Procesamiento del Lenguaje Natural, Revista nº 57, septiembre de 2016, págs. 117-124 recibido 28-03-2016 revisado 20-04-2016 aceptado 03-05-2016

Tratamiento de Redes Sociales en Desambiguación de Nombres de Persona en la Web

Treatment of Social Media in Person Name Disambiguation in the Web

Agustín D. Delgado[1], **Raquel Martínez**[1], **Soto Montalvo**[2], **Víctor Fresno**[1]
1. Universidad Nacional de Educación a Distancia (UNED), Juan del Rosal, 16, 28040 - Madrid
2. Universidad Rey Juan Carlos (URJC), Tulipán, S/N, 28933 - Móstoles
agustin.delgado@lsi.uned.es, raquel@lsi.uned.es, soto.montalvo@urjc.es, vfresno@lsi.uned.es

Resumen: En este trabajo presentamos dos heurísticas para tratar páginas web correspondientes a redes sociales en el problema de desambiguación de nombres de persona en la Web. Este problema consiste en agrupar las páginas web proporcionadas por un motor de búsqueda al consultar un nombre de persona según el individuo al que se refieren. Aunque estas páginas web pueden afectar negativamente en la agrupación de los resultados, la mayoría de sistemas del estado del arte no tienen en cuenta su papel en este problema. Hemos evaluado nuestras heurísticas con dos colecciones que contienen este tipo de páginas web. Para agrupar las páginas web hemos utilizado una extensión de un algoritmo del estado del arte. Ambas heurísticas obtienen mejoras cuando hay un número elevado de páginas sociales y el algoritmo propuesto es más independiente del nivel de ambigüedad de los nombres de persona que otros propuestos por el estado del arte.
Palabras clave: búsqueda de personas en la web, redes sociales, clustering

Abstract: In this work, we present two heuristics to treat web pages from social networks for person name disambiguation in the Web. This problem consists in clustering the results provided by a search engine when the query is a person name according to the individual they refer to. Although these web pages could negatively affect when grouping the results, most of the systems in the state-of-the-art do not take into account their role in this problem. We have evaluated our heuristics with two collections that contain this kind of web pages. We have used an extension of an algorithm of the state of the art to cluster the web pages. Both heuristics get improvements when there is a high number of social web pages, and the proposed algorithm is more independent with respect to the ambiguity degree of person names than other ones in the state of the art.
Keywords: web people search, social media, clustering

1 Introducción

La desambiguación de nombres de personas es un reto dentro del Procesamiento del Lenguaje Natural. Un escenario real donde es de gran ayuda diferenciar entre distintos individuos con el mismo nombre lo encontramos en los motores de búsqueda. Cuando un usuario quiere buscar información sobre una persona en particular, se encuentra con un ranking de links que pueden hablar de diferentes personas que comparten el mismo nombre, de manera que debe seleccionar del ranking aquellos resultados del individuo de su interés. Pese a que entre un 11-17 % de consultas realizadas por usuarios contienen un nombre de persona (Artiles et al., 2010), los motores de búsqueda más conocidos (Google, Yahoo!, Bing) solo proveen herramientas de desambiguación para las celebridades mediante sus grafos de conocimiento (*knowledge graphs*). Por otra parte, recientemente han aparecido varios buscadores de pago especializados en buscar personas (spokeo.com, pipl.com, intelius.com), lo que de-

* Este trabajo ha sido subvencionado por el Ministerio de Ciencia e Innovación [MED-RECORD Project, TIN2013-46616-C2-2-R] y el grupo CVIP de la URJC.

muestra el impacto de este problema en Internet.

Una de las mayores dificultades de este problema reside en que la temática tratada en las páginas web de un mismo individuo puede ser heterogénea. Por ejemplo, en páginas profesionales normalmente encontramos información laboral de una persona, mientras que en blogs o perfiles de redes sociales es habitual encontrar información personal, opiniones, hobbies, etc. Debido a la irrupción de las plataformas de redes sociales en los últimos años, cuando consultamos un nombre de persona en un motor de búsqueda es bastante habitual obtener enlaces a perfiles de este tipo de plataformas. A pesar de ello, la mayoría de los sistemas del estado del arte no tienen en cuenta este factor puesto que han sido evaluados en corpora que contienen un número reducido de webs de este tipo. La aparición de este tipo de páginas web puede afectar negativamente en este problema y, por tanto, deben ser tratadas de manera especial (Berendsen, 2015).

