TESIS DOCTORAL

2017

DESAMBIGUACIÓN DE NOMBRES DE PERSONA EN LA WEB EN UN CONTEXTO MULTILINGÜE

AGUSTÍN DANIEL DELGADO MUÑOZ

Máster en Informática por la Universidad Complutense de Madrid

PROGRAMA DE DOCTORADO EN SISTEMAS INTELIGENTES

Dra. RAQUEL MARTÍNEZ UNANUE

Dra. MARÍA DEL SOTO MONTALVO HERRANZ

TESIS DOCTORAL

2017

DESAMBIGUACIÓN DE NOMBRES DE PERSONA EN LA WEB EN UN CONTEXTO MULTILINGÜE

AGUSTÍN DANIEL DELGADO MUÑOZ

Máster en Informática por la Universidad Complutense de Madrid

PROGRAMA DE DOCTORADO EN SISTEMAS INTELIGENTES

Dra. RAQUEL MARTÍNEZ UNANUE

Dra. MARÍA DEL SOTO MONTALVO HERRANZ

A mi madre,
María Concepción.

Agradecimientos

Tenía pensado escribir estas líneas cuando empezase a ver algo de luz al final del túnel y, parece, aunque aún casi ni me lo crea, que ha llegado ese momento. Como no podía ser de otra manera, a continuación dedicaré unas palabras a todas las personas que han formado parte de mi vida durante el desarrollo de esta tesis doctoral: aquellas que forman parte de mi vida académica, y aquellas que forman parte de mi vida fuera de la universidad. Comenzaré por los primeros:

En primer lugar, tengo que dar las gracias a mis guías durante este viaje: mis directoras de tesis, Raquel Martínez y Soto Montalvo. Gracias por su infinita paciencia. Gracias por todo su tiempo y dedicación durante el desarrollo de este trabajo. Gracias por su supervisión concienzuda y meticulosa. Y gracias, porque sin ellas no hubiese sido posible que, por fin, empezase a otear un oasis en el desierto y presentase este trabajo. También quiero hacer una mención especial a Víctor Fresno. Durante estos años siempre ha dedicado parte de su tiempo en hablar conmigo, preguntarme qué tal estaba y cómo me iba, aconsejarme y darme alguna que otra idea o comentario interesante sobre la tesis. ¡Gracias, compañero!

Por supuesto, también quiero dar las gracias a toda la gente que he conocido en la UNED durante estos años. Tanto a los de mi Departamento, Lenguajes y Sistemas Informáticos, como a gente de fuera del Departamento. Tanto a los más veteranos, como a los más novatos. Tanto a los que están, como a los que se fueron. Gracias porque no podía imaginarme un mejor ambiente de trabajo cuando entré el primer día. La lista es muy extensa, pero estoy convencido de que todos ellos se reconocerán en estas palabras. En particular, quiero hacer una especial mención a los compañeros que han ido realizando su tesis doctoral al mismo tiempo que yo: Bernardo Cabaleiro, Ángel Castellanos, Andrés Duque y Javier Rodríguez. Durante estos años hemos compartido muchas charlas, risas y cervezas juntos, pero también hemos sido testigos de cómo hemos ido evolucionando como investigadores. Quiero que sepáis que ha sido un auténtico lujo haber compartido todos estos años con vosotros.

Durante el desarrollo de esta tesis he compaginado la actividad investigadora con la docencia. Me gustaría agradecer a Lourdes Araujo, Julio Gonzalo, Fernando López e Ignacio Mayorga toda su ayuda y consejos, que me han sido útiles para desempeñar

mucho mejor mi labor docente.

También quiero dar las gracias a Roberto Centeno por ser un excelente compañero de despacho, haberme prestado las plantillas de LaTeX con las que he elaborado este documento y por resolverme todas las dudas que me han ido surgiendo mientras trabajaba sobre ellas.

Quiero dedicar unas líneas de agradecimiento a Suresh Manandhar por su amabilidad a la hora de acogerme durante mi estancia en la Universidad de York y su disponibilidad absoluta a la hora de trabajar con él. Aparte de una estupenda experiencia personal, mi estancia en York me ha servido para ampliar mis conocimientos en el área fuera del ámbito de esta tesis doctoral. Hago extensivos estos agradecimientos a Alexandros Komninos y Nils Mönning por haber compartido conmigo su trabajo y haber hecho que mi estancia en York fuese mucho más amena y agradable.

Agradezco a Richard Berendsen que no haya dudado en responderme a todas las dudas que le he ido planteando en varias ocasiones sobre su trabajo en desambiguación de nombres de personas, y que me han servido de gran ayuda durante el desarrollo de esta tesis.

A continuación, quiero dedicar unas líneas a algunas personas que están fuera de mi vida universitaria:

Quiero agradecer a mis amigos Darío y Juan Carlos todo el ánimo que me han ido transmitiendo durante estos años, aguantarme alguna que otra chapa sobre este trabajo (unas veces a traición y otras sobre aviso) y por haberme salvado del régimen de reclusión en el que me he visto sometido en algunas ocasiones (más de las deseables). Parece que vais a tener razón y voy a terminar esta etapa a la que no veía fin. ¿Qué más deciros? Que habrá que pensar en algo para celebrarlo y que *¡Forza Atleti!*

Finalmente, quiero dar las gracias a la persona más importante de mi vida, mi madre, María Concepción. Ella ha sido la testigo de excepción de mi andadura en esta montaña rusa llamada tesis doctoral, con sus subidas, sus bajadas y algún que otro *looping*. Sé a ciencia cierta que cuando las cosas me han ido bien, ella se ha alegrado el doble que yo. Pero también me ha visto en mis peores momentos de desesperación y en algunas (más de las que hubiese querido) de mis noches maratonianas sin pegar ojo. Y sé que, en esos momentos, muy a mi pesar, también lo ha pasado mal. Pero, precisamente en esos momentos donde peor lo he pasado, ella ha sido mi pilar fundamental y me ha transmitido todas las fuerzas suficientes para seguir adelante. Por este motivo, esta tesis no podía estar dedicada a otra persona que no fuese ella. Gracias por todo.

Madrid, 25 de mayo del 2017
Agustín Daniel Delgado Muñoz

Resumen

Esta tesis doctoral trata la desambiguación de nombres de personas en la Web. Este problema puede describirse de la siguiente manera: dado el ranking de resultados devuelto por un motor de búsqueda tras consultar un nombre de persona, el objetivo consiste en agrupar los resultados de búsqueda de manera que cada grupo esté formado por las páginas web que hablan de un mismo individuo. Los motores de búsqueda más populares ofrecen pocas herramientas de desambiguación de este tipo de consultas, aunque sus estadísticas de uso reflejan que son muy frecuentes. Por este motivo, en los últimos años han surgido varias *start-ups* que ofrecen un servicio especializado de búsqueda de personas en Internet. Además, la comunidad científica ha mostrado interés en este problema por varias razones. Por un lado, los nombres de persona son un tipo de entidades nombradas especialmente ambiguo y, por este motivo, su desambiguación ha sido estudiada en diferentes contextos. Por otro lado, el escenario de búsqueda en la Web presenta varios retos: (i) las páginas web no tratan una temática determinada debido a su naturaleza heterogénea; (ii) la Web alberga cada vez más contenido en distintos idiomas debido a su naturaleza multilingüe; y (iii) la búsqueda en la Web requiere métodos poco costosos debido a que los usuarios de los motores de búsquedas esperan resolver sus consultas en muy poco tiempo. Por tanto, nos encontramos ante un problema real que ha suscitado el interés de la comunidad científica.

La desambiguación de nombres de personas en la Web ha sido tratada en el estado del arte como un problema de *clustering* compuesto por dos fases principales. El objetivo de la primera fase consiste en representar los resultados de búsqueda mediante rasgos adecuados que sean de utilidad a la hora de identificar y distinguir a distintos individuos con el mismo nombre. Por otro lado, la segunda fase consiste en aplicar un algoritmo de *clustering* para agrupar las páginas web de acuerdo al individuo que mencionan. En particular, los mejores sistemas del estado del arte emplean una representación de los resultados de búsqueda consistente en una rica selección de rasgos de distinto tipo y agrupan las páginas web mediante un algoritmo de agrupamiento jerárquico aglomerativo tras haber aprendido previamente el valor de un cierto umbral de similitud mediante datos de entrenamiento.

La presente tesis doctoral se centra en el estudio de tres aspectos del problema que no han sido estudiados en profundidad en el estado del arte:

- La agrupación de los resultados de búsqueda mediante un umbral de similitud aprendido a partir de datos de entrenamiento tiene el inconveniente de que puede generar resultados sesgados según sean las características de los datos de entrenamiento empleados. En particular, el valor del umbral de entrenamiento aprendido puede no ser de utilidad para otros datos con diferentes características, de modo que puede implicar que los sistemas basados en esta metodología sean poco robustos. Además, la obtención de suficientes datos de entrenamiento representativos para garantizar la consistencia de los resultados con diferentes nombres de persona requiere de un gran esfuerzo por parte de anotadores expertos.
- Los sistemas del estado del arte no tienen en cuenta el papel de las redes sociales en el problema. El éxito de estas plataformas ha tenido un impacto incuestionable en Internet. En particular, los motores de búsqueda devuelven frecuentemente este tipo de páginas web cuando se consulta un nombre de persona. Recientemente, se ha señalado que la presencia de estas páginas web tienen un impacto negativo en el problema y deben tratarse de manera diferente al resto de resultados de búsqueda. No obstante, las propuestas de tratamiento de las redes sociales no tienen en cuenta algunas situaciones habituales en la actualidad como, por ejemplo, que un individuo tenga perfiles en varias de estas plataformas.
- La desambiguación de nombres de personas en la Web se ha abordado bajo la asunción de un escenario monolingüe, donde todos los resultados de búsqueda están escritos en el mismo idioma. No obstante, Internet alberga cada vez más contenido escrito en diferentes idiomas, de modo que es necesario tener en cuenta el multilingüismo en un escenario de búsqueda real. Por ejemplo, es frecuente encontrar personas que publican información de su ámbito profesional en inglés, mientras que publican información de ámbito personal en su lengua nativa.

El objetivo de esta tesis consiste en definir y analizar nuevas estrategias de desambiguación de nombres de personas en la Web que sean capaces de resolver estos problemas. Además, se ha tenido en cuenta que las propuestas presentadas en esta tesis no sean muy costosas computacionalmente con el objetivo de que puedan ser llevadas a la práctica. En particular, los enfoques presentados en esta tesis para resolver los problemas anteriores son los siguientes:

- En primer lugar, se realiza un estudio sobre diferentes tipos de rasgos para comprobar cuáles de ellos son más adecuados para representar los resultados de búsqueda. Además, se presentan varios algoritmos de *clustering* que no requieren

ninguna información a priori. Los algoritmos se basan en el concepto de *umbral adaptativo*, introducido en esta tesis, que consiste en una función matemática que devuelve un umbral de similitud para agrupar resultados de búsqueda. El valor de los umbrales adaptativos depende exclusivamente de las características de los resultados de búsqueda que se comparan y los rasgos que tienen en común.

- Se proponen varias heurísticas para tratar de manera especial las redes sociales. Las heurísticas propuestas evitan las limitaciones de las políticas presentadas en el estado del arte, de modo que asumen que un individuo tenga perfiles en varias redes sociales o, incluso, en la misma red social. Además, se ha extendido el estudio a otro tipo de páginas relacionadas con las redes sociales: los buscadores de personas. Este tipo de páginas web se caracterizan por presentar un listado de enlaces a perfiles de redes sociales de diferentes individuos. La heurística que presenta un mejor comportamiento se caracteriza por ser independiente de la representación de los resultados de búsqueda y puede aplicarse de manera sencilla sobre cualquier algoritmo de *clustering* sin que se vea afectado la eficiencia computacional.

- Se presentan dos aproximaciones para tratar el multilingüismo en el problema. La primera aproximación se basa en el empleo de una herramienta de traducción automática. En cambio, la segunda propuesta no emplea recursos de traducción para tratar el multilingüismo. Este método identifica los rasgos que se escriben igual en distintos idiomas y les otorga un papel importante a la hora de comparar los resultados de búsqueda que estén escritos en idiomas diferentes. Esta aproximación obtiene mejoras significativas con respecto a los experimentos que emplean traducción automática, de modo que tiene la ventaja de que evita que el tiempo de proceso del sistema de desambiguación se incremente por una fase de preprocesamiento adicional dedicada a la traducción de las páginas web.

La evaluación de las propuestas presentadas en esta tesis se ha realizado con varias colecciones que contienen nombres de personas con distinto grado de ambigüedad, resultados de búsqueda correspondientes a redes sociales y páginas web escritas en diferentes idiomas.

Abstract

This thesis addresses person name disambiguation on the Web. This problem can be described as follows: given a web pages ranking retrieved by a search engine when looking for a person name, the goal is to group properly the search results, so each group contains all the search results which refer to the same individual. The most popular search engines provide little disambiguation tools for this kind of queries, although their usage statistics show that they are very frequent. Because of this, several start-ups offer specialized services in people search on the Web. In addition, the scientific community has shown interest in this problem for several reasons. On the one hand, person name disambiguation has been studied in several contexts due to person names are an especially ambiguous kind of named entities. On the other hand, the search scenario on the Web presents several challenges: (i) web pages do not treat a specific topic because of its heterogeneous nature; (ii) the Web increasingly hosts web pages written in different languages because of its multilingual nature; and (iii) the Web search scenario requires efficient methods due to users expect quick responses. Therefore, this is a real problem that has aroused the interest of the scientific community due to its characteristics.

Person name disambiguation has been dealt as a clustering problem composed by two main 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 goal of the second phase is to apply a clustering algorithm to group the web pages according to the individual they refer to. In particular, the best systems of the state-of-the-art represent the search results by means of a rich selection of features of different kind, while they employ the hierarchical agglomerative clustering algorithm to group the web pages after having previously learned the value of a similarity threshold by means of training data.

