(-) $SUPND$	0.95	0.68	0.78 ●
(-) $UPND$ (one in one)	0.98	0.62	0.74 †
(-) $UPND$	0.74	0.74	0.72 †

Table 4: Results of $SUPND$ and $UPND$ algorithms for ECIR2012 corpus: (+) means system with supervision and (-) without supervision. Significant differences between $SUPND$ and other systems are denoted by (†); (●) means that in this case the statistical significance is not evaluated.

After all these experiments, we can conclude that our approach gets the best results of all the completely unsupervised approaches. Moreover, the precision scores for all collections are very high and confirm that our approach is accurate to get relevant information for characterizing an individual. We also obtain competitive recall results, what lead to a competitive system that carries out person name disambiguation in web search results with minimum human supervision.

5 Conclusions and Future Work

We present a new approach for person name disambiguation of web search results. Our method does not need training data to calculate thresholds to determine the number of different individuals sharing the same name, or whether two web pages refer to the same individual or not. Although supervised approaches have been successful in many NLP and IR tasks, they require enough and representative training data to guaranty the results will be consistent for different data collections, which requires a huge human effort.

The two algorithms proposed provide a clustering solution for this task by means of data-driven methods that do not need learning from data. Our approach is not very expensive in computational cost, obtaining very competitive results in several data sets compared with the best state of the art systems.

Our proposal is based on getting reliable information for disambiguating, particularly long n-grams composed by uppercase tokens. According to our results, this hypothesis has shown successful, getting high precision values and acceptable recall scores. Anyway, we would like to improve recall results without losing of precision, filter out noisy capitalized n-grams, and build an alternative representation for web pages containing all their tokens in lowercase.

We have observed that this task gets harder when we have to deal with social media profiles. A system thought for being used in a real scenario has to take into account this kind of web pages, since they are usually returned by search engines when a user introduces a person name as a query. Most state of the art systems do not deal with this problem. We have proposed in this paper a new heuristic method for processing social platforms profiles for this clustering task.

Person name disambiguation has been mainly addressed in a monolingual scenario, e.g. WePS corpora are English data sets and Dutch the ECIR2012 collection. We would like to address this task in a multilingual scenario. Although search engines return their results taking into account the country of the user, with some queries we can get results written in several languages. This scenario has not been considered by the state of the art systems so far.

Acknowledgements

This work has been part-funded by the Spanish Ministry of Science and Innovation (MED-RECORD Project, TIN2013-46616-C2-2-R) and by UNED Project (2012V/PUNED/0004).

References

Miguel A. Andrade, and Alfonso Valencia. 1998. *Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families*. Bioinformatics, 14:600-607, 1998.

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)*, pages 64–69, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

Javier Artiles. 2009. *Web People Search.* PhD Thesis, UNED University.

Javier Artiles, Julio Gonzalo and Satoshi Sekine. 2009b. Weps 2 Evaluation Campaign: Overview of the Web People Search Clustering Task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

Javier Artiles, Andrew Borthwick, Julio Gonzalo, Satoshi Sekine and Enrique Amigó . 2010. WePS-3 Evaluation Campaign: Overview of the Web People Search Clustering and Attribute Extraction Tasks. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010.*

Amit Bagga and Breck Baldwin. 1998. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 79–85, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics.

Krisztian Balog, Jiyin He, Katja Hofmann, Valentin Jijkoun, Christof Monz, Manos Tsagkias, Wouter Weerkamp, and Maarten de Rijke. 2009. The University of Amsterdam at WePS-2. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

Richard Berendsen, Bogomil Kovachev, Evangelia-Paraskevi Nastou, Maarten de Rijke, and Wouter Weerkamp. 2012. Result Disambiguation in Web People Search. In *Proceedings of the 34th European conference on Advances in Information Retrieval*, ECIR'12, pages 146–157, Berlin, Heidelberg, 2012. Springer-Verlag.

Ying Chen and James Martin. 2007. CU-COMSEM: Exploring Rich Features for Unsupervised Web Personal Named Disambiguation. In *Proceedings of SemEval 2007, Assocciation for Computational Linguistics*, pages 125–128, 2007.

Ying Chen, Sophia Yat Mei Lee and Chu-Ren Huang. 2009. PolyUHK: A Robust Information Extraction System for Web Personal Names. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.

Ergin Elmacioglu, Yee Fan Tan, Su Yan, Min-Yen Kan, and Dongwon Lee. 2007. PSNUS: Web People Name Disambiguation by Simple Clustering with Rich Features. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 268–271, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

Yasuhiro Fujiwara, Go Irie and Tomoe Kitahara. 2011. Fast Algorithm for Affinity Propagation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence(IJCAI)- Volume Three*, 2238–2243, Barcelona, Catalonia, Spain.

Sara Lana-Serrano , Julio Villena-Román , José Carlos González-Cristóbal. 2010. Daedalus at WebPS-3 2010: k-Medoids Clustering using a Cost Function Minimization. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010.*

Zhengzhong Liu, Qin Lu, and Jian Xu. 2011. High Performance Clustering for Web Person Name Disambiguation using Topic Capturing. In *International Workshop on Entity-Oriented Search (EOS)*, 2011.

Chong Long and Lei Shi. 2010. Web Person Name Disambiguation by Relevance Weighting of Extended Feature Sets. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*.

Gideon S. Mann. 2006. *Multi-Document Statistical Fact Extraction and Fusion*. PhD thesis, Johns Hopkins University, Baltimore, MD, USA, 2006. AAI3213760.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

Octavian Popescu and Bernardo Magnini. 2007. IRST-BP: Web People Search Using Name Entities In *In Proceedings of SemEval 2007, Assocciation for Computational Linguistics*, pages 195–198, 2007.

Frank Wilcoxon. 1945. *Individual Comparisons by Ranking Methods*, volume 1 (6). Biometrics Bulletin, December 1945.