La principal contribución de este trabajo consiste en la propuesta de dos heurísticas para tratar las páginas de redes sociales en la desambiguación de nombres de personas en la Web. Hemos evaluado nuestras heurísticas en dos corpora de desambiguación de nombres de personas que incluyen resultados de redes sociales. En ambas colecciones, el uso de nuestras aproximaciones mejora los resultados obtenidos cuando no se tienen en cuenta este tipo de páginas web.

El resto del artículo se organiza del siguiente modo. En la sección 2, comentamos brevemente el estado del arte en desambiguación de nombres de personas. A continuación, en la sección 3 presentamos el algoritmo de clustering que hemos utilizado para agrupar las páginas web según el individuo al que se refieren. Posteriormente, la sección 4 presenta nuestras propuestas de tratamiento de las redes sociales en este problema. La sección 5 presenta y analiza los resultados obtenidos. Finalmente, en la sección 6 se presentan conclusiones y líneas de trabajo futuro.

2 *Estado del Arte*

Las campañas de evaluación WePS[1] (*Web People Search*) plantearon el problema de desambiguación de personas en la Web, publicando varios corpora anotados. Este marco de evaluación se ha convertido en un referente, puesto que ha permitido realizar estudios comparativos sobre el rendimiento de diferentes sistemas.

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

La desambiguación de nombres de personas en la web se ha tratado como un problema de clustering en el estado del arte, donde el objetivo es estimar el número de individuos diferentes mencionados en el ranking de páginas web, y organizar en grupos dichos resultados según el individuo particular al que se refieren. Los sistemas propuestos dividen el problema en dos pasos: (1) representación de páginas web, donde el objetivo es seleccionar rasgos adecuados para representar las páginas web; (2) aplicar un algoritmo de clustering para agrupar los resultados.

En cuanto a la representación de las páginas web, los sistemas más competitivos han usado el modelo de espacio vectorial. Algunos trabajos (Balog et al., 2009; Grüetze, et al., 2014) han concluido que el uso de modelos probabilísticos logran resultados más pobres. Los rasgos más utilizados han sido bolsas de palabras, Entidades Nombradas (ENs) y sintagmas nominales. Según (Artiles, Amigó y Gonzalo, 2009a) el uso de rasgos lingüísticos como las ENs, no otorgan ventajas sustanciales con respecto a usar rasgos que no requieren pre-procesamientos lingüísticos. Por otra parte, algunos autores (Nuray-Turan, Kalashnikov y Mehrotra, 2012; Delgado et al., 2014a) destacan la precisión de los n-gramas a la hora de agrupar adecuadamente las páginas web. Finalmente, algunos sistemas competitivos (Chen, Yat Mei Lee y Huang, 2012; Nuray-Turan, Kalashnikov y Mehrotra, 2012; Xu et al., 2015) enriquecen la representación tomando tokens de las URLs, snippets, extrayendo información de Wikipedia, realizando consultas adicionales a un buscador, o aplicando extracción de atributos para conseguir datos biográficos de los individuos.

En cuanto a los algoritmos de clustering utilizados, las campañas WePS (Artiles et al., 2010) destacan que sus mejores participantes han usado métodos basados en el algoritmo jerárquico aglomerativo (HAC). Esta conclusión se ha visto corroborada en trabajos posteriores (Liu, Lu y Xu, 2011; Xu et al., 2015), donde se presentan sistemas que obtienen mejores resultados y están basados en versiones de este algoritmo. La mayoría de los anteriores sistemas requieren de datos de entrenamiento para obtener un valor de umbral que corte el dendograma devuelto por HAC. Sin embargo, el comportamiento de HAC es muy sensible a dicho valor y puede conllevar resultados sesgados según la naturaleza del corpus de entrenamiento. Finalmente, otros trabajos

(Delgado et al., 2014a; Xu et al., 2015) presentan sistemas que evitan el uso de datos de entrenamiento. En particular, (Delgado et al., 2014a) presenta el algoritmo de clustering UPND, basado en la compartición de n-gramas y una función de umbral adaptada a los documentos que se comparan, y cuyo uso evita la necesidad de datos de entrenamiento de HAC.