The present thesis is focused in the study of three aspects of the problem that have not been studied deeply in the state-of-the-art:

- Grouping the search results using a similarity threshold learned by means of training data has the disadvantage that it can generate biased results according to the characteristics of the training data used. In particular, the value of the similarity

threshold learned may not be useful for other data with different characteristics, so that it may imply that the systems based on this methodology are not robust. In addition, this methodology requires enough and representative training data to guaranty the results will be consistent for different data collections, which requires a huge human effort.

- The systems of the state-of-the-art do not take into account the role of social networking sites in the problem. The success of these platforms has had an unquestionable impact on the Internet. In particular, search engines frequently return this kind of search results when the query is a person name. Recently, it has been pointed out that the presence of these web pages has a negative impact on the problem, so they should be treated differently. However, the proposals of the state-of-the-art do not take into account current situations such as, for instance, that an individual could have profiles on several of these platforms.

- Person name disambiguation on the Web has been addressed assuming a monolingual scenario. However, the Internet increasingly hosts content written in different languages, so the disambiguation systems have to take into account multilingualism in a real search scenario. For instance, web pages that contain professional information about a non-English native speaker might be written in English, while other web pages containing personal information about the same individual might be written in their native language.

The goal of this thesis is to define and analyze new person name disambiguation strategies able to solve these problems. In addition, we have considered that the proposals presented in this thesis should be inexpensive in order to put them in practice in a real scenario. In particular, the approaches presented in this thesis to solve the above problems are the following:

- First, a study is done in order to identify what kind of features are suitable to represent the search results. In addition, we present several clustering algorithms which do not need any prior information. These algorithms are based on the concept of *adaptive threshold*, introduced in this thesis, which are mathematical functions which return a similarity threshold to compare search results. The adaptive thresholds exclusively depend on the characteristics of the compared search results and how many features they have in common.

- We propose several heuristics to treat in a special way the web pages from social networking platforms. The proposed heuristics avoid the limitations of the proposal of the state-of-the-art, so they assume that an individual can have several

profiles in different social networks or even in the same social platform. In addition, we have extended this study to other kind of web pages related to social networks: people search engines. This kind of web pages usually present a list of social profiles of different individuals. The best proposed heuristic is independent with respect to the search results representation so it can be applied in any clustering algorithm, and it does not lead to increase the computational cost.

- We present two approaches to treat multilingualism in the problem. The first approach is based on the use of a machine translation tool. On the other hand, the second proposal does not need any translation resource to treat multilingualism. This method is based on comparing search results written in different languages giving an special role to those features which are written the same way in both languages. This approach improves the results obtained by the experiments that use machine translation resources and it avoids an increment of the processing time due to an additional phase dedicated to translate the search results.

The evaluation of the approaches presented in this thesis have been conducted with several collections that contain person names with different ambiguity degrees, results from social networking platforms and web pages written in different languages.

Índice general

Índice de figuras

Índice de tablas

Listado de Acrónimos

- **ACE** *Automatic Content Extraction*
- **ACL** *Association for Computational Linguistics*
- **AE** Extracción de Atributos (*Attribute Extraction*)
- **AP** *Affinity Propagation*
- **ATC** *Adaptive Threshold Clustering*
- **ATMC** *Adaptive Threshold for Multilingual Clustering*
- **BoW** Bolsa de Palabras (*Bag of Words*)
- **CDCR** *Cross-Document Co-reference Resolution*
- **DC** Clustering de Documentos (*Document Clustering*)
- **DF** Frecuencia de Documento (*Document Frequency*)
- **DF-ICF** Frecuencia de Documento-Frecuencia Inversa de Cluster (*Document Frequency-Inverse Cluster Frequency*)
- **ECDL** *European Conference on Digital Libraries*
- **EDL** *Entity Discovery and Linking*
- **EL** *Entity Linking*
- **HAC** Clustering Jerárquico Aglomerativo (*Hierarchical Agglomerative Clustering*)
- **HTML** *HyperText Markup Language*
- **ICF** Frecuencia Inversa de Cluster (*Inverse Cluster Frequency*)
- **IDF** Frecuencia Inversa de Documento (*Inverse Document Frequency*)
- **IE** Extracción de Información (*Information Extraction*)
- **IR** Recuperación de Información (*Information Retrieval*)

- **ITU** Unión Internacional de Telecomunicaciones (*International Telecommunication Union*)
- **KLD** Divergencia de Kullback-Leibler (*Kullback-Leibler Divergence*)
- **LDA** *Latent Dirichlet Allocation*
- **LSA** Análisis de Semántica Latente (*Latent Semantic Analysis*)
- **NE** Entidad Nombrada (*Named Entity*)
- **NED** Desambiguación de Entidades Nombradas (*Named Entity Disambiguation*)
- **NER** Reconocimiento de Entidades Nombradas (*Named Entity Recognition*)
- **MI** Información Mutua (*Mutual Information*)
- **MUC** *Message Understanding Conference*
- **NePS** *News People Search*
- **NIST** *National Institute of Standards and Technology*
- **NLP** Procesamiento del Lenguaje Natural (*Natural Language Processing*)
- **OIOS** *ONE IN ONE Social*
- **pLSA** Análisis de Semántica Latente Probabilístico (*Probabilistic Latent Semantic Analysis*)
- **QT** *Quality Threshold*
- **SemEval** *Semantic Evaluation*
- **SPC** *Single Pass Clustering*
- **SVD** Descomposición en Valores Singulares (*Singular Value Decomposition*)
- **TAC** *Text Analysis Conference*
- **TF** *Term Frequency*
- **TF-IDF** *Term Frequency - Inverse Document Frequency*
- **TM** Minería de Textos (*Text Mining*)
- **UNESCO** Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (*United Nations Educational, Scientific and Cultural Organization*)
- **UPND** *Unsupervised Person Name Disambiguator*

- **VSM** Modelo de Espacio Vectorial (*Vector Space Model*)
- **WePS** *Web People Search*
- **WSD** Desambiguación del Sentido de las Palabras (*Word Sense Disambiguation*)

1 Introducción

"Vimos que mil resultados no eran necesariamente tan útiles como diez buenos resultados."

— Sergey Brin —

La presente tesis se centra en la desambiguación de nombres de persona en un escenario de búsqueda en la Web. Este capítulo describe brevemente este problema explicando su utilidad para los usuarios de motores de búsqueda y por qué se considera interesante por parte de la comunidad científica. Se describen cuáles son los frentes abiertos a día de hoy en el problema, se presentan las hipótesis de investigación planteadas en esta tesis y se detalla la metodología científica llevada a cabo. Por último, se presenta la estructura de esta memoria junto con un breve resumen del contenido tratado en los posteriores capítulos de la tesis.

1.1. Motivación

Internet nos permite acceder a una cantidad ingente de información de todo tipo. Tras su popularización durante la década de los años 90 del siglo XX, no ha parado de aumentar tanto su número de usuarios como la cantidad de contenido que alberga. Según estimaciones de la ITU[1] (*International Telecommunication Union*), en el año 2016 los usuarios de Internet en todo el mundo han sido 3488 millones, frente a los 1151 millones de usuarios que estimaron en 2006. Por otro lado, dado el constante crecimiento de la Web, es complicado estimar su tamaño. Habitualmente, este cálculo se ha realizado a partir de la denominada Web superficial (*surface web*), esto es, el conjunto de páginas web indexadas por los motores de búsqueda. De acuerdo con Google[2], su primer prototipo presentado por Brin y Page [1998] era capaz de indexar alrededor de 26 millones de páginas web, frente a aproximadamente un billón (10^{12}) en 2008. Estimaciones más recientes [Bosch et al., 2016] calculan que Google indexaba en el año 2015 unos 4 billones de páginas web. No obstante, estos cálculos dejan fuera a la denominada Web profunda (*deep web*), compuesta por los sitios web no indexados por los principales motores de

[1] `http://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx`

[2] `https://googleblog.blogspot.com.es/2008/07/we-knew-web-was-big.html`

búsqueda, de la que se ha estimado que alberga un contenido 500 veces mayor que el de la Web superficial [He et al., 2007].

Los motores de búsqueda se han convertido en una herramienta fundamental a la hora de acceder rápidamente al contenido en la Web. Estos sistemas reciben por parte de un usuario de Internet una consulta (o *query*) consistente en una serie de términos y devuelven un ranking de resultados (páginas web, ficheros, imágenes, etc.) relacionados con dicha consulta. El uso de estas herramientas no ha parado de crecer a lo largo de los últimos años. Según el *Statistic Brain Research Institute*[3], Google tuvo un tráfico de 2.8 billones de consultas en el año 2015 frente a los 3.6 millones de consultas que recibió en el año 1998. Algunos buscadores ofrecen a los usuarios facilidades a la hora de relacionar entre sí los resultados de búsqueda de acuerdo con un determinado criterio. Por ejemplo, *Yippy*[4] (antes denominado *Clusty*) agrupa los resultados de acuerdo a su temática. Por otro lado, algunos buscadores comerciales sugieren distintos significados cuando la consulta consiste en un término ambiguo, aunque no clasifican los resultados de búsqueda en consecuencia. Para ilustrar esta última situación, la Figura 1.1 muestra las sugerencias proporcionadas por Google al consultar el término *jaguar*: la marca de automóviles y el animal. Esta situación convierte la Web en un escenario real idóneo a la hora de aplicar técnicas de clasificación automática y *clustering* con el objetivo de facilitar la búsqueda de información a los usuarios. Además, teniendo en cuenta las desorbitantes cifras presentadas anteriormente, se aprecia una necesidad cada vez mayor de mejorar los servicios de búsqueda en la Web para un creciente número de usuarios y contenidos. Por añadidura, la inmediatez solicitada por los usuarios a la hora de buscar información en la Web hace imprescindible que el uso de este tipo de técnicas impliquen poco coste computacional.

1.1.1. Descripción del problema

La presente tesis se centra en el problema de la desambiguación de nombres de persona en un escenario de búsqueda en la Web. Este problema presenta una situación muy habitual para cualquier usuario de motores de búsqueda de Internet interesado en encontrar información sobre un determinado individuo. El objetivo del problema es doble: por un lado, estimar el número de individuos mencionados en los resultados de búsqueda y, por otro lado, agrupar adecuadamente las páginas web de acuerdo con el individuo al que hacen referencia. De esta manera, el usuario puede seleccionar únicamente los resultados de búsqueda que mencionan al individuo que le interesa con mayor rápidez y sin necesidad de refinar su consulta. Los primeros trabajos en plantear

[3] http://www.statisticbrain.com/google-searches/

[4] https://yippy.com/

Figura 1.1: Sugerencias de Google al consultar el término *jaguar*.

la desambiguación de nombres de personas en el dominio Web (ej. [Mann y Yarowsky, 2003; Al-Kamha y Embley, 2004; Bekkerman y McCallum, 2005; Wan et al., 2005; Bollegala et al., 2006]) surgieron bajo la influencia de trabajos anteriores (ej. [Bagga y Baldwin, 1998; Winchester y Lee, 2002]) centrados en la resolución de correferencias de este tipo de entidades en el dominio de las noticias. Posteriormente, las campañas de evaluación WePS (*Web People Search*) [Artiles et al., 2007, 2009b, 2010] estandarizaron el problema bajo el punto de vista del *clustering* de los resultados de búsqueda y proporcionaron varias colecciones de evaluación. Desde entonces, la mayoría de los trabajos publicados sobre el problema lo han enfocado de esta manera y han utilizado las colecciones de datos proporcionadas por estas campañas de evaluación.

Los nombres de personas son un tipo de entidad nombrada (*Named Entity*, NE) especialmente ambiguo, por lo que es frecuente que una misma entidad haga referencia a varios individuos diferentes. Según los datos disponibles por el Censo de EEUU, solo 90.000 nombres diferentes eran compartidos por unos 100 millones de personas [Artiles et al., 2010]. Para ilustrar esta situación mediante un ejemplo, la Figura 1.2 muestra algunos resultados de búsqueda devueltos por Google al consultar el nombre de persona *Neil Clark*. Los enlaces mencionan a varios individuos llamados así, por ejemplo, un periodista (primer y segundo enlace), un músico (tercer y quinto enlace) y un jugador de rugby (cuarto enlace). Ante esta situación, el usuario se ve en la obligación de ir seleccionando aquellos resultados que contienen información del individuo que busca, lo cual implica una inversión de tiempo adicional. Alternativamente, el usuario puede refinar su consulta añadiendo más términos de búsqueda, lo cual reduce la cantidad de páginas web recuperadas por el motor de búsqueda y se puede perder información disponible en la Red del individuo buscado no relacionada con los nuevos términos introducidos. La Figura 1.3 muestra la salida esperada por un sistema de desambiguación de nombres de personas, de forma que los resultados de búsqueda se agrupan según el individuo al

que mencionan.

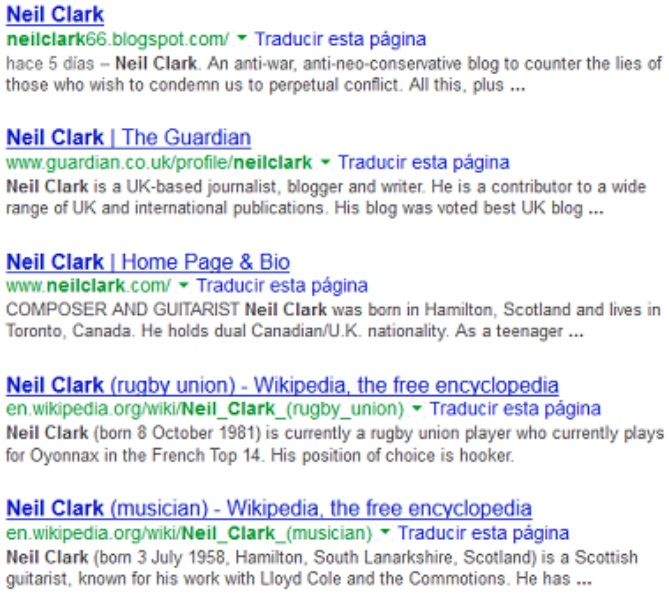

Figura 1.2: Búsqueda en Google del nombre de persona *Neil Clark*.

