ATMC team at M-WePNaD task

Agustín D. Delgado

Universidad Nacional de Educacin a Distancia (UNED)
Juan del Rosal, 16, 28040 - Madrid
agustin.delgado@lsi.uned.es

Abstract. This paper presents our participation in the task Multilingual Web Person Name Disambiguation (M-WePNaD) at IBEREVAL 2017 workshop. Given a ranking of search results written in different languages retrieved by a search engine when looking for a person name, the goal of the task is to group the web pages according to the individual they refer to. We have grouped the search results by means of a clustering algorithm which does not need any kind of prior information. On the other hand, we deal with multilingualism by two different ways. The first one just use a machine translation tool. The second one is a method to compare search results written in different languages which is based on giving a special role to those features written the same way in several languages. Both approaches get similar results, but the second one is more efficient because it avoids additional preprocessing caused by the translation of the search results.

1 Introduction

The disambiguation of person names on the Web has been addressed in last years due to two main reasons: (i) Person names are a kind of named entities (NEs) specially ambiguous, so that their disambiguation has been studied in several scenarios like Cross-Document Coreference Resolution [3, 11], Entity Linking and wikification [12, 13] or author name disambiguation [10, 16]; and (ii) The search scenario on the Web presents several challenges: web pages do not talk about an specific topic; search results could not have in common an specific structure as happens with news, scientific papers or references; and the proposed methods must be efficient due to users expect quick responses to their queries.

Person name disambiguation on the Web has been addressed as a clustering problem composed by two phases. The goal of the first phase is to represent the search results by means of suitable features to identify and distinguish different individual with the same name. On the other hand, the second phase is to apply a clustering algorithm to group the search results according to the individual they refer to. In particular, the best systems of the state-of-the-art represent the search results with a rich selection of features from different nature and groups the web pages by means of the Hierarchical Agglomerative Clustering (HAC) algorithm after learning a similarity threshold by means of training data [2]. However, some authors have pointed out that the results obtained by HAC are very sensitive with respect to little variations of the similarity threshold, so this methodology is not robust.

On the other hand, this problem has been addressed assuming that all the search results are written in the same language. However, the search engines are able to retrieve web pages written in several languages and there are increasingly web pages written in different languages due to the popularization of the Internet in non-English speaking countries [18]. There are few proposals that take into account the presence of multilingualism in this problem. For instance, in [15] is presented a method based on extracting biographical information of the individuals, like birth dates and places. For this purpose, the authors propose to learn several patterns of each biographical fact in several languages by means of training data. However, this approach needs enough training data for each biographical fact in each language, which requires a huge human effort. On the other hand, in [14] the authors claim that Latent Dirichlet Allocation (LDA) is able to deal with the problem for any language. These authors used a collection that contains news written in English, Spanish, Bulgarian and Romanian to check the suitability of their approach in several languages. Nevertheless, the web pages associated to each entity are written in the same language, so the disambiguation process is not multilingual.

In this paper, we present several methods to deal with multilingualism in person name disambiguation on the Web. To this end, we have used a data set called MC4WePS[1] [17] provided by the M-WePNaD organizers. First, we detail our approaches in Section 2. Next, we present the results in Section 3 and we discuss them in Section 4. Finally, Section 5 presents some conclusions and future lines of work.

2 Methods

This section presents four methods to solve the M-WePNaD task, which could be divided into two kinds: those that just take into account the original content of the web pages; and those that employ a machine translation tool. First, Subsection 2.1 describes the clustering algorithm used to group the web pages according to the individual they refer to. Right after, we detail the translation process and the preprocessing of the search results in Subsections 2.2 and 2.3 respectively. Finally, Subsection 2.4 presents several approaches to compare search results written in different languages.

2.1 Clustering Algorithm

We have used the algorithm *Adaptive Threshold Clustering* (ATC) [7, 8] to group the search results. ATC is composed by three phases and its grouping strategy is the following: the goal of the two first phases is to obtain initial cohesive clusters with a high value of precision, while the third phase merges them in order to improve the recall score. The phases of ATC are briefly described as follows:

[1] http://nlp.uned.es/web-nlp/resources

- Phase 1 (grouping by links): the search results are grouped if they are linked or they share some link under the assumption that they refer the same individual in that case. Therefore, each web page is represented by its URL and its links. Note that the M-WePNaD organizers provide the URL of all the search results in their metadata files.
- Phase 2 (UPND algorithm): the search results are grouped by means of the algorithm UPND [5]. In this phase, the web pages are represented by means of their *capitalized* 3*-grams*, which are described as a sequence of three consecutive words with their first letter written in uppercase. This kind of features has shown suitable to distinguish between different individuals [6].
- Phase 3 (fusion of clusters): merge of the most similar clusters generated in the previous phases. The clusters are represented as bag of words by means of their centroids. However, some features are filtered of the centroids according to the following properties: (i) they have a low document frequency within the cluster; and (ii) they appear in most of the clusters. In this phase the search results are represented by means of their 1-grams due to these features allow to represent as much search results as possible unlike the capitalized 3-grams.