Las colecciones proporcionadas por WePS contienen un número muy pequeño de páginas web correspondientes a redes sociales. Sin embargo, es común que este tipo de páginas web aparezcan cuando se consulta un nombre de persona en un buscador. En (Berendsen, 2015), se estudia el impacto de este tipo de páginas web, concluyendo que su aparición puede llevar a obtener agrupaciones incorrectas y deben ser tratadas de manera diferenciada. Propone un método que distingue las webs sociales del resto, agrupando de forma separada las páginas web no sociales y las sociales. Las páginas no sociales se agrupan mediante HAC aplicando un valor de umbral obtenido mediante datos de entrenamiento, mientras que las páginas sociales se dejan en clusters unitarios. Finalmente, propone un algoritmo de mezcla de ambos grupos basado en penalizar aquellos clusters que contienen páginas sociales. Su propuesta la prueba utilizando un nuevo corpus que contiene un número considerable de páginas web sociales.

3 Algoritmo de Clustering

El algoritmo de clustering que hemos usado para agrupar las páginas web consiste en una extensión del método UPND presentado en (Delgado et al., 2014a), el cual se basa, por un lado, en la compartición de n-gramas largos compuestos por palabras escritas en mayúsculas y, por otro lado, en el uso de funciones que computan automáticamente un umbral cuando se comparan dos páginas web. En la Sección 3.1 detallamos los rasgos adicionales que utiliza nuestra propuesta para mejorar la representación de las páginas web. A continuación, en la Sección 3.2 presentamos una nueva función de umbral y, finalmente, el método propuesto se describe en la Sección 3.3.

3.1 Representación de páginas web

Puesto que tomamos como punto de partida el algoritmo UPND, nuestro método asume las dos hipótesis sobre representación de las páginas web de este algoritmo: (H1) La coaparición de n-gramas permite decidir si dos documentos hablan de un mismo individuo. Además, cuanto mayor sea el valor n, más probable es la afirmación anterior. (H2) Las palabras en mayúsculas aportan información especialmente útil a la hora de desambiguar entre diferentes individuos. Combinando ambas hipótesis, se asume que la coaparicón de n-gramas en mayúsculas es un buen indicador para decidir si dos documentos se refieren al mismo individuo. Hemos añadido una hipótesis adicional: (H3) Dos páginas web de un ranking hablan del mismo individuo si están enlazadas entre sí, esto es, una de ellas contiene como link la URL de la otra.

Una limitación de la combinación de las hipótesis (H1) y (H2) es que quedan páginas web sin representar o infrarepresentadas, por ejemplo, aquellas escritas principalmente en minúsculas. Para evitar este problema, tras aplicar UPND, se ejecutan dos fases en las que los documentos se representan respectivamente mediante 1-gramas de palabras mayúsculas y 1-gramas de todas las palabras.

3.2 Umbrales Adaptativos

A la hora de comparar dos páginas web, el algoritmo UPND emplea una función de umbral que depende únicamente del contenido de las páginas web con el objetivo de evitar el cálculo de umbrales mediante datos de entrenamiento. Las funciones propuestas en (Delgado et al., 2014a; Delgado et al., 2014b) no dependen del valor n de los n-gramas, pese a que se asume que cuanto más largos sean los que comparten dos páginas web, mayor es la probabilidad de que ambas hablen del mismo individuo. Por ello, para cumplir formalmente la hipótesis (H1) proponemos una nueva función de umbral, de manera que decrece el umbral si el valor n de los n-gramas aumenta:

$$\gamma(W_i^n, W_j^n) = \frac{\gamma_{max}(W_i^n, W_j^n) + \gamma_{min}(W_i^n, W_j^n)}{2 \cdot n} \quad (1)$$

donde W_i es una página web, W_i^n denota a su bolsa de n-gramas asociada y

$$\gamma_{max}(W_i^n, W_j^n) = \frac{min(|W_i^n|, |W_j^n|) - |W_i^n \cap W_j^n|}{max(|W_i^n|, |W_j^n|)} \quad (2)$$

$$\gamma_{min}(W_i^n, W_j^n) = \frac{min(|W_i^n|, |W_j^n|) - |W_i^n \cap W_j^n|}{min(|W_i^n|, |W_j^n|)} \quad (3)$$

Dada una función de similitud sim, la *condición de agrupamiento* empleada por el algoritmo UPND para agrupar dos páginas web W_i y W_j es la siguiente: $sim(W_i^n, W_j^n) > \gamma(W_i^n, W_j^n)$. Cuando se cumple la anterior condición de agrupamiento el algoritmo une los clusters a los que pertenecen ambos documentos, por lo que UPND agrupa los documentos de manera transitiva.