Figura 1.3: Desambiguación de resultados de búsqueda del nombre de persona *Neil Clark*.

La búsqueda de información sobre personas es una práctica común entre los usuarios de los motores de búsqueda. Según Artiles et al. [2010], un 4 % de consultas web consisten exclusivamente en un nombre de persona, mientras que entre un 11 % y 17 % contienen un nombre de persona. En el año 2016, tres de los diez términos de búsque-

da más populares consultados a Google en todo el mundo consistieron en nombres de personas, según los datos proporcionados por la herramienta Google Trends[5] que determina qué términos de búsqueda son populares entre los usuarios de este buscador. Pese a que los nombres de persona son un tipo de consulta habitual, los principales motores de búsqueda de Internet (Google, Yahoo! y Bing) solo incluyen herramientas de desambiguación mediante la información proporcionada por sus grafos de conocimiento (*knowdlege graphs*) cuando el nombre de persona es compartido por varias celebridades o personajes históricos. Esta situación no ha sido pasada por alto y por esta razón en los últimos años algunas *start-ups* han lanzado a la Red buscadores verticales de pago especializados en la búsqueda de personas (ej. Spokeo[6], Intelius[7], Pipl[8], ...). No obstante, estos servicios, basados en métodos de búsqueda de datos en la Web profunda, tampoco ofrecen herramientas de desambiguación y sus resultados se centran en perfiles de redes sociales.

El dinamismo en la Red puede provocar que los resultados devueltos por los motores de búsqueda varíen a lo largo del tiempo. Un ejemplo reciente lo encontramos en la popularización de las redes sociales. Estas plataformas son portales web que permiten que usuarios que comparten alguna relación o interés se comuniquen entre sí a través de Internet. En los últimos años ha crecido de manera significativa el número de usuarios de estos servicios web. Por ejemplo, Twitter[9] cuenta con 313 millones de usuarios activos mensuales. Por su parte, Facebook[10] en diciembre del año 2016 contaba con 1860 millones de usuarios activos mensuales. El uso cotidiano de las redes sociales por parte de un gran número de usuarios ha tenido como consecuencia que sea común obtener como resultados de búsqueda este tipo de páginas web cuando se realiza una consulta a un buscador. Para ilustrar esta situación con un ejemplo, la Figura 1.4 muestra los diez primeros resultados de búsqueda devueltos por Google al consultar el nombre de persona *Javier Martínez*. Cinco de estos resultados de búsqueda se corresponden con las redes sociales *Facebook*, *Twitter* y *LinkedIn*[11]. No obstante, los principales sistemas de desambiguación de nombres de personas en la Web no han tenido en cuenta este fenómeno, dado que han sido evaluados con colecciones de datos recopiladas con anterioridad al éxito de este tipo de servicios web.

El escenario de búsqueda en la Web cuenta con dos características que suponen un reto en tareas de desambiguación en los ámbitos de la Minería de Textos (*Text Mining*, TM), el Procesamiento del Lenguaje Natural (*Natural Language Processing*, NLP) y la

[5] `https://trends.google.es/trends/topcharts#geo&date=2016`
[6] `http://www.spokeo.com/`
[7] `https://www.intelius.com/`
[8] `https://pipl.com/`
[9] `https://about.twitter.com/company` (Fecha de acceso: 20/02/2017)
[10] `https://newsroom.fb.com/company-info/` (Fecha de acceso: 20/02/2017)
[11] `https://www.linkedin.com`

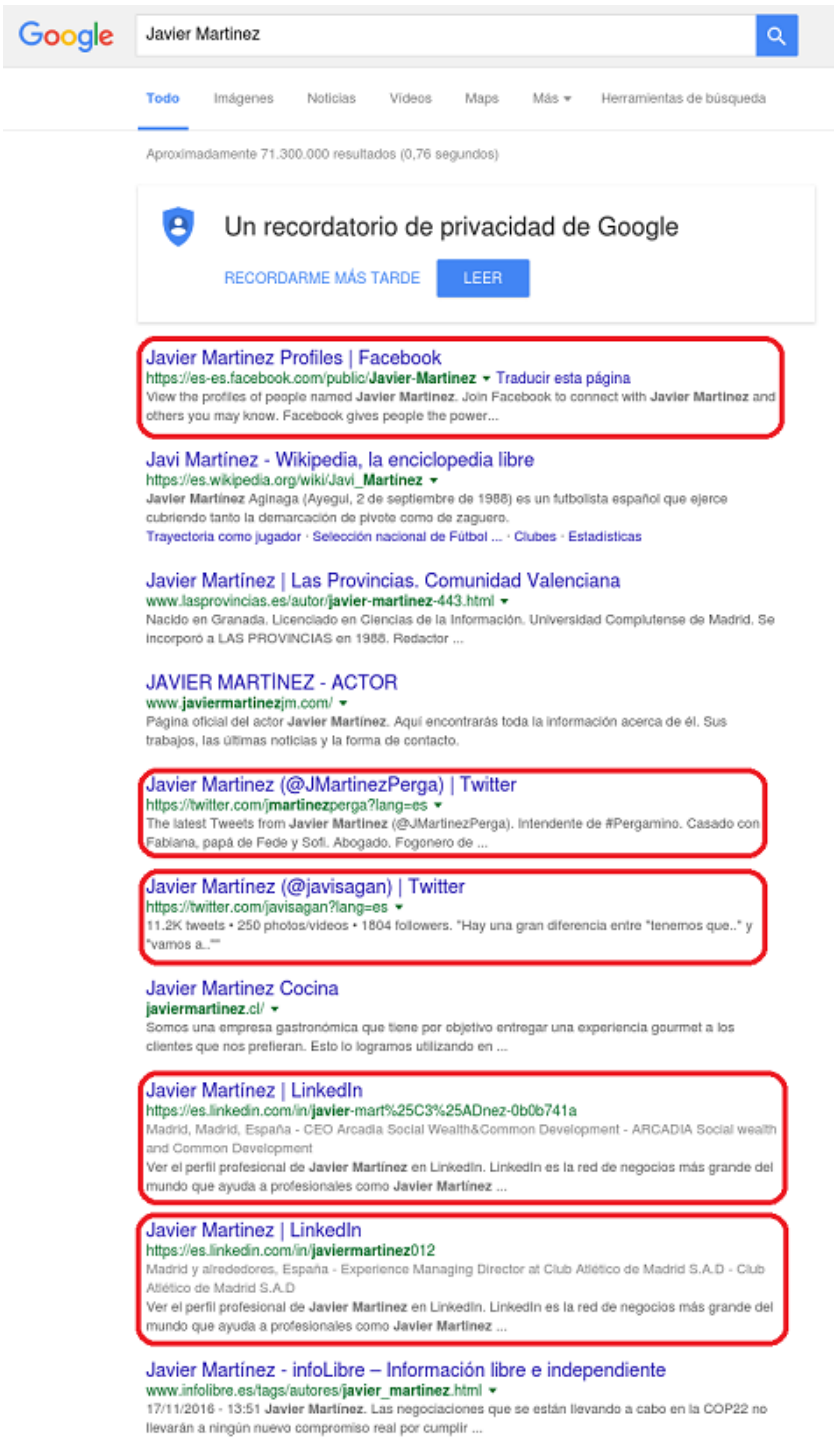

Figura 1.4: Resultados de búsqueda del nombre de persona *Javier Martínez*, con los enlaces correspondientes a redes sociales señalados mediante marcos de color rojo.

Recuperación de Información (*Information Retrieval*, IR). Por un lado, la naturaleza heterogénea de las páginas web conduce a que podamos encontrarnos con documentos de distintas temáticas que pueden no tener relación entre sí. La presencia de temáticas diversas en los documentos supone una dificultad añadida en los procesos de desambi-

guación de entidades [Hoffart et al., 2011]. Por otra parte, el carácter global de la Web permite que los usuarios puedan acceder a contenidos escritos en diferentes idiomas, por lo que las técnicas utilizadas en su tratamiento deben manejar adecuadamente el multilingüismo. Actualmente, el multilingüismo sigue siendo considerado un reto en problemas relacionados con el reconocimiento y la desambiguación de NEs [Carmel et al., 2014].

Un ejemplo de la heterogeneidad de la Web se puede encontrar analizando el contenido de las páginas web de un mismo individuo. Las páginas dedicadas al ámbito laboral pueden no tener ninguna información común con páginas web de ámbito personal donde se exponen opiniones, experiencias o aficiones de ese individuo. Relacionado con esto, surge la dificultad adicional de que la manera en la que los usuarios escriben en Internet difiere según el tipo de página web: en páginas web de ámbito laboral se tiende a escribir de una manera formal, mientras que en foros o redes sociales se escribe de manera informal usando un mayor número de abreviaciones (ej. *k* en lugar de *que*), jerga de Internet (ej. *LOL*) o *emoticonos* (ej. *:)*).

En cuanto al caracter multilingüe de la Web, según datos de la UNESCO (*United Nations Educational, Scientific and Cultural Organization*), pese a que el inglés continúa siendo el idioma más utilizado en la Red, cada vez existe un mayor contenido en otros idiomas debido a la popularización más reciente de Internet en países de habla no inglesa [Pimienta et al., 2009]. Esto implica una necesidad cada vez mayor de tratar de manera efectiva el multilingüismo en la Web [Montalvo et al., 2015a]. Por ejemplo, las páginas web con información laboral de personas no angloparlantes pueden estar escritas en inglés, mientras que sus páginas personales suelen estar escritas en sus idiomas nativos. En el caso de personajes conocidos, los resultados de búsqueda pueden incluir noticias que les mencionen en medios de comunicación en diferentes idiomas. No obstante, los principales trabajos relacionados con la desambiguación de nombres de persona en la Web no han tenido en cuenta el multilingüismo.

1.1.2. Problemas relacionados

En los últimos años, la comunidad científica ha planteado diferentes escenarios en los que resulta crucial la desambiguación de nombres de persona. A continuación, enumeramos los más relevantes:

- ***Cross-Document Coreference Resolution* (CDCR):** este problema consiste en obtener la *cadena de correferencias* de las entidades que aparecen en los documentos, esto es, el conjunto de expresiones en las que se menciona a cada una de ellas. La dificultad del problema reside en las múltiples expresiones que pueden utilizarse para

referirse a una misma entidad. Por ejemplo, *Donald Trump*, *Mr. Trump*, *Presidente de EEUU* o *ganador de las últimas elecciones* pueden ser correferencias que aluden a un mismo individuo. Varios autores se han centrado en resolver este problema exclusivamente para nombres de persona [Bagga y Baldwin, 1998; Winchester y Lee, 2002; Fleischman y Hovy, 2004; Gooi y Allan, 2004] tomando en consideración sus posibles variantes a la hora de escribirlos [Baron y Freedman, 2008]. Estos trabajos han estudiado el problema dentro del dominio de las noticias, debido a que muchas de ellas se centran en los individuos que las protagonizan [Coll Ardanuy et al., 2016] y son una importante fuente de información para una gran cantidad de usuarios [Montalvo, 2012]. Un ejemplo es la tarea NePS (*News People Search*) propuesta por las campañas de evaluación EVALITA [Bentivogli et al., 2013] donde se estudia el problema a partir de noticias de medios de comunicación italianos.

- ***Entity Linking* (EL):** las conferencias TAC (*Text Analysis Conference*) organizadas por la agencia gubernamental americana NIST (*National Institute of Standards and Technology*) llevan proponiendo tareas relacionadas con la desambiguación de NEs desde el año 2009[12], incluyendo nombres de personas, localidades y organizaciones. En concreto, la tarea EL, actualmente denominada *Entity Discovery and Linking*, consiste en enlazar entidades mencionadas en una colección de documentos a una base de conomiento de referencia, o descubrir nuevas entidades no contenidas en dicha base de conocimiento, identificando los documentos que mencionan a cada una de ellas. Habitualmente, la base de conocimiento utilizada es Wikipedia[13] y el problema se denomina *wikificación*. Además de las conferencias TAC, varios autores han propuestos escenarios de EL centrados exclusivamente en nombres de persona. La segunda conferencia CIPS-SIGHAN[14], enfocada en el procesamiento del lenguaje chino, propuso un escenario de EL únicamente sobre nombres de personas debido a la alta ambigüedad de estas entidades en ese idioma [He et al., 2012]. Más recientemente, Grütze et al. [2014] presentaron una colección de wikificación compuesta solamente por nombres de persona.

- **Desambiguación de autores (*author name disambiguation*):** el alto crecimiento que han experimentado las librerías digitales ha motivado el problema de la desambiguación de autores, consistente en la correcta asignación de autores a sus publicaciones. Además de la ambigüedad de los nombres de los autores, la dificultad de esta tarea reside en varios factores como, por ejemplo, errores y falta de información en las referencias o el uso de diferentes abreviaturas de las publicaciones [Ferreira et al., 2012]. Las librerías digitales han demandado mejores sistemas

[12] `http://www.nist.gov/tac/tracks/index.html`
[13] `https://www.wikipedia.org/`
[14] `http://www.cipsc.org.cn/clp2012/task2.html`

de desambiguación [Smalheiser y Torvik, 2009], por lo que en la actualidad sigue siendo un reto abierto para la comunidad científica [Geiß y Gertz, 2016; Momeni y Mayr, 2016].

Pese a que la desambiguación de nombres de persona se ha estudiado en distintos contextos, el escenario de búsqueda en la Web presenta unas características especiales que no permiten que sea tratado de forma similar. Por un lado, no se dispone de ninguna base de conocimiento que albergue las entidades ya desambiguadas. Esto implica que el proceso de desambiguación no puede basarse en una selección de posibles entidades candidatas pertenecientes a una base de conocimiento para decidir a cuál de ellas se refiere cada documento, a diferencia de los métodos de EL [Shen et al., 2015]. No obstante, el uso de bases de conocimiento como Wikipedia pueden facilitar el proceso de desambiguación [Dornescu et al., 2010; Long y Shi, 2010; Xu et al., 2015]. Por otro lado, las páginas web no tienen por qué ser documentos del mismo tipo como sucede en los contextos de las publicaciones científicas y las noticias. Las publicaciones científicas contienen una serie de rasgos específicos (autores, revista o congreso, fecha de publicación, ...) y las noticias suelen seguir una estructura informativa determinada basada en lo que se conoce en la jerga periodística como las *cinco W y una H* (*What?*, *Who?*, *When?*, *Why?*, *Where?*, *How?*). Esto implica que el proceso de desambiguación no puede basarse en rasgos particulares de un determinado contexto o en la asunción de que todos los documentos siguen una determinada estructura, puesto que no puede asumirse que dichos rasgos estén presentes en todos los resultados de búsqueda ni que todos ellos respeten un mismo estilo de escritura.