We have used the configuration of ATC described in [8]: these authors show that the results are not affected with respect to the function used to weight the capitalized 3-grams, so they are weighted with the binary function because it is the most simple one. However, in the case of the 1-grams, the TF-IDF function gets better results. On the other hand, ATC compares search results W_i and W_j by means of their cosine similarity $sim(W_i, W_j)$ and a mathematical function $\gamma(W_i, W_j)$ called *adaptive threshold* which returns a similarity threshold which depends on the search results characteristics and their number of shared features. The web pages are merged if $sim(W_i, W_j) > \gamma(W_i, W_j)$. This way, ATC is able to estimate the number of clusters, so it does not need any prior information to group the search results unlike HAC or k-means algorithms.

On the other hand, the presence of web pages from social media platforms could lead to obtain worse performance [4]. Thus, we have applied an heuristic to treat social media platforms [7], which do not allow comparisons between web pages from the same social platform. In addition, this heuristic is extended to web pages of people search engines in the phase 1 because these web pages usually contain links to profiles of different social platforms of people with the same name, so they could lead to merge incorrectly web pages when they are represented by their links.

2.2 Translation Process

Some experiments are based on the use of a machine translation tool. This Subsection describes the translation process conducted for these runs for each person name.

First, we have to select the *anchor language* to translate the web pages to. As the computational cost must be light in a web search scenario, we have decided to translate as few web pages as possible. Thus, we identify the anchor language as the most frequent language of the search results contained in the ranking. Although the M-WePNaD organizers provide the language of each search result annotated by experts, we have used a language identification tool available in the Internet[2] that uses a Naive Bayes classifier which looks to sequences of characters within the text to detect the language. We have evaluated the performance of the language detector taking into account the language annotations from the experts obtaining 96.17% accuracy.

On the other hand, we have used the translation service provided by the Russian technology company Yandex [3]. This tool is able to translate documents from 94 different languages, including those identified by the language detector. This tool uses statistical techniques to translate the documents by means of several dictionaries and modeling each language with web pages written in several languages, for instance, taking the version of the web site of companies in different languages and comparing them. This translator could not perform correctly due to mistakes made by the language detector. For instance, a Spanish web page that has been detected as Catalan is not entirely translated because the translator does not find Catalan words in the text with the exception of the shared vocabulary between the two languages.

2.3 Preprocessing

The preprocessing of the search results when using the machine translation tool for each person name is the following: we obtain the plain text of the search results using the parsers provided by the library *TiKa Apache*[4]. This tool is also able to obtain the links of the search results used in the phase 1 of ATC. Right after we identify the language of each search result by means of the language detection tool. Next, we translate to the anchor language those search results written in other languages. Then, we split the plain texts into sentences and we delete the stop words of the anchor language because all the search results are written in the same language after the translation step. In addition, we delete the person name due to it is the query so we assume that all the search results contain them.

Those experiments which do not use the machine translation tool conduct the same preprocessing with the exception of two differences: (i) we do not translate any search result; and (ii) we delete the stop words of the language identified by the language detector.

After the preprocessing phase, we extract the textual features of each sentence used by ATC: capitalized 3-grams and 1-grams. Finally, we remove those features which only appear in one search result of the ranking.

[2] https://code.google.com/p/language-detection/

[3] https://www.yandex.com/

[4] https://tika.apache.org/

2.4 Approaches

We have conducted several experiments based on ATC which take different representations of the search results or apply different policies when comparing them:

- Run 1 (ATC): the search results are represented by means of their original textual features without using any translation resource.
- Run 2 (ATC+TRAD): we translate to the anchor languages those search results written in other languages.
- Run 3 (ATC+CENT_TRAD): we translate separately to the anchor language only the 1-grams contained in the centroids used to represent the clusters in the last phase of the algorithm.
- Run 4 (ATMC): we compare the search results taking into account those features written the same way in different languages without using any translation resource. We have called this run *Adaptive Threshold for Multilingual Clustering* (ATMC) [9]. Below we detail this method.