3.3 Algoritmo Propuesto

El algoritmo propuesto se muestra en Algoritmo 1. Inicialmente, se agrupan las páginas web que están enlazadas comparando sus links con sus URLs (según H3) mediante el método $groupByLinks$. A continuación, se aplica el algoritmo UPND tomando 3-gramas en mayúsculas. Posteriormente se ejecutan dos fases adicionales que usan 1-gramas para representar las páginas web, evitando así el problema de baja representación de páginas web de UPND. La primera fase extra agrupa los clusters obtenidos previamente usando 1-gramas en mayúsculas. Esto se justifica por dos razones: (i) como los rasgos usados por UPND son muy discriminantes, se asume que puede devolver varios clusters que se refieren al mismo individuo y (ii) se toman rasgos en mayúsculas asumiendo la hipótesis (H2) de UPND. Finalmente, la segunda fase extra agrupa las páginas web que no se han agrupado con anterioridad (*isolated pages*, conjunto I), tomando como rasgos todos los 1-gramas. Para cada *isolated page*, se calcula su similitud con los clusters no-unitarios existentes (conjunto C_{aux}), y se agrupa en el más similar tal que cumpla la condición de agrupamiento de UPND (método $bestCluster$). En caso de no agruparse en ningún cluster, la propia página web *isolated* se trata como un cluster más, de manera que se permite que las páginas *isolated* puedan agruparse entre sí.

Las fases adicionales comparan clusters con clusters y páginas web *isolated* con clusters. La representación de los clusters consiste en una bolsa de palabras, al igual que las páginas web. Para obtener la bolsa de palabras asociada a un cluster, se calcula su centroide y posteriormente se filtran aquellos rasgos no representativos. Se asume que los rasgos representativos de un cluster son aquellos tales que: (i) aparecen en muchos documentos del cluster (tienen un alto valor de frecuencia de documento dentro del cluster (DF)) y (ii) aparecen en pocos clusters (tienen un alto valor de frecuencia inversa por cluster (ICF)). Para obtener los rasgos representativos de un cluster se sigue el siguiente proceso: se calcula el valor DF*ICF de todos los rasgos y se obtiene la mediana de todos esos valores. Finalmente, se filtran del cluster aquellos rasgos cuyos valores DF-ICF no superen la mediana, esto es, los que no son representativos. La elección de la mediana se justifica porque se trata de un estadístico que no es sensible a casos extremos. Dado un cluster C_k, denotamos como CT_k a su centroide obtenido de esta manera.

Algoritmo 1 $ExtendedUPND(\mathcal{W}, sim, \gamma)$

Entrada: Conjunto de páginas web $\mathcal{W}$= $\{W_1, W_2, ..., W_N\}$, medida de similitud sim y función de umbral γ.
Salida: Conjunto de clusters $\mathcal{C} = \{C_1, C_2, ..., C_l\}$

```
// Agrupar páginas web enlazadas
C = groupByLinks(W)
// Algoritmo UPND
para i = 1 to N hacer
    para j = i + 1 to N hacer
        si sim(W_i^3M, W_j^3M) > γ(W_i^3M, W_j^3M)
            C_i = C_i ∪ C_j
            C = C \{C_j}
        fin si
    fin para
fin para
// FASE extra 1: Agrupación de clusters
para k = 1 to |C| hacer
    para l = k + 1 to |C| hacer
        si sim(CT_k^1M, CT_l^1M) > γ(CT_k^1M, CT_l^1M)
            C_k = C_k ∪ C_l
            C = C \{C_l}
        fin si
    fin para
fin para
// FASE extra 2: Agrupación páginas isolated
I = {W ∈ W : ∃C ∈ C : C = {W}}
Caux = {C ∈ C : |C| > 1}
para W_I ∈ I hacer
    C_sim = bestCluster(W_I, Caux)
    si C_sim ≠ ∅
        C_sim = C_sim ∪ {W_I}
        C = C \{{W_I}}
    si no
        C_aux = C_aux ∪ {W_I}
    fin si
fin para
devolver C
```

4 Tratamiento de páginas web sociales

En los últimos años se ha elevado de forma significativa el número de usuarios que utilizan asiduamente redes sociales. Por ejemplo, Facebook[2] sobrepasa los 1000 millones de usuarios activos mensuales mientras que Twitter[3] sobrepasa los 300 millones. De ahí que los buscadores devuelvan habitualmente páginas web perte-

[2]https://newsroom.fb.com/company-info/
[3]https://about.twitter.com/company

necientes a redes sociales cuando la consulta se corresponde con un nombre de persona.