1.1.3. Líneas abiertas en la desambiguación de nombres de persona en la Web

La desambiguación de nombres de persona en un contexto de búsqueda en la Web se ha abordado en los últimos años como un problema de *clustering* donde no se conoce de antemano el número de *clusters* (individuos que comparten el mismo nombre). No obstante, hay ciertos aspectos del problema que se pueden mejorar o no han sido tratados:

- **Robustez:** la gran mayoría de sistemas propuestos en el estado del arte requieren de datos de entrenamiento para aprender y prefijar el valor de determinados parámetros utilizados por el algoritmo de *clustering* que agrupa los resultados de búsqueda. Estos parámetros son empleados para decidir si varias páginas web se refieren a un mismo individuo, estimar el número de individuos mencionados en los resultados de búsqueda o definir un criterio de parada del proceso de desambiguación. Artiles [2009] concluyó que los sistemas basados en esta metodología

son muy sensibles con respecto al valor de estos parámetros, puesto que sus resultados dependen, entre otros factores, del grado de ambigüedad de los nombres de persona. Según cuales sean las características de los datos de entrenamiento, se pueden obtener valores sesgados que no sean de utilidad para otros datos con características distintas, lo que puede conllevar a que estos métodos sean poco robustos. Por otro lado, la obtención de suficientes datos de entrenamiento representativos para garantizar la consistencia de los resultados con diferentes nombres de persona y en un entorno Web cambiante requiere de un gran esfuerzo por parte de anotadores expertos.

- **Redes sociales:** los principales sistemas de desambiguación de nombres de persona han sido evaluados con colecciones recopiladas cuando estas plataformas no existían o tenían poco impacto en la Red, de manera que no consideraron este fenómeno. No obstante, Berendsen [2015] concluyó que la aparición de estas páginas web puede afectar a los resultados obtenidos por sistemas de desambiguación de nombres de persona y deben tratarse de manera diferenciada.

- **Multilingüismo:** las colecciones de evaluación de desambiguación de nombres de persona en la Web se han centrado en un escenario monolingüe. Por este motivo, los métodos propuestos no han tenido en cuenta el multilingüismo a la hora de resolver el problema. No obstante, dado el carácter multilingüe de la Web, se requieren nuevas técnicas capaces de tratar esta situación de manera efectiva [Montalvo et al., 2015a].

1.2. Objetivos e hipótesis

El objetivo principal de esta tesis doctoral consiste en definir y analizar nuevas estrategias de desambiguación de nombres de personas en la Web capaces de manejar el impacto de las redes sociales y el multilingüismo y que no dependa de datos aprendidos por entrenamiento. Además, con el fin de que los métodos propuestos en este trabajo sean de utilidad en un escenario real, se tomará en cuenta que no involucren un alto coste computacional.

Las hipótesis en las que se basa la investigación de este trabajo pueden dividirse en base a los problemas abiertos descritos anteriormente:

- **Robustez:** se enuncian dos hipótesis. Por un lado, la primera hipótesis establece qué clase de rasgos son adecuados para decidir si dos páginas web hablan del mismo individuo, es decir, se centra en la representación de los documentos. Por otro lado, la segunda hipótesis establece un criterio que decide si dos páginas web

hablan del mismo individuo, de modo que no se requiera la utilización de datos de entrenamiento.

  - **Representación de los documentos:** *la compartición de n-gramas compuestos por palabras escritas en mayúsculas (la letra inicial) es un criterio eficaz a la hora de decidir si dos páginas web hablan del mismo individuo.*
  - **Comparación entre documentos:** *se puede decidir si dos sitios web hablan del mismo individuo a partir de su similitud y de un valor de umbral obtenido mediante una función matemática que dependa exclusivamente de las características de cada uno de ellos.*

- **Redes sociales:** se establece como hipótesis que el impacto negativo de la presencia de redes sociales en el problema se debe a la comparación entre páginas web de la misma plataforma:

  Las páginas web de la misma red social pueden dar lugar a agrupaciones erróneas debido a que enlazan a perfiles de distintos individuos llamados de la misma manera y comparten vocabulario común.

- **Multilingüismo:** una primera aproximación para tratar el multilingüismo consiste en emplear recursos de traducción automática. No obstante, el empleo de estas herramientas implica necesariamente un incremento del coste computacional del proceso de desambiguación. La siguiente hipótesis establece qué tipo de rasgos pueden ser útiles a la hora de decidir si dos páginas web hablan de un mismo individuo sin necesidad de emplear recursos de traducción:

  Se puede decidir si dos páginas web escritas en distinto idioma hablan del mismo individuo a partir de rasgos que se escriben de la misma manera en ambos idiomas, sin necesidad de utilizar recursos de traducción adicionales (traducción automática, diccionarios, . . .).

La metodología empleada para alcanzar el objetivo de esta tesis doctoral ha sido la siguiente:

- Estudiar el estado del arte en torno a la desambiguación de nombres de personas en el escenario de búsqueda en la Web poniendo especial énfasis en las representaciones de las páginas web y los algoritmos de *clustering* empleados.
- Establecer hipótesis sobre la representación de las páginas web. Evaluar la validez de las mismas usando algoritmos del estado del arte y colecciones de datos disponibles.
- Establecer hipótesis sobre cómo comparar las páginas web sin necesidad de umbrales o parámetros aprendidos a partir de datos de entrenamiento. Evaluar la

validez de las mismas usando algoritmos del estado del arte y colecciones de datos disponibles.

- Proponer nuevos algoritmos de *clustering* en base a las hipótesis establecidas anteriormente. Evaluar la robustez de los algoritmos propuestos y comparar su rendimiento con respecto a los principales sistemas del estado del arte. Analizar los resultados y extraer conclusiones sobre sus ventajas e inconvenientes.

- Estudiar y analizar el impacto de las redes sociales en el problema. Establecer hipótesis sobre el tratamiento de este tipo de páginas web y evaluar su validez. Plantear técnicas para tratar las páginas web sociales en base a las hipótesis anteriores. Evaluar las propuestas con colecciones que contengan páginas web pertenecientes a estas plataformas. Analizar los resultados y extraer conclusiones sobre sus ventajas e inconvenientes.

- Estudiar y analizar el impacto del multilingüismo mediante el uso de una herramienta de traducción automática. Enunciar hipótesis sobre el tratamiento del multilingüismo en el problema y evaluar su validez. Proponer un método que trate el multilingüismo sin hacer uso de recursos de traducción para evitar que se incremente el coste computacional del proceso de desambiguación. Evaluar la propuesta con una colección de datos que contenga páginas web escritas en distintos idiomas. Analizar los resultados y extraer conclusiones sobre sus ventajas e inconvenientes.

- Resumir las conclusiones extraídas tras la investigación. Recopilar las principales contribuciones y establecer líneas de trabajo futuras.

1.3. Estructura de la memoria

A continuación, se hace un breve resumen del contenido presentado en el resto de capítulos de esta memoria:

- **Capítulo 2:** se presenta una revisión del estado de la cuestión en desambiguación de nombres de persona en el escenario de búsqueda en la Web. Se presenta una tipología de las principales técnicas de acuerdo a los rasgos empleados para capturar el contenido de las páginas web, el modelo de representación de documentos utilizado, los algoritmos de *clustering* empleados para agrupar los resultados de búsqueda y otros aspectos como el requerimiento de datos de entrenamiento o el uso de recursos externos.

- **Capítulo 3:** se presenta el marco de experimentación utilizado en las propuestas presentadas en este trabajo de investigación. En primer lugar, se describe brevemente el modelo de representación de documentos, las diferentes funciones de pesado de términos y las medidas de similitud entre documentos que se emplearán a lo largo de este trabajo. A continuación, se detalla la configuración de los algoritmos de *clustering* del estado del arte que serán empleados en estudios preliminares. Posteriormente, se presenta el marco de evaluación utilizado en los experimentos llevados a cabo en este trabajo. Por un lado, se detallan las características de las colecciones empleadas en la experimentación y se analizan sus principales diferencias. Por otro lado, se definen las métricas de evaluación y el test de significancia estadística empleados para identificar mejoras significativas entre distintos experimentos.

- **Capítulo 4**: se presenta un primer algoritmo de desambiguación de nombres de personas que evita la necesidad de usar datos de entrenamiento. Se presentan los resultados de la propuesta comparándolos con los obtenidos por los principales métodos del estado del arte en distintas colecciones. Se realiza un análisis de los resultados desde varias perspectivas. Finalmente, se estudian las ventajas e inconvenientes del algoritmo propuesto.

- **Capítulo 5:** se presenta un segundo algoritmo de desambiguación de nombres de personas. Este método, compuesto por varias fases, consiste en una extensión del primer algoritmo, de modo que resuelve sus principales limitaciones. Se analizan los resultados de cada una de las fases del algoritmo y se comparan sus resultados con respecto a los obtenidos por los principales métodos del estado del arte en varias colecciones.

- **Capítulo 6:** se estudia el impacto de las redes sociales en el problema y se proponen dos métodos heurísticos para tratar este tipo de páginas web de manera diferenciada. A continuación, se evalúan los resultados de las propuestas presentadas con dos colecciones caracterizadas por incluir un alto porcentaje de páginas web de este tipo. Por último, se analizan las ventajas e inconvenientes de cada una de ellas.

- **Capítulo 7:** se presenta la propuesta final de esta tesis para la desambiguación de nombres de personas en la Web, consistente en una generalización para el escenario multilingüe. En primer lugar, se estudia la utilidad de una herramienta comercial de traducción automática en el problema. Posteriormente, se presenta una propuesta para tratar este problema sin hacer uso de este tipo de herramientas u otros recursos externos con el fin de no aumentar el coste computacional del proceso de desambiguación. Finalmente, se analizan los resultados obtenidos por

la propuesta sobre una colección caracterizada por contener resultados de búsqueda reales escritos en diferentes idiomas para diferentes nombres de persona.

- **Capítulo 8:** se presentan las principales conclusiones y contribuciones de este trabajo de investigación y posibles líneas de trabajo futuro.

2
Estado del arte

"Un hombre provisto de papel, lápiz y goma, y sujeto a una disciplina estricta, es en efecto una máquina [de Turing] universal."

— Alan Turing —

En este capítulo se presenta un resumen de los principales trabajos sobre desambiguación de nombres de personas en la Web. En primer lugar, se introduce brevemente la manera general en la que se ha abordado el problema mediante dos fases principales. Por un lado, una primera fase de representación de documentos cuyo principal objetivo es seleccionar rasgos adecuados a la hora de distinguir entre individuos diferentes con el mismo nombre. Por otro lado, una fase de agrupamiento consistente en la aplicación de un algoritmo de clustering que agrupe adecuadamente los resultados de búsqueda de acuerdo al individuo al que se refieren. Se repasarán las técnicas de representación de páginas web y algoritmos de clustering utilizados por los sistemas de desambiguación. A continuación, se revisará el tratamiento que se ha realizado sobre las redes sociales, puesto que se trata de un tipo de páginas web que suelen aparecer habitualmente cuando se consultan nombres de persona en los motores de búsqueda. Posteriormente, repasaremos de qué manera se ha afrontado el multilingüismo en el problema. Tras ello, se presenta una clasificación de los mejores sistemas del estado del arte de acuerdo con los factores anteriores. Por último, se presentan las conclusiones que resumen el contenido del capítulo, haciendo especial hincapié en aquellas cuestiones abiertas que serán tratadas en esta tesis con posterioridad.

2.1. Introducción

Los problemas de desambiguación suponen un reto abierto dentro de las áreas de TM, NLP e IR desde hace décadas. La resolución de la desambiguación es clave para que una computadora sea capaz de comprender el lenguaje natural y es de gran utilidad para otro tipo de problemas como la resolución de co-referencias o la mejora de la relevancia de resultados por parte de motores de búsqueda. Por ejemplo, el problema de la desambiguación del sentido de las palabras (*Word Sense Disambiguation*, WSD), consistente en determinar el correcto significado de una palabra polisémica en un cierto

contexto, lleva estudiándose desde finales de la década de los 40 del siglo XX [Navigli, 2009].

Durante la década de los años 90, la comunidad científica puso el foco en tareas consistentes en identificar NEs. El concepto de NE surgió en la sexta conferencia MUC (*Message Understanding Conference*, [Grishman y Sundheim, 1996]) dedicada a la Extracción de Información (*Information Extraction*, IE). De acuerdo con Chinchor [1997], una NE es una unidad informativa perteneciente a alguna de las siguientes categorías:

- Entidades (ENAMEX): personas (PER), organizaciones (ORG) o lugares (LOC).
- Expresiones temporales (TIMEX): horas o fechas.
- Expresiones numéricas (NUMEX): divisas o porcentajes.

Posteriormente, en las conferencias ConLL (*Conference of Natural Language Learning*, [Sang, 2002; Sang y de Meulder, 2003]) se extendió el concepto de NE añadiendo otro tipo de información no perteneciente a las categorías anteriores y recogido en una nueva categoría miscelánea (MISC) que abarca eventos o sucesos históricos (ej. *Segunda Guerra Mundial*), obras de arte (ej. *La Monalisa*), etc. Por otro lado, Sekine et al. [2002] propusieron una clasificación más específica consistente en 150 tipos de NEs. Más adelante, el programa ACE (*Automatic Content Extraction*, [Doddington et al., 2004]), sucesor de las conferencias MUC, centró el reconocimiento de NEs en siete categorías: *Persons*, *Organizations*, *Locations*, *Facilities*, *Vehicles*, *Weapons* y *Geo-Political Entities*.