Runs 2 and 3 allow us to study the suitability of applying a machine translation tool to compare the search results. In particular, Run 3 allows us to study the suitability of translating some words separately with respect to translate the whole document as Run 2 does. On the other hand, Runs 1 and 4 do not use any translation resource to make the disambiguation process lighter in terms of cost because it avoids an additional phase dedicated to translate the web pages. This is desirable in problems related to searching on the Web due to users want quick responses to their queries. Run 1 just applies the ATC algorithm [8] using the features extracted of the original content the web pages, while Run 4 compares the documents written in different languages providing more importance to those features which are written the same way in both languages.

ATMC compares search results written in different languages giving a special role to those of their features orthographically identical in several languages. This usually happens with NEs as organizations or person names. However, it also happens with other kind of information which is not usually detected as NEs, for instance, titles of films, books, TV shows, papers, and so on, which could be useful to identify an individual.

Let be $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$ the set of search results returned by a search engine when looking for a person name. We denote as F_i to the set of features of the search result W_i that is written in the language l_i. On the other hand, $L(\mathcal{W}) = \bigcup_{i=1}^{N}\{l_i\}$ denotes the set of languages of the search results contained in $\mathcal{W}$. We can tag each feature $f \in \mathcal{F}$ with the set of languages of the search results where it appears computing $L(f) = \{l_i \in L(\mathcal{W}) | f \in F_i\} \subseteq L(\mathcal{W})$. Then, any feature $f \in F_i$ must hold that $l_i \in L(f)$. Given two features $f, f' \in \mathcal{F}$, they are *comparable features* if $L(f) \cap L(f') \neq \emptyset$. The set of comparable features of W_i and W_j are defined as follows:

$$F_{i,j} = \{f_i \in F_i | l_j \in L(f_i)\} \subseteq F_i \tag{1}$$

$$F_{j,i} = \{f_j \in F_j | l_i \in L(f_j)\} \subseteq F_j \tag{2}$$

This definition can be easily generalized for clusters of search results just taking into account the set of languages of the web pages contained in each cluster. In addition, note that $l_i = l_j$ implies that $F_i = F_{i,j}$ and $F_j = F_{j,i}$, so this guarantees that comparing web pages taking $F_{i,j}$ and $F_{j,i}$ has no effect in the monolingual scenario. On the other hand, $l_i \neq l_j$ implies that we do not compare the search results taking into account features that we already know are not shared by both web pages by means of the language detection, so it is more probable that they can be grouped than using all their features. This means that if we only use comparable features to compare the search results then we give more benefit to those comparisons between web pages written in different languages. In order to avoid this effect as much as possible, we propose to balance the comparison of the search results taking into account all their features and their comparable features by means of the following formulas:

$$sim_{ML}(W_i, W_j) = \alpha_{i,j} \cdot sim(F_i, F_j) + (1 - \alpha_{i,j}) \cdot sim(F_{i,j}, F_{j,i}) \tag{3}$$

$$\gamma_{ML}(W_i, W_j) = \alpha_{i,j} \cdot \gamma(F_i, F_j) + (1 - \alpha_{i,j}) \cdot \gamma(F_{i,j}, F_{j,i}) \tag{4}$$

where $\alpha_{i,j} = \frac{|F_{i,j}|+|F_{j,i}|}{|F_i|+|F_j|}$ is the proportion of comparable features with respect to all the features of the compare search results. A high value of $\alpha_{i,j}$ means that most of features are comparable, so the similarity and the adaptive threshold values would be similar to the ones used by ATC assumming a monolingual scenario. On the other hand, if $\alpha_{i,j}$ has a low value, then few features are comparable so they are more weighted when comparing search results written in different languages.

3 Results

Tables 1 and 2 show the results obtained by the proposed approaches with the test collection and the baselines ALL IN ONE and ONE IN ONE provided by the M-WePNaD organizers. In particular, Table 1 shows the results obtained when taking into account only the search results related to some individual according to the annotators, while Table 2 shows the results obtained when considering all the search results. On the other hand, the baseline ALL IN ONE just returns one cluster which contains all the search results for each person name, while ONE IN ONE returns each search results as a singleton cluster for each person name. The evaluation metrics used are reliability (R) and sensibility (S) [1], and their F-measure ($F_{0.5}$), which weights equally both metrics.