(Berendsen, 2015) fue el primero en estudiar el impacto de las redes sociales en la desambiguación de nombres de personas en la Web y concluye que la aparición de páginas sociales puede provocar que los sistemas del estado del arte realicen agrupaciones incorrectas. Su propuesta consiste en tratar a las páginas sociales de manera diferenciada bajo el supuesto de que cada página web social se refiere a un individuo diferente. Propone aplicar la política *one in one* sobre las webs sociales, lo cual significa mantener cada una de estas páginas web en un cluster unitario. No obstante, esta asunción tiene un par de limitaciones: (i) un individuo puede tener cuenta de usuario en varias redes sociales y (ii) un individuo puede tener varias cuentas de usuario en una misma red social.

En este trabajo proponemos dos heurísticas para tratar las páginas web sociales que corrigen las limitaciones de la política *one in one*. Ambas heurísticas requieren conocer si una página web pertenece a una red social y, en caso afirmativo, a cuál en concreto. Para ello, se ha tomado una lista de redes sociales de Wikipedia[4]. Tomando el dominio de una cierta página web a través de su URL, se puede comparar si se corresponde con algún dominio de la lista de redes sociales y, en caso afirmativo, conocer qué red social es.

4.1 P1: *One in one per social network*

Esta heurística asume que en un ranking las páginas web sociales correspondientes a una misma red social se refieren a individuos diferentes, porque suelen corresponderse con perfiles de usuario y además en un ranking no se repiten páginas. Esta heurística no permite la comparación de páginas web pertenecientes a la misma red social, pero sí permite la comparación de webs sociales de diferente red social, corrigiendo la primera limitación de la política *one in one*. Puesto que el algoritmo UPND agrupa páginas web por transitividad, varias páginas web de una misma red social pueden finalmente acabar en un mismo cluster, solucionando la segunda limitación. Si dos páginas web de la misma red social se agrupan por separado con una tercera página web de otro dominio, entonces las tres pertenecerán al mismo cluster. La aplicación de la heurística en el algoritmo consiste en evitar comparaciones de páginas web sociales de la misma red social.

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

4.2 P2: Eliminación de rasgos comunes

Esta heurística asume que muchas agrupaciones incorrectas con páginas sociales se deben a la compartición de vocabulario común de estas plataformas. En un escenario multilingüe, esto se cumple para las páginas web escritas en el mismo idioma. El tratamiento de las redes sociales de esta heurística es el siguiente: se forman grupos de páginas web sociales según la red social a la que pertenezcan e idioma en el que están escritas. En los grupos en los que existan al menos dos páginas web, se calculan qué rasgos aparecen en la mayoría de ellas y se eliminan de esas páginas web, asumiendo que se trata de vocabulario específico de dicha red social. Hemos tomado la política de eliminar aquellos rasgos que aparezcan en al menos el 75 % de las páginas web de un grupo. Esta heurística no impone restricciones a la hora de comparar páginas web sociales entre sí, de manera que se permite tanto la comparación de páginas web de distinta red social, como páginas web de la misma red social. Para poder aplicar esta heurística, se efectúa la eliminación de los rasgos de cada grupo antes de aplicar el algoritmo.

5 Experimentación

En esta sección presentamos las colecciones de páginas web utilizadas. Posteriormente, presentamos y analizamos los resultados obtenidos comparándonos con el otro sistema del estado del arte que hace un tratamiento diferenciado de las páginas web sociales.

5.1 Corpora de evaluación

Los corpora que hemos utilizado se caracterizan por contener un número significativo de páginas web sociales. Por esta razón, no hemos utilizado las colecciones de referencia en la tarea de las campañas de evaluación WePS. Estas colecciones contienen un pequeño número de webs sociales, y en particular, en WePS-2, los organizadores no consideraron para la evaluación de los resultados (Artiles, Gonzalo y Sekine, 2009b).

Las colecciones utilizadas son el corpus ECIR2012[5] y un nuevo corpus denominado MC4WePS[6]. La Tabla 1 muestra el número de nombres de personas y de documentos contenidos en las colecciones, el porcentaje de páginas web sociales y los porcentajes de nombres muy ambiguos y poco ambiguos. Se ha considerado

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

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

que un nombre es muy ambiguo si en las páginas web se hace referencia a más de 10 individuos diferentes. En caso contrario, hemos considerado que el nombre es poco ambiguo.