El problema del reconocimiento de NEs (*Named Entity Recognition*, NER) consiste en identificar este tipo de información en un texto, pero no es suficiente a la hora de decidir a qué entidades concretas se refiere. Por ejemplo, la NE *Hollywood* puede hacer referencia a varias localidades estadounidenses o a la industria del cine de aquel país. Al igual que sucede con el problema WSD, se hace imprescindible decidir a qué entidades hace referencia un texto como paso previo para que una computadora sea capaz de comprenderlo o para relacionar textos diferentes entre sí. De esta necesidad nace el problema de la desambiguación de NEs (*Named Entity Disambiguation*, NED).

En particular, la comunidad científica ha puesto un especial interés en la desambiguación de nombres de personas debido a que son un tipo de NE especialmente ambiguo y algunos autores [Bagga y Baldwin, 1998; Winchester y Lee, 2002; Gooi y Allan, 2004] solamente se han centrado en la desambiguación de esta clase de NEs. Debido al amplio uso de motores de búsqueda por parte de los usuarios de Internet, la desambiguación de nombres de personas comenzó a estudiarse en un contexto de búsqueda en la Web [Mann y Yarowsky, 2003; Al-Kamha y Embley, 2004; Bekkerman y McCallum, 2005; Wan et al., 2005; Bollegala et al., 2006]. No obstante, la mayoría de las colecciones

de evaluación empleadas por estos autores eran muy pequeñas (ej. [Mann y Yarowsky, 2003; Bekkerman y McCallum, 2005; Bollegala et al., 2006]) o estaban sesgadas en algún sentido, como en el caso de la colección empleada por Wan et al. [2005], donde la mayoría de los nombres de personas se corresponden con celebridades. Posteriormente, en el año 2007, surge la primera campaña WePS [Artiles et al., 2007] con el fin de abordar un doble objetivo: por un lado, formular la desambiguación de nombres de personas en la Web como un problema de *clustering* y, por otro lado, presentar colecciones de nombres de persona que no tuviesen las limitaciones anteriores. Desde entonces, la mayoría de trabajos que han tratado el problema han asumido el escenario propuesto en WePS y han sido evaluados mediante las colecciones proporcionadas en estas campañas. La gran mayoría de estos sistemas de desambiguación tienen en común que se componen de las dos fases que se presentan a continuación:

- **Representación de documentos:** el objetivo de esta fase consiste en seleccionar rasgos adecuados para distinguir individuos llamados igual y emplear un modelo de representación de documentos para poder comparar las páginas web entre sí.
- **Agrupación de resultados de búsqueda:** el objetivo de esta fase consiste en la aplicación de un algoritmo de *clustering* para agrupar las páginas web de acuerdo al individuo que mencionan.

La sección 2.2 presenta los diferentes tipos de rasgos y modelos de representación empleados por los sistemas de desambiguación en la fase de representación de documentos. A continuación, la sección 2.3 presenta los diferentes algoritmos de *clustering* empleados por los sistemas de desambiguación en la fase de agrupación de los resultados de búsqueda. Posteriormente, las secciones 2.4 y 2.5 revisan, respectivamente, los trabajos del estado del arte que han tratado el impacto de las redes sociales y el multilingüismo en el problema. Tras ello, la sección 2.6 presenta una clasificación de los sistemas del estado del arte teniendo en cuenta los factores mencionados anteriormente. Por último, la sección 2.7 resume las principales conclusiones extraídas tras la revisión de los trabajos del estado del arte en desambiguación de nombres de personas en la Web.

2.2. Representación de documentos

La representación de los documentos es una fase crucial a la hora de poder procesarlos automáticamente de una manera adecuada. Para ello, en primer lugar se debe identificar qué tipo de vocabulario puede ser de utilidad a la hora de decidir si dos

documentos se refieren a un mismo individuo. Posteriormente, debe realizarse una adecuada selección de rasgos de cada documento, de manera que contengan información útil para poder agruparlos correctamente. Finalmente, debe utilizarse un modelo de representación de toda la colección de documentos que permita identificar la relevancia de cada uno de los rasgos seleccionados y comparar cómo de similares son los documentos entre sí.

La fase de representación de documentos llevada a cabo por los sistemas de desambiguación de nombres de persona puede dividirse en los siguientes pasos:

- *Preprocesamiento*: consiste en la aplicación de un conjunto de técnicas para procesar los documentos desde su formato de origen.
- *Selección de rasgos*: se escogen rasgos que permitan distinguir adecuadamente a distintos individuos llamados de la misma manera, evitando otro tipo de información que pueda añadir ruido.
- *Modelo de representación de documentos*: los documentos se representan en base a un determinado modelo matemático. Normalmente, este modelo asigna a cada rasgo seleccionado un valor que representa su importancia con respecto al documento al que pertenece o sobre toda la colección de documentos.

La mayoría de los principales métodos de desambiguación del estado del arte han empleado las colecciones de datos proporcionadas por las tres ediciones de las campañas de evaluación WePS [Artiles et al., 2007, 2009b, 2010]. El amplio número de sistemas de desambiguación propuestos para este problema ha permitido estudiar y comparar el impacto de las múltiples técnicas empleadas en cada uno de los pasos descritos anteriormente. Los siguientes apartados resumen detalladamente las técnicas empleadas, poniendo especial énfasis en el impacto en los resultados de cada una de ellas.

2.2.1. Preprocesamiento

De acuerdo con Sedding y Kazakov [2004], el preprocesamiento tiene una relevancia similar a la elección del algoritmo de *clustering* empleado, dado que este último obtendrá buenos resultados solo si sus datos de entrada fueron generados correctamente. Por tanto, durante la etapa de preprocesamiento se debe identificar qué información contenida en los documentos puede ser relevante y cuál no, de manera que se pueda eliminar esta última con el fin de evitar la introducción de ruido y al mismo tiempo reducir el tamaño del vocabulario para que el proceso de agrupamiento de resultados de búsqueda pueda llevarse a cabo con menor coste computacional. Además, se realizan todos

aquellos procesamientos lingüísticos y no lingüísticos que faciliten la posterior selección de rasgos.

Las técnicas de preprocesamiento empleadas habitualmente por los sistemas de desambiguación nombres de personas en la Web son las siguientes:

Obtención del texto plano

La entrada consiste en un ranking de resultados devuelto por un motor de búsqueda cuando se consulta un nombre de persona. Pese a que los motores de búsqueda son capaces de devolver enlaces a archivos de cualquier tipo ubicados en la Red, lo más habitual es que los resultados de búsqueda obtenidos se correspondan con páginas web. Por este motivo, el primer paso desempeñado por los sistemas de desambiguación consiste en obtener el texto plano de las páginas web mediante diferentes herramientas de análisis sintáctico (o *parsers*) capaces de detectar y eliminar el código fuente contenido en este tipo de documentos. Adicionalmente, estos recursos son útiles a la hora de obtener información que normalmente las páginas web contienen en determinadas etiquetas del lenguaje HTML (*HyperText Markup Language*). Este es el caso del título de la página web, los hipervínculos, los metadatos, palabras resaltadas en negrita o cursiva, etc. Algunas de las herramientas utilizadas por los sistemas de desambiguación para este fin son *Beatiful Soup*[1] [Chen et al., 2012; Liu et al., 2011] o *HTML Parser*[2] [Nuray-Turan et al., 2012]. Ferrés y Rodríguez [2010] concluyeron que el correcto filtrado de información en documentos HTML es un paso crucial a la hora de evitar ruido que puede afectar negativamente en el rendimiento de los sistemas de desambiguación.

Análisis léxico

El objetivo de esta fase consiste en identificar el léxico presente en el texto plano de las páginas web, de forma que se pueda generar el vocabulario empleado para representar los resultados de búsqueda. Generalmente, suelen tomarse como rasgos iniciales las palabras separadas por espacios en blanco. Por otro lado, durante esta etapa se eliminan algunos caracteres no alfanúmericos (ej. *$*, *, ...) dado que se considera que no aportan ninguna información de utilidad. En cambio, otros caracteres especiales se mantienen puesto que pueden servir para identificar determinada información útil. Por ejemplo, los puntos (.) son útiles a la hora de dividir el texto en oraciones e identificar siglas (ej. *U.N.E.D.*), o el símbolo @ es útil a la hora de identificar si una cadena de caracteres se corresponde con una dirección de correo electrónico. Lan et al. [2009] concluyeron que llevar a cabo un análisis léxico meticuloso conlleva a mejoras en los resultados con respecto a un análisis léxico más superficial para el problema.

[1] `https://www.crummy.com/software/BeautifulSoup/`

[2] `http://htmlparser.sourceforge.net/`

***Stemming* y lematización**

El objetivo de las técnicas de *stemming* y lematización consiste en asignar una única forma canónica para aquellas palabras formadas a partir de un mismo lexema bajo el supuesto de que todas ellas suelen representar el mismo concepto. En particular, ambos métodos son útiles a la hora de identificar las múltiples formas flexivas de una determinada palabra. Por ejemplo, las formas flexivas de un verbo suelen corresponderse con sus conjugaciones, de manera que algunas formas flexivas del verbo *correr* son *corro*, *corres*, etc. En el caso de los sustantivos y adjetivos, su género y número constituyen formas flexivas, de manera que, por ejemplo, *gata* y *gatos* son formas flexivas de *gato*.

El resultado de aplicar *stemming* a una cierta palabra consiste en la obtención de un fragmento de la misma, denominado *stem*, común a todas sus formas flexivas. Para ello, los algoritmos de *stemming*, denominados *stemmers*, eliminan los prefijos y sufijos contenidos en la palabra. Por ejemplo, *corr-* es el *stem* del verbo *correr* y sus conjugaciones verbales, mientras que *gat-* es el *stem* de *gato* y sus formas flexivas. Esta técnica ha sido utilizada por la mayoría de los sistemas del estado del arte (ej. [Liu et al., 2011; Nuray-Turan et al., 2012]). Puesto que el problema se ha tratado mayoritariamente en lengua inglesa, los métodos de desambiguación han empleado el *algoritmo de Porter* [Porter, 1980], que es el *stemmer* más ampliamente utilizado por la comunidad científica para dicho idioma. No obstante, existen algoritmos de este tipo distribuidos libremente en otros idiomas. *Snowball*[3] pone a disposición *stemmers* para diferentes idiomas, además de un marco que facilita la implementación de estas herramientas.

Por otro lado, la *lematización* consiste en asignar a cada palabra su *lema*, definido como la forma flexionada aceptada por convenio como representante de todas las demás. En el caso de los verbos, los lemas suelen corresponderse con su infinitivo (ej. *correr*), mientras que en el caso de los sustantivos y adjetivos, el lema suele ser el masculino singular (ej. *gato*). Esta técnica ha sido empleada por un menor número de sistemas [Kozareva et al., 2007; Artiles et al., 2009a] con respecto al *stemming*.

Mientras que el *stemming* suele aplicarse a palabras individuales sin requerir ningún tipo de información adicional, la lematización necesita conocer el contexto de cada palabra dentro del texto donde aparece junto con su categoría gramatical. Por este motivo, esta última técnica debe emplearse junto con recursos lingüisticos adicionales que permitan obtener esa información.

No parece estar claro el impacto de estas técnicas en los resultados de los sistemas de desambiguación. Por un lado, Martínez-Romo y Araujo [2009] han señalado que el *stemming* no aporta mejoras significativas, mientras que Balog et al. [2009] y Monz y Weerkamp [2009] concluyen que su aplicación obtiene mejores resultados. Por otro

[3] `http://snowballstem.org/`

lado, Artiles et al. [2009a] indica que la lematización no aporta mejoras significativas en los resultados. No obstante, dado que estas técnicas sirven para representar varias palabras mediante una única forma canónica, su aplicación tiene como consecuencia que se reduce el vocabulario a usar en la posterior representación de los documentos, obteniendo una representación más compacta, lo cual es positivo en términos de coste computacional.

Eliminación de palabras vacías y palabras poco frecuentes

Es habitual eliminar del vocabulario aquellas palabras denominadas *palabras vacías* (o *stop words*) en problemas relacionados con NLP, IR y TM. Se trata de palabras de uso frecuente en un determinado idioma entre las que normalmente se encuentran preposiciones, pronombres personales, artículos o verbos de uso común. La eliminación de este tipo de palabras se realiza por dos razones fundamentales. En primer lugar, puesto que estas palabras suelen corresponderse con nexos que ayudan a articular el discurso, su aportación semántica no es relevante a la hora de comprender el contenido de un texto. Por otro lado, son palabras de poca utilidad a la hora de discriminar el contenido de distintos documentos, puesto que aparecen en la mayoría de ellos. Siguiendo la misma lógica, los sistemas de desambiguación tratan el nombre de persona introducido como consulta como otra palabra vacía, dado que los resultados devueltos por el motor de búsqueda consistirán en páginas web en donde aparece dicho nombre. Algunos sistemas utilizan listas de *stop words* mucho más amplias de las empleadas habitualmente, como es el caso de Jiang et al. [2009]. Por su parte, Nagy [2012] utiliza una lista de *stop words* elaborada manualmente que incluye términos habituales en Internet que no aportan ninguna información relevante, como por ejemplo *webmaster*, *wiki*, *support*, ...

Por otro lado, también es habitual que se eliminen palabras que aparecen de manera poco frecuente en la colección de documentos. En particular, se trata de palabras que aparecen en un único documento o en muy pocas ocasiones en toda la colección. Dada la poca frecuencia de aparición de estas palabras, se considera que tienen una baja capacidad discriminativa. Según Aggarwal y Zhai [2012], en el dominio Web este tipo de palabras se corresponden con información ruidosa que dificulta la agrupación de documentos entre sí.