4 Discussion

The baseline ALL IN ONE improves the results of ONE IN ONE which means that most individuals in the collection have associated several web pages. The

Run	R	S	$F_{0.5}$
ALL IN ONE	0.47	0.99	0.54
ONE IN ONE	1.00	0.32	0.42
Run 1 (ATC)	0.79	0.83	0.79
Run 2 (ATC+TRAD)	0.82	0.79	0.80
Run 3 (ATC+CENT_TRAD)	0.80	0.84	0.81
Run 4 (ATMC)	0.79	0.85	0.81

Table 1. Results obtained by the proposed approaches with the test collection of the M-WePNaD task taking into account related search results.

Run	R	S	$F_{0.5}$
ALL IN ONE	0.47	1.00	0.56
ONE IN ONE	1.00	0.25	0.36
Run 1 (ATC)	0.78	0.73	0.74
Run 2 (ATC+TRAD)	0.82	0.69	0.73
Run 3 (ATC+CENT_TRAD)	0.79	0.74	0.75
Run 4 (ATMC)	0.78	0.75	0.75

Table 2. Results obtained by the proposed approaches with the test collection of the M-WePNaD task taking into account all the search results.

tables also show that the proposed approaches improve the results of both baselines. However, the results between the four approaches are close. This could be explained because of two reasons: several person names in the collection are monolingual (9 of 35) and we translate as less web pages as possible, which modifies the representation of few web pages. In particular, the results of ATC are slightly worse than the ones obtained by the experiments that use the machine translation tool (ATC+TRAD and ATC+CENT_TRAD) and ATMC. On the one hand, this means that the translation process has a positive impact. In particular, ATC+CENT_TRAD is more suitable because it translate a lower amount of text. This experiment obtains a lower reliability score with respect to ATC+TRAD but it gets better results of sensibility. This is explained because when the words are translated separately (ATC+CENT_TRAD) the translator always return the same output, but when we translate the whole texts (ATC+TRAD), the translation of each word could be different depending on the context, so the documents share more vocabulary in the case of ATC+CENT_TRAD which leads to a higher number of groupings. Finally, ATMC slightly improves ATC+ORIGINAL although both approaches use the original features. This means that the proposed method to compare web pages written in different languages is suitable. In particular, ATC and ATMC get the same reliability score, but ATMC obtains higher sensibility, which means that ATMC is able to group correctly a higher number of search results without loss of precision. In addition, ATC+TRAD and ATC+CENT_TRAD do not improve the results of ATMC although they use a machine translation tool. Then, ATMC is a better choice because it does not need an additional preprocessing step for translating the search results which necessarily increase the processing time of the disambiguation process. Note that this is desirable in a scenario involving searching on the Web due to users expect response as soon as possible.

Regarding the results of both tables, the baseline ALL IN ONE is the only experiment that improves its results when considering all the web pages, including the not related search results. These web pages have been identified by the annotators according to several criteria, for instance, they do not mention any individual with the person name given as query, or they refer to other categories of NEs which are not person names, as happens with *John Fitzgerald Kennedy International Airport* instead of the former president of the United States. These not related search results are grouped by the annotators in the same cluster for each person name although they could refer to different people, so this situation benefits ALL IN ONE but has a negative impact to ONE IN ONE. On the other hand, the proposed approaches do not identify and group the not related search results, so they get worse results when considering all the web pages.

5 Conclusions

This paper has described our participation in the M-WePNaD task at IBEREVAL 2017 workshop, which goal is to address person name disambiguation on the Web in a multilingual scenario. We have proposed four approaches to address the multilingualism in the problem. Two of them are based on the use of a machine translation tool while the other ones do not use any translation resource in order to avoid additional preprocessing steps. On the one hand, the use of a translator improves slightly the results obtained using the original features. On the other hand, we have seen that the comparable features are useful to compare web pages written in different languages without the need of translation resources. As future work, we want to explore how to enrich the representation by means of comparable features. For instance, this representation could be extended identifying features written similarly in different languages in addition to those written orthographically the same. Those features could be detected by means of NEs alignment techniques and cognate identification methods. In addition, a future line of work is to detect not related search results due to they have a negative impact in our methods when we consider the whole ranking of web pages. This kind of search results could be identified by means of checking if they mention the person name given as query, and those mentions are not other categories of NEs than person names.

Acknowledgment

This work has been part-funded by the Spanish Ministry of Science and Innovation (MAMTRA-MED Project, TIN2016-77820-C3-2-R and MED-RECORD Project, TIN2013-46616-C2-2-R).