Dato	ECIR2012	MC4WePS
#Nombres	33	100
#Docs	3487	10432
%Social	34.73 %	8.36 %
%MuyAmbiguos	81.82 %	51.00 %
%PocoAmbiguos	18.18 %	49.00 %

Tabla 1: Datos de los corpora ECIR2012 y MC4WePS

Los nombres de persona contenidos en el corpus ECIR2012 son neerlandeses y las páginas web están escritas en dicho idioma. Los resultados de búsqueda incluidos en el corpus fueron devueltos por varios buscadores, concretamente Google, Yahoo! y Bing, de manera que no son rankings reales devueltos tras consultar cada nombre de persona. Esta colección fue construida para estudiar el impacto de las redes sociales en el problema, por lo que se añadieron artificialmente páginas de las redes sociales Facebook, Hyves[7], LinkedIn, Twitter y MySpace. Por esta razón, este corpus contiene un porcentaje muy elevado de webs sociales. La mayoría de los nombres de persona incluidos en esta colección son muy ambiguos.

En el caso de MC4WePS se incluyen nombres de persona de origen diverso, aunque mayoritariamente anglosajón e hispano. Los datos de esta colección fueron recopilados sobre los ranking de links devueltos por el buscador Google al realizar las consultas, de manera que se trata de resultados reales. Además, este corpus se caracteriza por contener páginas web escritas en diferentes idiomas, al contrario que las colecciones de referencia para este problema. Los anotadores identificaron páginas web escritas en 30 idiomas diferentes, prevaleciendo el inglés y el castellano (96.08 % entre ambos idiomas). Por otra parte, en este corpus está equilibrado el número de nombres muy ambiguos y poco ambiguos.

5.2 Resultados

En esta sección presentamos los resultados obtenidos por nuestro algoritmo para las dos colecciones descritas anteriormente. Las métricas utilizadas son las B^3-Cubed (Bagga y Baldwin, 1998): precisión (BEP), recall (BER) y medida-F ($F_{0,5}$). Estas métricas son adecuadas para evaluar sistemas de desambiguación de nombres de persona (Artiles, Gonzalo y Sekine, 2009b). Para la experimentación se han pesado los rasgos utilizando TF-IDF y se ha empleado la similitud coseno para comparar las páginas web.

La Tabla 2 presenta los resultados obtenidos sobre todas las páginas web y las páginas web sociales por el algoritmo propuesto por (Berendsen, 2015), al que hemos denominado BEREN, y por nuestra propuesta teniendo en cuenta varias configuraciones diferentes. En el caso del algoritmo BEREN, se presentan los resultados obtenidos por la mejor y la peor de sus configuraciones, denotados por BERENbest y BERENworse respectivamente. La primera penaliza la presencia de redes sociales a la hora de mezclar clusters no sociales con clusters sociales, mientras que la segunda no aplica dicha penalización. En el caso de nuestro algoritmo, presentamos los resultados obtenidos sin aplicar ningún tratamiento de redes sociales (ExtendedUPND) y aplicando nuestras dos heurísticas de tratamiento de páginas web sociales (ExtendedUPND+P1 y ExtendedUPND+P2). Por otra parte, se han incluido los resultados obtenidos por el algoritmo UPND. La tabla también muestra información relativa a la significancia estadística de los resultados calculada mediante el test de Wilcoxon (Wilcoxon, 1945) con un nivel de confianza del 95 %, tomando los pares de valores $F_{0,5}$ de cada nombre de persona. Para cada columna, los experimentos se marcan con (k) donde $k \in \mathbb{N}$, de modo que dos experimentos con la misma marca obtienen resultados similares, y un experimento marcado con (k) obtiene mejoras signiticativas sobre otro marcado con (l) si $k < l$.

Los resultados muestran que en el caso del corpus ECIR2012, las configuraciones de los algoritmos que tratan de manera especial a las webs sociales obtienen mejores resultados con respecto a las que no lo hacen. Esto corrobora lo concluido por (Berendsen, 2015) respecto al papel de este tipo de páginas web en este problema. En el caso de la colección MC4WePS, los resultados obtenidos son muy similares entre sí, lo cuál puede explicarse por el menor porcentaje de páginas web sociales presentes en dicho corpus con respecto a ECIR2012 (ver Tabla 1), por lo que el impacto de las redes sociales es mucho menor en esta colección.

[7] Se trata de una red social similar a Facebook, popular en los Países Bajos.