2.2.2. Selección de rasgos

De acuerdo con Xu et al. [2016], la selección adecuada de rasgos que representen el contenido de los documentos es una pieza clave a la hora de poder mejorar los resultados de los métodos basados en técnicas de *clustering*. Los sistemas del estado del arte han empleado un amplio abanico de rasgos de diferente naturaleza que podemos dividir en los siguientes tipos de acuerdo a la manera en la que son extraídos:

- **Rasgos lingüísticos:** se trata de palabras pertenecientes a ciertas categorías gramaticales o que cumplen algún tipo de propiedad lingüística, de modo que su extracción requiere llevarse a cabo mediante el uso de recursos de NLP. Por ejemplo, dentro de este tipo de rasgos se encuentran las NEs, los sintagmas nominales, los verbos, los adjetivos, los sustantivos, etc.

- **Rasgos no lingüísticos:** se trata de contenido textual para el que no se requiere utilizar ningún tipo de recurso lingüístico para su extracción. Dentro de este tipo de rasgos se encuentran las palabras, los *n*-gramas y los *k-skip-n*-gramas.

- **Rasgos web:** se trata de información de las páginas web proporcionada por el motor de búsqueda o extraída a partir de ciertas etiquetas del lenguaje HTML. Por ejemplo, los motores de búsqueda muestran para cada resultado de búsqueda tanto su posición en el ranking como una caja de información denominada *snippet*. Por su parte, el código fuente de las páginas web contiene etiquetas especiales que permiten extraer información que no siempre aparece dentro del texto de la misma, como por ejemplo los hipervínculos y los metadatos.

- **Información personal:** se trata de datos personales de los individuos que comparten el mismo nombre de persona. La obtención de esta información se lleva a cabo mediante técnicas de Extracción de Atributos (*Attribute Extraction*, AE). Entre la amplia variedad de información biográfica extraída destacan las fechas de nacimiento y/o muerte, lugares de nacimiento y/o muerte, profesiones, titulaciones académicas o nombres de familiares.

- **Rasgos externos:** se trata de rasgos extraídos a partir de fuentes externas como por ejemplo bases de datos, enciclopedias, diccionarios o consultas auxiliares en motores de búsqueda.

Dado que el problema WePS debe resolverse en tiempo real, hay que tener en cuenta el coste computacional que supone la extracción de los rasgos. Por un lado, los rasgos web y los no lingüísticos pueden extraerse de manera poco costosa durante el preprocesamiento mediante el uso de *parsers* HTML y herramientas de tokenización. En cambio, la obtención de rasgos lingüísticos e información personal es más costosa puesto que requiere de recursos adicionales para su extracción. Finalmente, la extracción de rasgos externos es especialmente costosa porque implica necesariamente el procesamiento de una mayor cantidad de información, como sucede con la exploración de artículos de Wikipedia o la explotación de consultas auxiliares en motores de búsqueda.

A continuación, se detalla el papel de cada uno de los tipos de rasgos descritos anteriormente en los sistemas de desambiguación de nombres de personas en la Web.

Rasgos lingüísticos

Artiles [2009] destaca que las NEs son uno de los rasgos más utilizados por los sistemas de desambiguación de nombres de personas. En particular, los sistemas de desambiguación normalmente han empleado las siguientes categorías de NEs: nombres de persona (PER), localidades (LOC) y organizaciones (ORG). No obstante, otros sistemas incluyen otros tipos de NEs. Por ejemplo, Popescu y Magnini [2007] también incluyen expresiones temporales, mientras que Artiles et al. [2009a] emplean un sistema de reconocimiento de NEs descrito en Sekine [2008] que distingue 100 categorías de NEs más específicas. La extracción de este tipo de rasgos se ha llevado a cabo mediante sistemas NER, siendo empleado habitualmente el sistema descrito por Finkel et al. [2005], que distribuye libremente la Universidad de Stanford[4].

El impacto de las distintas categorías de las NEs para desambiguar nombres de personas ha sido estudiado por algunos autores. Ikeda et al. [2009] asignan mayor relevancia a las NEs de las categorías PER y ORG con respecto a las de tipo LOC en base a resultados obtenidos con datos de entrenamiento. Por su parte, Artiles et al. [2009a] concluyen que las NEs de la categoría LOC logran peores resultados con respecto a las otras categorías, y las NEs de las categorías ORG y PER logran resultados muy pobres de cobertura, por lo que no constituyen información suficiente para poder representar convenientemente las páginas web. En este sentido, Nuray-Turan et al. [2012] señalan que muchas páginas web incluyen nombres de localidades muy generales, tales como nombres de países (ej. *United States*), que son muy ambiguas porque son comunes a muchos individuos y, por tanto, no son útiles durante el proceso de desambiguación. Estos autores también advierten de que algunas NEs de la categoría PER tienen un impacto perjudicial en el proceso de desambiguación. En particular, aconsejan evitar aquellas entidades de personas consistentes en un único nombre (ej. *John*) porque son demasiado ambiguas y pueden dar lugar a agrupar incorrectamente los resultados de búsqueda.

[Artiles et al., 2009a] comparan los resultados obtenidos mediante una representación basada en varios rasgos no lingüísticos (palabras y n-gramas) con respecto a esa misma representación añadiendo NEs, llegando a la conclusión de que estas últimas tienen un pequeño efecto negativo en los resultados y su rendimiento depende del sistema NER que se haya empleado para su extracción. No obstante, también concluyen que estos rasgos son de utilidad a la hora de caracterizar cada uno de los grupos resultantes del proceso de desambiguación, por lo que ayudarían al usuario a seleccionar el conjunto de páginas web del individuo que le interesa.

Además de las NEs, unos pocos sistemas de desambiguación han hecho uso de otros rasgos lingüísticos caracterizados por pertenecer a ciertas categorías gramaticales

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

o por su función en las oraciones en las que aparecen, por lo que su extracción se realiza mediante el empleo de herramientas de *chunking* o de etiquetado gramatical (*part of speech tagging*). Chen y Martin [2007a] asumen que los sintagmas nominales proporcionan información de utilidad sobre los individuos. No obstante, de acuerdo con Artiles et al. [2009a], usar exclusivamente estos rasgos en la representación presenta un bajo rendimiento. Por otro lado, Rao et al. [2007] consideran que los sustantivos y los adjetivos son las únicas clases de palabras que proporcionan una información semántica relevante, y Xu et al. [2015] toman también en cuenta los verbos. No obstante, ambos sistemas no analizan el impacto de este tipo de rasgos.

Rasgos no lingüísticos

De acuerdo con [Artiles, 2009], la mayoría de los sistemas de desambiguación han representado los resultados de búsqueda a partir de rasgos obtenidos tras un proceso de *tokenización* del texto sin el requerimiento de ningún tipo de recurso lingüístico adicional, por lo que su extracción implica un menor coste computacional con respecto a la de los rasgos lingüísticos. En particular, los rasgos de este tipo más utilizados son las palabras (o *tokens*) incluidas en las páginas web. Además, algunos sistemas han empleado rasgos compuestos por varias palabras, en particular *n*-gramas y *k-skip-n*-gramas.

Los sistemas de desambiguación han tomado dos políticas diferentes a la hora de extraer los rasgos de este tipo. Por un lado, algunos sistemas (ej. [Elmacioglu et al., 2007; Gong y Oard, 2009; Long y Shi, 2010; Berendsen, 2015]) han aplicado una política de *extracción completa de rasgos* que toma en consideración todo el contenido de los resultados de búsqueda. Por otro lado, otras propuestas (ej. [Ono et al., 2008; Ikeda et al., 2009; Romano et al., 2009; Xu et al., 2015]) han aplicado una política de *extracción local de rasgos* por la que se toma en consideración las palabras que aparecen en los párrafos o frases donde se menciona el nombre de persona consultado o, alternativamente, aquellas situadas alrededor del nombre de persona en un contexto definido a partir de un tamaño de ventana determinado. Finalmente, otros autores [Liu et al., 2011; Chen et al., 2012] representan las páginas web usando un vector de *rasgos completos* y otro compuesto por *rasgos locales*. De acuerdo con Artiles et al. [2009a], los *rasgos locales* tomados de las frases donde aparece el nombre de persona proporcionan altos valores de precisión, pero muy bajos valores de cobertura, por lo que no constituyen información suficiente para representar los documentos convenientemente. Por su parte, Monz y Weerkamp [2009] concluyeron que los *rasgos locales* que aparecen alrededor del nombre de persona sí mejoran los resultados con respecto a tomar todo el contenido del documento, pero advierten que esta política es sensible con respecto al valor que se prefije para el tamaño de ventana.

Normalmente los sistemas que representan las páginas web mediante palabras suelen utilizar algún criterio que decida cuáles de ellas son relevantes para el proceso de

desambiguación. Por ejemplo, González et al. [2009] y Venkateshan [2009] toman las palabras más frecuentes de cada documento, pero sus propuestas obtienen resultados muy pobres [Artiles et al., 2009b]. Habitualmente, esta selección de palabras suele realizarse mediante funciones de pesado de términos [Lefever et al., 2009; Song et al., 2009] o técnicas de extracción de palabras clave (o *keywords*) basadas en modelos probabilísticos [Xu et al., 2015].

Por otro lado, algunos sistemas competitivos [Popescu y Magnini, 2007; Yoshida et al., 2010; Chen et al., 2012; Nuray-Turan et al., 2012] han utilizado n-gramas para representar los resultados de búsqueda. Estos rasgos consisten en secuencias de n palabras que aparecen consecutivamente en un texto. Nótese que las palabras son un caso particular de estos rasgos cuando $n = 1$, por lo que también suelen denominarse *unigramas*. La elección de los n-gramas se justifica porque son rasgos capaces de recoger información precisa que caracterice a un único individuo (ej. *Professor University Alberta Canada*) [Popescu y Magnini, 2007]. Además, puesto que tienen en cuenta el orden de aparición de las palabras en los textos, los n-gramas también son capaces de capturar los tópicos de los que hablan las páginas web [Nuray-Turan et al., 2012]. Artiles et al. [2009a] muestran que cuanto mayor sea el número n de palabras de las que están formados, estos rasgos obtienen mejores valores de precisión debido a que su número es cada vez más escaso y es menos probable que sean compartidos por documentos diferentes.

Finalmente, Xu et al. [2012] han explorado el uso de *k-skip-n*-gramas obteniendo resultados prometedores. Estos rasgos consisten en combinaciones de n palabras en el orden en el que aparecen en un texto, pero permitiendo que entre ellas pueda haber saltos no superiores a k palabras. De acuerdo con Guthrie et al. [2006], estos rasgos pueden verse como una generalización de los n-gramas cuando $k = 0$ (no hay saltos). Esto implica que el número de *k-skip-n*-gramas es siempre mayor o igual al de los n-gramas, de modo que su uso implica tener que utilizar un vocabulario más extenso. La extracción de todos los *k-skip-n*-gramas posibles de un texto se realiza dividiéndolo en frases y fijando el valor de k para cada frase como el número de palabras que contiene. La elección de los *k-skip-n*-gramas se basa en que son capaces de recoger adecuadamente el contexto de cada palabra, por lo que han sido empleados en técnicas de semántica distribucional como los *word embeddings* [Mikolov et al., 2013].

Rasgos web

La información proporcionada por el motor de búsqueda y ciertas partes del código fuente de las páginas web ha sido utilizada por diferentes sistemas bajo la hipótesis de que es información poco ruidosa [Rao et al., 2007]. Por un lado, el motor de búsqueda indica la posición en el ranking de cada página web e incluye para cada una de ellas una caja de información denominada *snippet* compuesta por el título de la página web, un fragmento de texto de la misma donde aparecen los términos consultados y su URL.

La Figura 2.1 muestra un ejemplo de *snippet* proporcionado por Google tras consultar el nombre de persona *Javier Martínez*. Por otro lado, los metadatos de las páginas web suelen consistir en una serie de palabras clave relacionadas con su contenido que pueden ser de utilidad durante el proceso de desambiguación. Además, mediante los hipervínculos (o *links*) podemos determinar si dos resultados de búsqueda están relacionados entre sí.

Figura 2.1: *Snippet* generado por Google para un resultado de búsqueda devuelto tras consultar el nombre de persona *Javier Martínez*.

Algunos sistemas [Han y Zhao, 2009; Dornescu et al., 2010; Ferrés y Rodríguez, 2010; Liu et al., 2011] extraen las palabras contenidas en el título de las páginas web, los metadatos y el *snippet* tratándolas de la misma manera que los rasgos no lingüísticos. Sin embargo, hay autores [Lefever et al., 2009; Xu et al., 2015] que les asignan un mayor peso al considerar que se trata de palabras más relevantes que el resto. No obstante, de acuerdo con los resultados presentados por Nagy [2012], las representaciones basadas exclusivamente en estos rasgos obtienen peores resultados que las representaciones que toman todo el contenido textual. Por otro lado, los rasgos contenidos en las URLs (dominio, etc.) se extraen porque pueden corresponderse con información importante como, por ejemplo, nombres de empresas o universidades [Chen y Martin, 2007a]. Además, las URLs se han utilizado habitualmente junto con los *links* para determinar si dos páginas web están enlazadas [Elmacioglu et al., 2007; Yoshida et al., 2010; Nuray-Turan et al., 2012], asumiendo que en caso afirmativo ambas mencionan al mismo individuo. Algunos autores [Kozareva et al., 2007; Smirnova et al., 2010] hacen esta misma asunción cuando dos páginas web comparten *links* entre sí, sin que necesariamente se encuentre la URL de los resultados de búsqueda en el conjunto de hipervínculos.

Información personal

Los datos personales sirven de gran ayuda para poder caracterizar adecuadamente a un individuo y de esta manera poder distinguirlo de sus tocayos. La información de este tipo más frecuentemente utilizada incluye fechas y lugares de nacimiento y defunción, profesiones, titulaciones académicas, direcciones postales, *e-mails* y números de teléfono y fax. La extracción de estos datos se lleva a cabo mediante técnicas de AE basadas en el uso de patrones léxicos [Mann y Yarowsky, 2003], sistemas basados en reglas [Lefever et al., 2007], expresiones regulares [Dornescu et al., 2010] o la información contenida en diferentes *gazetteers* o diccionarios temáticos [Rao et al., 2007]. Por otro lado, se hace uso de sistemas NERs para la extracción de otros tipos de información personal, como los

lugares de residencia, los nombres de familiares o las empresas y asociaciones a las que está vinculado un individuo [Lan et al., 2009].