References

1. Enrique Amigó & Julio Gonzalo & Felisa Verdejo: A General Evaluation Measure for Document Organization Tasks Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 643–652, 2013. http://doi.acm.org/10.1145/2484028.2484081.
2. Javier Artiles: Web People Search. PhD Thesis, E.T.S. Ingeniería Informática, UNED, 2009. http://e-spacio.uned.es/fez/eserv/tesisuned:IngInf-Jartiles/Documento.pdf.
3. Amit Bagga & Breck Baldwin: Entity-based Cross-document Coreferencing Using the Vector Space Model. Proceedings of the 17th International Conference on Computational Linguistics - vol. 1, pp. 79–85, 1998. University of Amsterdam (2015). http://dx.doi.org/10.3115/980451.980859.
4. Richard Berendsen: Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation. PhD Thesis. University of Amsterdam (2015). http://doi.acm.org/10.1145/2484028.2484081.
5. Agustín D. Delgado & Raquel Martínez & Víctor Fresno & Soto Montalvo: A Data Driven Approach for Person Name Disambiguation in Web Search Results. Proceedings of the 25th International Conference on Computational Linguistics, pp. 301–310, 2014. http://aclweb.org/anthology/C/C14/C14-1030.pdf.
6. Agustín D. Delgado & Raquel Martínez & Soto Montalvo & Víctor Fresno: An Unsupervised Algorithm for Person Name Disambiguation in the Web. Procesamiento del Lenguaje Natural, 53:51–58, 2014. http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5042.
7. Agustín D. Delgado & Raquel Martínez & Soto Montalvo & Víctor Fresno: Tratamiento de redes sociales en desambiguacin de nombres de persona en la web. Procesamiento del Lenguaje Natural, 57:117-124, 2016. http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5344.
8. Agustín D. Delgado & Raquel Martínez & Soto Montalvo & Víctor Fresno: Person Name Disambiguation in the Web Using Adaptive Threshold Clustering. Journal of the Association for Information Science and Technology, 2017. https://doi.org/10.1002/asi.23810.
9. Agustín D. Delgado: Desambiguación de nombres de persona en la Web en un contexto multilingüe. PhD Thesis, E.T.S. Ingeniería Informática, UNED, 2017.
10. Johanna Geiß & Michael Gertz: With a Little Help from My Neighbors: Person Name Linking Using the Wikipedia Social Network. Proceedings of the 25th International Conference Companion on World Wide Web, pp. 985–990, 2016. http://dx.doi.org/10.1145/2872518.2891109.
11. Chung Heong Gooi & James Allan: Cross-Document Coreference on a Large Scale Corpus. Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 9–16, 2004. http://aclweb.org/anthology/N/N04/N04-1002.pdf
12. Toni Grütze & Gjergji Kasneci & Zhe Zuo & Felix Naumann: Bootstrapping Wikipedia to answer ambiguous person name queries. Workshops Proceedings of the 30th International Conference on Data Engineering Workshops, pp. 56.61. 2014. http://dx.doi.org/10.1109/ICDEW.2014.6818303.
13. Zhengyan He & Houfeng Wang & Sujian Li: The Task 2 of CIPS-SIGHAN 2012 Named Entity Recognition and Disambiguation in Chinese Bakeoff. Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing, pp. 108–114. 2012. http://www.aclweb.org/anthology/W12-6321.

14. Zornitsa Kozareva & Sujith Ravi: Unsupervised Name Ambiguity Resolution Using a Generative Model. Proceedings of the First Workshop on Unsupervised Learning in NLP, pp. 105–112, 2011. http://dl.acm.org/citation.cfm?id=2140458.2140471.
15. Gideon S. Mann & David Yarowsky: Unsupervised Personal Name Disambiguation. Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4, pp. 33–40. http://dx.doi.org/10.3115/1119176.1119181.
16. Fakhri Momeni & Philipp Mayr: Using Co-authorship Networks for Author Name Disambiguation. Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries (JCDL 2016), pp. 261–262. http://doi.acm.org/10.1145/2910896.2925461.
17. Soto Montalvo & Raquel Martínez & Leonardo Campillos & Agustín D. Delgado & Víctor Fresno & Felisa Verdejo: MC4WePS: a multilingual corpus for web people search disambiguation Language Resources and Evaluation (2016). http://dx.doi.org/10.1007/s10579-016-9365-4.
18. Daniel Pimienta & Daniel Prado & Álvaro Blanco: Twelve years of measuring linguistic diversity in the Internet: balance and perspectives. UNESCO publications for the World Summit on the Information Society (2009). http://unesdoc.unesco.org/images/0018/001870/187016e.pdf.