Corpus	ECIR2012						MC4WePS					
Webs	Todas las webs			Webs sociales			Todas las webs			Webs sociales		
Ejecución	BEP	BER	$F_{0,5}$	BEP	BER	$F_{0,5}$	BEP	BER	$F_{0,5}$	BEP	BER	$F_{0,5}$
BERENbest	0.90	0.80	0.83 (1)	1.00	0.79	0.87 (1)	0.91	0.43	0.50 (3)	0.99	0.68	0.77 (2)
BERENworse	0.74	0.82	0.76 (2)	0.55	0.85	0.62 (5)	0.91	0.43	0.50 (3)	0.99	0.68	0.77 (2)
ExtendedUPND	0.72	0.75	0.72 (3)	0.45	0.87	0.55 (6)	0.83	0.76	0.77 (1)	0.83	0.80	0.77 (2)
ExtendedUPND+P1	0.92	0.70	0.78 (2)	0.92	0.77	0.82 (2)	0.87	0.74	0.78 (1)	0.92	0.77	0.81 (1)
ExtendedUPND+P2	0.86	0.72	0.77 (2)	0.73	0.83	0.75 (3)	0.86	0.74	0.78 (1)	0.90	0.79	0.80 (1)
UPND	0.80	0.70	0.73 (3)	0.64	0.84	0.70 (4)	0.86	0.67	0.72 (2)	0.84	0.79	0.77 (2)

Tabla 2: Resultados de los algoritmos BEREN, ExtendedUPND y UPND para las colecciones ECIR2012 y MC4WePS sobre todas las páginas web y únicamente las páginas web de redes sociales.

En cuanto a la comparación de los algoritmos de clustering, vemos que en el corpus ECIR2012, la mejor configuración del algoritmo BEREN obtiene mejoras significativas con respecto al resto de ejecuciones, pero las dos configuraciones de este algoritmo obtienen resultados muy pobres en el corpus MC4WePS. Esto se explica porque el rendimiento de HAC depende fuertemente del valor del umbral usado como criterio de agrupamiento (Artiles, Gonzalo y Sekine, 2009b). UPND y ExtendedUPND evitan este problema gracias al uso de la función de umbral. Por otra parte, las diferencias en el grado de ambigüedad de ambos corpora también influyen en los resultados. Por un lado, ECIR2012 se compone de muchos nombres muy ambiguos, mientras que MC4WePS contiene un mayor equilibrio entre nombres poco ambiguos y muy ambiguos (ver Tabla 1), de manera que el umbral que recibe HAC en el sistema BEREN, con valor 0.225, funciona mejor con nombres muy ambiguos (con más clusters), pero mucho peor para nombres poco ambiguos (con menos clusters). En cambio, tanto UPND como ExtendedUPND son menos sensibles al grado de ambigüedad de los nombres de persona contenidos en ambas colecciones. Por otra parte, la penalización aplicada en el algoritmo de mezcla de webs sociales y no sociales de (Berendsen, 2015) no tiene efecto en el corpus MC4WePS, porque incluye un menor número de redes sociales y el umbral que usan en dicha fase, $\tau = 0{,}5$, es demasiado estricto, de modo que en ambas ejecuciones se agrupan el mismo número de webs sociales con webs no sociales. Finalmente, ExtendedUPND, además de corregir los defectos de representación de UPND, logra un mejor equilibrio entre precisión y recall, y en el caso de MC4WePS consigue mejoras significativas con respecto a UPND.

La tabla muestra que aplicando las heurísticas propuestas sobre todas las webs se obtienen resultados similares en los dos corpora, obteniendo mejoras significativas en ECIR2012 respecto UPND y ExtendedUPND. Además, ambas mejoran significativamente los agrupamientos de páginas sociales respecto no aplicarlas. Su efecto consiste en mejorar los resultados de precisión sin alterar drásticamente los valores de cobertura con respecto a no aplicarlas, lo que significa que evitan agrupamientos incorrectos. La heurística P2 obtiene resultados de precisión más bajos en el corpus ECIR2012. Esto se debe a que P2 tiende a agrupar páginas sociales de la misma red social en el mismo cluster pese a la eliminación de rasgos comunes. Este tipo de agrupamientos normalmente son incorrectos en ambos corpora puesto que se corresponden con perfiles de distintos individuos tal y como presupone la heurística P1. En cambio, en el corpus MC4WePS ambas heurísticas se comportan de manera similar. Esto se debe a que la mayoría de grupos de webs sociales por red social e idioma formados por P2 se componen de dos páginas web, de modo que se eliminan todos los rasgos comunes de las páginas de la misma red social, lo que imposibilita su agrupación como sucede con P1. Lo anterior indica que la suposición *one in one per social network* es adecuada, por lo que P1 es preferible a la hora de tratar las redes sociales.