La mayoría de los rasgos de este tipo son utilizados de una manera similar a las URLs y los *links*, puesto que los sistemas asumen que dos resultados de búsqueda hablan del mismo individuo si existen coincidencias de algún dato biográfico entre ellos. No obstante, algunos de estos rasgos son utilizados para asumir lo contrario, como sucede en el caso de los *middle names*. Estos nombres aparecen habitualmente en nombres de persona de países angloparlantes situándose entre el nombre de pila (*first name*) y el apellido (*given name*), por ejemplo *John* es el *middle name* de *Donald John Trump*. Nuray-Turan et al. [2012] asumen que dos páginas web hablan de individuos diferentes cuando se detecta que no existen coincidencias entre los *middle names*.

En general, los datos personales suelen utilizarse junto con otro tipo de rasgos porque las páginas web no siempre incluyen este tipo de información [Song et al., 2009]. Por ejemplo, Jiang et al. [2009] presentan un sistema competitivo basado en el uso de información biográfica junto con NEs, *links* y URLs. No obstante, algunas propuestas se basan exclusivamente en información personal. En particular, tanto Han y Zhao [2009] como Martínez-Romo y Araujo [2009] basan el proceso de desambiguación en la identificación de la profesión de los individuos bajo la hipótesis de que muchas páginas web se corresponden con perfiles profesionales. Los pobres resultados obtenidos por estos métodos indican que es necesario ampliar la representación de los resultados de búsqueda con otro tipo de información.

Rasgos externos

Algunos sistemas de desambiguación han empleado fuentes externas de diversa índole para enriquecer la representación de los resultados de búsqueda. Entre los recursos externos más utilizados se encuentran la base de datos léxica *WordNet*[5] [Miller, 1995], la enciclopedia *online* Wikipedia y páginas web adicionales obtenidas a partir de consultas auxiliares solicitadas a los motores de búsqueda.

WordNet ha sido empleada por algunos sistemas [Lefever et al., 2007; Han y Zhao, 2009] con el objetivo de enriquecer la representación mediante información semántica. No obstante, Monz y Weerkamp [2009] advierten que el uso de este recurso tiene un impacto negativo en los resultados.

Entre los autores que han empleado Wikipedia se encuentran Long y Shi [2010], que extraen rasgos a partir de artículos recopilados manualmente. Por otro lado, Dornescu et al. [2010] exploran artículos de Wikipedia relacionados con el contenido de las páginas web automáticamente mediante el uso de la herramienta *Wikipedia Miner*[6]. Por su parte,

[5] `https://wordnet.princeton.edu/`

[6] `https://sourceforge.net/projects/wikipedia-miner/`

Xu et al. [2015] buscan las entradas de Wikipedia correspondientes a las NEs situadas alrededor del nombre de persona. Todos ellos concluyen que la información extraída de esta fuente es de gran ayuda para poder distinguir a los individuos entre sí.

Finalmente, otros autores [Rao et al., 2007; Nuray-Turan et al., 2012] destacan que la extracción de información adicional mediante consultas auxiliares en los motores de búsqueda tienen un impacto positivo en los resultados. En particular, Rao et al. [2007] extraen los *snippets* de las mil primeras páginas web devueltas por otro motor de búsqueda distinto tras consultar el nombre de persona. Por su parte, Nuray-Turan et al. [2012] construye nuevas consultas mediante la concatenación del nombre de persona y ciertos bigramas que aparecen en las páginas web originales y, posteriormente, extraen información de los primeros resultados devueltos por el motor de búsqueda.

Conclusión

En general, los métodos del estado del arte han representado los resultados de búsqueda mediante una rica selección de rasgos pertenecientes a los tipos descritos anteriormente. La elección de rasgos más popular ha consistido en una combinación de rasgos lingüísticos y no lingüísticos, siendo especialmente común el uso de palabras y NEs [Artiles, 2009]. Algunos autores [Balog et al., 2009; Berendsen, 2015] han resaltado que una representación basada exclusivamente en rasgos no lingüísticos es suficiente para obtener resultados competitivos. En particular, la mayoría de los sistemas más competitivos [Chen y Martin, 2007a; Long y Shi, 2010; Liu et al., 2011; Xu et al., 2015] tienen como característica común que han empleado este tipo de rasgos sin incluir necesariamente NEs, por lo que su uso no parece ser imprescindible para obtener resultados prometedores. Varios trabajos [Saggion, 2008; Monz y Weerkamp, 2009] han concluido que la representación mediante palabras obtiene mejoras significativas con respecto a utilizar una representación exclusivamente formada por NEs. Esto puede deberse por dos razones. Por un lado, Popescu y Magnini [2007] apuntan que la estructura de las páginas web puede dificultar la identificación de las NEs por parte de los sistemas NER. Por otro lado, Dornescu et al. [2010] afirman que los sistemas NER normalmente son entrenados empleando colecciones compuestas por noticias, de modo que presentan un peor rendimiento en contextos más ruidosos como el dominio Web.

El resto de tipos de rasgos normalmente se han utilizado para complementar la representación de las páginas web. En particular, los *rasgos web* y los datos biográficos se emplean como evidencias para determinar cuándo dos páginas web hablan del mismo individuo. No obstante, varios autores [Artiles et al., 2009a; Song et al., 2009; Nagy, 2012] concluyen que estas dos clases de rasgos no constituyen por sí mismas información suficiente para representar adecuadamente las páginas web y presentan unos resultados de cobertura muy bajos.

Finalmente, algunos autores [Rao et al., 2007; Long y Shi, 2010; Nuray-Turan et al., 2012] destacan que enriquecer la representación mediante *rasgos externos* tiene un efecto positivo en los resultados del proceso de desambiguación. En particular, el uso de información extraída de Wikipedia o de páginas web obtenidas mediante consultas adicionales es especialmente beneficioso con respecto al uso de información semántica como la proporcionada por WordNet.

2.2.3. Modelos de representación de documentos

Tras haber seleccionado los rasgos que contienen la información que será de utilidad para distinguir a los individuos, se debe emplear un modelo de representación de documentos que los convierta en objetos matemáticos y asigne un cierto valor de importancia a cada uno de los rasgos que contienen. En Fresno [2006] puede encontrarse una definición formal de *modelo de representación de documentos*.

Los modelos de representación de documentos pueden clasificarse de diferentes maneras. En particular, Huang y Kuo [2010] distinguen cinco tipos de modelos de representación de documentos atendiendo al marco matemático en el que se basan:

- **Modelos basados en teoría de conjuntos:** representan un documento como el conjunto de palabras que contiene. El más representativo es el *Modelo Booleano* propuesto por Lancaster y Gallup [1973], basado en el uso de teoría de conjunto y lógica booleana.

- **Modelos algebraicos:** representan un documento mediante un vector, una tupla o una matriz de palabras, de modo que cada una de ellas tiene asignado un valor de importancia obtenido mediante una función de pesado de términos. Los modelos de este tipo más utilizados son el *Modelo de Espacio Vectorial* (*Vector Space Model*, VSM) y el *Análisis de Semántica Latente* (*Latent Semantic Analysis*, LSA) presentado por Deerwester et al. [1990].

- **Modelos probabilísticos:** expresan la importancia de los rasgos mediante probabilidades. Dentro de esta categoría se encuentran *Análisis de Semántica Latente Probabilístico* (*Probabilistic Latent Semantic Analysis*, pLSA) propuesto por Hofmann [1999], el modelo de n-gramas y *Latent Dirichlet Allocation* (LDA) propuesto por Blei et al. [2003].

- **Modelos basados en grafos:** representan mediante un grafo cada documento o toda la colección de documentos. Salton et al. [1997] fueron de los primeros autores en proponer este tipo de modelo de lenguajes.

- **Modelos híbridos:** son aquellos que combinan varios de los modelos presentados anteriormente.

Los dos primeros tipos de modelos suelen basarse en la representación conocida como *bolsa de palabras* (*Bag of Words*, BoW) donde cada documento se representa mediante el conjunto de palabras que contiene sin añadir ningún tipo de información semántica al asumir que el orden de aparición de las mismas es irrelevante. No obstante, LSA es capaz de extraer el contenido conceptual de los documentos bajo la hipótesis de que las palabras que se utilizan en el mismo contexto suelen tener un significado similar. Pese a ello, se trata de una técnica costosa computacionalmente debido a que utiliza el método de factorización de matrices denominado *Descomposición en Valores Singulares* (*Singular Value Decomposition*, SVD).

En cuanto a los métodos probabilísticos, destacan pLSA y LDA. Ambas técnicas extraen los temas (denominados *tópicos*) de los que trata el contenido de los documentos. Se considera que un tópico consiste en un conjunto de términos relacionados entre sí que aparecen en los documentos. Ambos métodos constituyen por sí mismos técnicas de clasificación de documentos, puesto que son capaces de agruparlos en base a los tópicos extraídos. Girolami y Kabán [2003] prueban que ambas técnicas están relacionadas entre sí, dado que pLSA puede verse como un estimador máximo a posteriori de LDA bajo determinadas condiciones especiales.

Finalmente, los modelos basados en grafos tienen la ventaja de que son capaces de conservar la estructura de los documentos originales y suelen ajustarse especialmente bien al dominio Web [Brin y Page, 1998]. No obstante, son métodos que suelen requerir mayor coste en espacio y, puesto que muchos algoritmos que operan sobre grafos tratan problemas NP-duros [Nastase et al., 2015], también pueden implicar un alto coste computacional temporal.

Los modelos utilizados por los sistemas de desambiguación de nombres de personas son de tipo algebraico, probabilístico o híbridos que combinan una representación mediante grafos con el modelo algebraico. A continuación, se describe más detalladamente cada uno de ellos. Finalmente, se resume brevemente cuáles han sido de más utilidad para el problema.

Modelos algebraicos

La mayoría de los sistemas más competitivos de desambiguación de nombres de personas (ej. [Ikeda et al., 2009; Long y Shi, 2010; Liu et al., 2011; Chen et al., 2012]) han utilizado el modelo VSM para representar los documentos mediante un vector de palabras, asignando a cada una de ellas un valor que refleja su importancia obtenido mediante una función de ponderación o de pesado de términos. En particular, la función

de este tipo más utilizada es TF-IDF (*Term Frequency - Inverse Document Frequency*) [Salton y Buckley, 1988], que tiene en cuenta tanto la frecuencia de cada rasgo en un documento, como su aparición en el resto de documentos de la colección. Algunos autores [Romano et al., 2009; Chen et al., 2012] obtienen la frecuencia de las palabras a través del *corpus Web 1T 5-gram*[7] de Google, que contiene las frecuencias de *n*-gramas de longitudes entre 1 y 5 extraídos de páginas web escritas en inglés. Además, es habitual normalizar las frecuencias de las palabras con el fin de evitar predisposiciones a documentos largos que suelen repetir palabras en muchas ocasiones [Manning et al., 2008]. En este sentido, Monz y Weerkamp [2009] muestran que el uso de frecuencias normalizadas tiene un efecto más positivo en los resultados con respecto a utilizar frecuencias absolutas.

Algunos autores obtienen la importancia de cada palabra mediante otro tipo de funciones de ponderación. Por ejemplo, Martínez-Romo y Araujo [2009] emplean la *Divergencia de Kullback-Leibler* (*Kullback-Leibler Divergence*, KLD) para pesar la importancia de un término con respecto a un documento, mientras que Mann y Yarowsky [2003] emplean *Información Mutua* (*Mutual Information*, MI), que obtiene la importancia de un término con respecto a toda la colección de documentos.

Por otra parte, Kozareva et al. [2007] emplean LSA para obtener la similitud semántica entre los resultados de búsqueda, pero obtienen resultados muy pobres. Según los autores, esto se debe a que las páginas web contienen un número variable de palabras, por lo que proponen como trabajo futuro fijar un tamaño de ventana similar para todas ellas.

Modelos probabilísticos

Los modelos probabilísticos han sido empleados por un número reducido de autores y normalmente han obtenido pobres resultados. En particular, los métodos más empleados son pLSA y LDA, basados en la extracción de los tópicos de los que hablan los documentos. No obstante, varios trabajos han evidenciado que estas técnicas no parecen adecuadas para representar las páginas web en este problema.

Balog et al. [2007] muestran que los resultados obtenidos mediante una representación basada en VSM obtiene mejoras significativas con respecto a emplear los tópicos extraídos por pLSA, de modo que concluyen que una representación sencilla como VSM es suficiente para obtener resultados competitivos en el problema.

Por otro lado, Ono et al. [2008] usan un modelo basado en procesos de Dirichlet propuesto por Ferguson [1973] y Kozareva y Ravi [2011] emplean LDA, y ambos sistemas presentan muy bajo rendimiento. En el caso de LDA, esto puede explicarse porque requiere conocer de antemano el número de tópicos, el cual se corresponde con el número de individuos diferentes mencionados en los resultados de búsqueda de acuerdo con el

[7] `https://catalog.ldc.upenn.edu/ldc2006t13`

modelo propuesto por Kozareva y Ravi [2011]. En particular, estos autores prefijan este número en 40, de modo que obtienen resultados particularmente bajos para nombres de persona poco ambiguos donde hay un número reducido de individuos distintos.

Más recientemente, Xu et al. [2015] han propuesto un modelo probabilístico competitivo basado en el modelo jerárquico de correferencias propuesto por Wick et al. [2012]. Este método se diferencia del resto porque se basa en generar una estructura jerárquica de entidades mediante la comparación de las menciones de los nombres de persona en los documentos a partir de la coaparición de una rica selección de rasgos. El conjunto de rasgos empleados por este método es mucho mayor con respecto a los utilizados por el resto de sistemas basados en métodos probabilísticos. Esto puede significar que los bajos resultados obtenidos por los sistemas mencionados anteriormente sean debidos, en parte, a una representación de los resultados de búsqueda mediante pocos tipos de rasgos.