6 Conclusiones y Trabajo Futuro

En este trabajo hemos presentado dos heurísticas para tratar las páginas web pertenecientes a redes sociales en el problema de desambiguación de nombres de personas. Además, hemos utilizado una extensión del algoritmo de clustering UPND que tiene en cuenta si las páginas están enlazadas, y que permite representar un mayor número de páginas web que el algoritmo original. Por un lado, nuestras heurísticas obtienen mejoras en el corpus ECIR2012 compuesto por más páginas

web sociales que MC4WePS. El efecto conseguido consiste en mejorar los resultados de precisión de los agrupamientos, sin que esto implique una caída drástica en los valores de cobertura, de manera que se evitan agrupaciones incorrectas. Por otro lado, ExtendedUPND ofrece mejores resultados que el algoritmo BEREN en el corpus MC4WePS. Este algoritmo es más independiente del grado de ambigüedad de los nombres respecto al método BEREN, cuyo rendimiento depende del umbral obtenido mediante entrenamiento.

Como trabajo futuro, proponemos realizar un tratamiento distintivo sobre los resultados que se corresponden con entradas de Wikipedia. Varios autores (Long y Shi, 2010; Xu et al., 2015) han destacado que la información proporcionada por esta enciclopedia online sirve de gran ayuda a la hora de diferenciar entre distintos individuos.

Bibliografía

Artiles, J. 2009. Web People Search. PhD Thesis, UNED University.

Artiles, J., J. Gonzalo, and S. Sekine. 2007. The SemEval-2007 WePS Evaluation: Establishing a Benchmark for the Web People Search Task. En *Proceedings of SemEval-2007*, pages 64-69. ACL.

Artiles, J., E. Amigó, and J. Gonzalo. 2009a. The Role of Named Entities in Web People Search. En *Proceedings of EMNLP 2009*.

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

Artiles, J., A. Borthwick, J. Gonzalo, S. Sekine, and E. Amigó. 2010. WePS-3 Evaluation Campaign: Overview of the Web People Search Clustering and Attribute Extraction Tasks. En *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*.

Bagga, A. and B. Baldwin. 1998. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. En *Proceedings of the COLING/ACL'98 - Volume 1*, pages 79-85.

Balog, K., J. He, K. Hofmann, V. Jijkoun, C. Monz, M. Tsagkias, W. Weerkamp, and M. de Rijke. 2009. The University of Amsterdam at WePS-2. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*.

Richard Berendsen 2015. Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation. PhD Thesis. Informatics Institute, University of Amsterdam.

Chen. Y., Yat Mei Lee, S., and Huang, C.R. 2012. A Robust Web Personal Name Information Extraction System. En *Expert Systems with Applications*, Vol. 32, Issue 3, pp. 2690-2699.

Delgado, A. D, R. Martínez, V. Fresno, and S. Montalvo. 2014. A Data Driven Approach for Person Name Disambiguation in Web Search Results. En *Proceedings of COLING 2014*, pages 301-310.

Delgado, A. D, R. Martínez, S. Montalvo, and V. Fresno. 2014. An Unsupervised Algorithm for Person Name Disambiguation in the Web. En *Procesamiento del Lenguaje Natural, 53*, pages 51-58.

Grüetze, T., Kasneci, G., Zuo, Z., and Naumann, F. 2014. Bootstrapping Wikipedia to answer ambiguous person name queries. En *Proceedings of the 30th International Conference on Data Engineering Workshops (ICDE)*, pages 56-61. Chicago, IL, USA.

Liu, Z., Q. Lu, and J. Xu. 2011. High Performance Clustering for Web Person Name Disambiguation using Topic Capturing. En *International Workshop on Entity-Oriented Search (EOS)*.

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

Nuray-Turan, R., Kalashnikov, D. V., and Mehrotra S. 2012. Exploiting Web querying for Web People Search. *ACM Transactions on Database Systems (TODS)*, Vol. 37, Issue 1.

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

Xu, J., Lu, Q., Li, M., and Li, W. 2015. Web Person Disambiguation Using Hierarchical Co-Reference Model. En *Proceedings of CICLing 2015, Part I, pages 279-291*.