Modelos híbridos basados en grafos

Algunos sistemas competitivos [Bekkerman y McCallum, 2005; Jiang et al., 2009; Smirnova et al., 2010; Nuray-Turan et al., 2012] emplean un modelo híbrido que combina grafos y VSM para representar los resultados de búsqueda. Estos sistemas se caracterizan porque se componen de dos fases. Durante la primera fase, estos sistemas representan los resultados de búsqueda mediante un grafo y tienen como objetivo obtener *clusters* iniciales cohesivos. En cambio, durante la segunda fase, estos sistemas emplean el modelo VSM y tienen como objetivo obtener los *clusters* finales a partir de la fusión de los *clusters* iniciales.

Podemos distinguir estos sistemas atendiendo a los rasgos que utilizan en la construcción de los grafos. Mientras que algunos sistemas [Jiang et al., 2009; Nuray-Turan et al., 2012] usan el contenido de las páginas web para crear los nodos y etiquetar las aristas, otros sistemas [Iria et al., 2007; Han y Zhao, 2010] construyen el grafo mediante recursos externos. En particular, Iria et al. [2007] utiliza el operador *related:URL* de Google que devuelve páginas relacionadas con la página web de la que se da la URL, y posteriormente aplican el algoritmo *Random Walks* para determinar la similitud de las páginas web. Por su parte, Han y Zhao [2010] proponen una representación de *grafo semántico* cuyos nodos son conceptos extraídos de los documentos y las aristas son relaciones semánticas entre ellos obtenidas de recursos externos como Wikipedia o Wordnet.

Conclusión

Podemos concluir que no se requiere una representación muy costosa para poder obtener resultados prometedores en desambiguación de nombres de persona. En particular, los mejores sistemas de desambiguación han empleado el modelo VSM pesando las palabras mediante la función de pesado de términos TF-IDF. Por otro lado, los mo-

delos basados en extracción de *tópicos* tienden a agrupar indebidamente las páginas web y obtienen resultados muy pobres. Finalmente, las representaciones mediante grafos normalmente se emplean como paso previo para obtener *clusters* iniciales con una alta precisión, pero no sirven por sí mismos para obtener resultados competitivos y requieren una representación adicional mediante VSM.

2.3. Algoritmos de *clustering*

Los algoritmos de *clustering* o agrupamiento son métodos no supervisados [Hofmann, 1999; Aggarwal y Zhai, 2012] que dividen un conjunto de objetos en grupos denominados *clusters* de manera que (i) los objetos pertenecientes al mismo *cluster* son lo más similares posibles entre sí de acuerdo con algún criterio determinado; y (ii) los objetos pertenecientes a distintos *clusters* tienen el menor grado de similitud posible entre sí. Estos métodos se han aplicado en áreas diversas tales como la biología [Nugent y Meila, 2010], la medicina [Nithya et al., 2013] o el marketing [Vidden et al., 2016].

En particular, el *Clustering de Documentos* (*Document Clustering*, DC) consiste en la aplicación de estas técnicas en colecciones compuestas por documentos de texto. Habitualmente, el DC se ha empleado en tareas de IR consistentes en la organización y clasificación de grandes volúmenes de documentos [Gaussier et al., 2002] bajo la *hipótesis de agrupamiento* (*cluster hypothesis*) formulada por van Rijsbergen [1979], que establece que *los documentos fuertemente asociados tienden a ser relevantes para una misma consulta*. De este modo, se asume que la agrupación de documentos debe realizarse en base a su contenido y que los documentos similares entre sí serán aquellos que contengan rasgos similares. El criterio que determina la similitud entre documentos habitualmente viene dado mediante alguna función de distancia o alguna medida de similitud que permite determinar la proximidad entre ellos. Por ejemplo, cuando los documentos se representan mediante VSM, es habitual calcular su proximidad mediante la distancia euclídea o el coseno del ángulo formado por los vectores.

El DC es especialmente útil en el dominio de búsqueda en la Web porque, además de organizar la información en base a algún criterio, facilita la presentación de los resultados a los usuarios [Cigarrán, 2008]. Dado que el problema WePS encaja con estas características, se ha planteado desde el principio como un problema de *clustering* [Artiles, 2009], de modo que se han empleado este tipo de algoritmos para distinguir los resultados de búsqueda de acuerdo con el individuo al que hacen referencia.

El problema de desambiguación de nombres de persona cuenta con dos características que afectan a la política de agrupamiento que debe llevarse a cabo:

- La estimación del número de individuos diferentes es uno de los retos a resolver,

puesto que no es posible conocer ese dato de antemano. Por esa razón, lo más lógico es utilizar un algoritmo de *clustering* que determine automáticamente el número de individuos distintos (*clusters*) que se mencionan en los documentos [Artiles, 2009].

- Dado que un mismo documento puede hablar de varios individuos distintos que comparten el mismo nombre, pueden generarse *clusters* solapados (*soft clustering*). Por ejemplo, basta pensar en cualquier página web que hable de los ex-presidentes de EEUU *George H.W. Bush* y *George W. Bush*, como sucede con las entradas de Wikipedia de ambos. No obstante, la mayoría de los métodos propuestos en el estado del arte no admiten solapamientos entre los *clusters* (*hard clustering*), salvo excepciones como el método propuesto por Ikeda et al. [2009]. Esto se debe a que normalmente se asume la hipótesis de que cada página web habla de un único individuo llamado como el nombre introducido por consulta (*one person per document*, [Bagga y Baldwin, 1998]). Esta asunción consiste en la adaptación a la desambiguación de nombres de personas de la hipótesis de *un sentido por discurso* (*one sense per discourse*) propuesta por Gale et al. [1992] para el problema WSD, y que sostiene que las ocurrencias de una misma palabra en un discurso (o documento) denotan casi siempre el mismo sentido. De acuerdo con Artiles [2009], resulta razonable suponer esta asunción, puesto que hay un número muy pequeño de páginas web que hablan de varios individuos con el mismo nombre y su presencia no tiene impacto en los resultados.

Los siguientes apartados repasan los algoritmos de *clustering* más utilizados por los sistemas de desambiguación de nombres de persona. Como se explicó en el apartado 2.2.3, los métodos pLSA y LDA constituyen por sí mismos técnicas de agrupamiento de documentos. Por tanto, con el fin de evitar repetir información, el siguiente repaso no los tiene en consideración.

2.3.1. Agrupamiento Jerárquico Aglomerativo

El Agrupamiento Jerárquico Aglomerativo (*Hierchichal Agglomerative Clustering*, HAC) ha sido el algoritmo más empleado por los sistemas de desambiguación de nombres de personas, tanto por los participantes en las campañas de evaluación WePS [Artiles et al., 2007, 2009b, 2010], como por trabajos posteriores [Han y Zhao, 2010; Yoshida et al., 2010; Liu et al., 2011; Berendsen, 2015].

El algoritmo se denomina *jerárquico* porque presenta las agrupaciones que lleva a cabo mediante una estructura jerárquica dividida en distintos niveles de especialización. En cambio, los algoritmos denominados *planos o de partición* se caracterizan porque los

clusters que generan no se presentan mediante una estructura que los vincule entre sí. Por otro lado, el algoritmo se denomina *aglomerativo* porque considera que inicialmente cada objeto conforma un *cluster* (*singleton cluster*), de manera que en cada iteración se van uniendo *clusters* hasta agrupar todos los objetos en un único *cluster* que los contenga. En cambio, los algoritmos denominados *divisivos* consideran que inicialmente todos los objetos están contenidos en un único *cluster*, de modo que en cada iteración se van dividiendo en grupos más pequeños hasta obtener tantos *clusters* como número de objetos recibidos como entrada. Los algoritmos aglomerativos suelen ser más empleados porque son menos costosos que los divisivos, puesto que estos últimos requieren tomar en consideración todas las posibles divisiones entre *clusters* en cada iteración. No obstante, de acuerdo con Manning et al. [2008], los algoritmos divisivos pueden dar lugar a jerarquías más precisas porque parten de información global de todos los objetos, mientras que los aglomerativos parten de información local de cada uno de ellos.

HAC recibe un conjunto de objetos, normalmente representados mediante VSM, y opera iterativamente agrupando en cada paso aquellos *clusters* que sean más similares entre sí. La salida generada por HAC se representa mediante una estructura de *dendrograma* que muestra qué *clusters* han sido agrupados en cada iteración junto con su grado de similitud. Los *clusters* resultantes se corresponden con un nivel del dendrograma y normalmente se obtienen de dos maneras distintas:

- Devolver el nivel del dendrograma que contenga un cierto número de *clusters*.
- Seleccionar un valor de *umbral de similitud* que corte el dendrograma de acuerdo al grado de similitud de cada iteración. Normalmente, este umbral se aprende previamente a partir de datos de entrenamiento o se prefija de manera manual.

En particular, los sistemas de desambiguación de nombres de personas escogen la segunda opción, habitualmente mediante el empleo de datos de entrenamiento. Esto se debe a que la selección de un umbral de similitud evita realizar una estimación a priori del número de *clusters*, correspondiente al número de diferentes individuos que se mencionan en los resultados de búsqueda. Solamente el sistema propuesto por Heyl y Neumann [2007] emplea HAC prefijando 12 *clusters*. Por su parte, Iria et al. [2007] estiman el número de *clusters* mediante el *criterio de Caliński-Harabasz* [Caliński y Harabasz, 1974]. No obstante, ambos sistemas obtienen un bajo rendimiento.

Los criterios de similitud entre *clusters* se denominan *políticas de enlace*. Todas ellas tienen en común que se basan en la comparación de *clusters* mediante alguna función de distancia o medida de similitud entre los vectores que representan a los documentos que contienen. No obstante, se distinguen entre sí porque generan *clusters* con distinto grado de *cohesión*, la cual mide la similitud entre los objetos pertenecientes a un mismo

cluster. Suponiendo que sim_D es una medida de similitud entre documentos y C_i y C_j son *clusters* de documentos, a continuación se pasan a explicar las políticas de enlace más utilizadas por HAC para calcular la similitud entre *clusters*, que se denotará mediante sim_C:

- **Enlace simple:** se caracteriza porque los *clusters* se agrupan o dividen a partir de la similitud de los dos objetos más proximos pertenecientes a distintos clusters. Formalmente:

$$sim_C(C_i, C_j) = \max_{D_i \in C_i, D_j \in C_j} \{sim_D(D_i, D_j)\} \tag{2.1}$$

  Al tomar como referencia los objetos más próximos entre distintos *clusters*, esta estrategia suele generar *clusters* poco cohesivos y tiene como consecuencia que genera un fenómeno conocido como encadenamiento (*chaining*), que normalmente da lugar a que los *clusters* finalmente generados estén compuestos por un alto número de objetos.

- **Enlace completo:** se caracteriza porque los *clusters* se agrupan o dividen a partir de la similitud de los dos objetos más lejanos pertenecientes a distintos clusters. Formalmente:

$$sim_C(C_i, C_j) = \min_{D_i \in C_i, D_j \in C_j} \{sim_D(D_i, D_j)\} \tag{2.2}$$

  Esta política genera *clusters* muy cohesivos pero, al tomar como referencia los objetos más alejados entre sí, es bastante sensible a la aparición de valores extremos (*outliers*).

- **Grupo promedio:** se caracteriza porque los *clusters* se agrupan o dividen a partir de la media aritmética de las similitudes de los documentos pertenecientes a distintos *clusters*. Formalmente:

$$sim_C(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{D_i \in C_i} \sum_{D_j \in C_j} sim_D(D_i, D_j) \tag{2.3}$$

  Este criterio suele ser ampliamente utilizado porque equilibra las dos políticas anteriores, de modo que es capaz de evitar tanto el fenómeno del encadenamiento como efectos perniciosos ocasionados por los *outliers*.

Algunos autores [Balog et al., 2009; Dornescu et al., 2010] han concluido que la política de enlace simple es la más adecuada para la desambiguación de nombres de persona,

tras comparar los resultados obtenidos por el resto de políticas. Esto se debe a que generalmente suele haber un individuo con mucha más presencia que el resto en la Web [Artiles, 2009], lo cual encaja con el efecto de encadenamiento provocado por esta política. No obstante, otros sistemas competitivos como el propuesto por Ikeda et al. [2009] han empleado una política de enlace promedio obteniendo resultados prometedores, aunque es mucho más sensible al valor del umbral que se emplea para cortar el dendrograma. Por otro lado, la mayoría de los sistemas de desambiguación han calculado los valores de similitud mediante el coseno del ángulo formado por los vectores que representan los documentos, pero no se ha realizado un estudio comparativo utilizando otras medidas de similitud.

El Algoritmo 2.1 muestra el pseudocódigo del esquema general de HAC. El algoritmo recibe como entrada un conjunto de objetos $\mathcal{O}$ y una medida de similitud entre *clusters* sim_C que define la política de enlace empleada, y devuelve como salida una lista ordenada $\mathcal{A}$ que contiene las agrupaciones de *clusters* realizadas en cada iteración junto con su valor de similitud. Inicialmente, la lista $\mathcal{A}$ es vacía (línea 1) y cada objeto se almacena en un *cluster* unitario diferente debido a la naturaleza aglomerativa del algoritmo (líneas 2-5). A continuación, se fusionan los *clusters* más similares entre sí de acuerdo con la política de enlace utilizada (líneas 12-15) y se añade dicha agrupación a la lista $\mathcal{A}$ por el final (línea 16). En cada iteración se deben recalcular los valores de similitud entre los distintos *clusters* (líneas 7-11). Este proceso finaliza cuando queda un único *cluster* (línea 6). Finalmente, se devuelve la lista de agrupaciones de *clusters* generada (línea 18).

Manning et al. [2008] muestran que el esquema general de HAC tiene una complejidad temporal en $\Theta(N^3)$, siendo N el número de documentos dados como entrada. No obstante, existen versiones del algoritmo más eficientes cuando se asumen condiciones especiales. En particular, los algoritmos SLINK [Sibson, 1973] y CLINK [Defays, 1977] aplican las políticas de enlace simple y completo respectivamente, y tienen una complejidad en $\Theta(N^2)$. Por otro lado, la política de grupo promedio tiene una complejidad en $\Theta(N^2 \log(N))$ [Manning et al., 2008], pero Day y Edelsbrunner [1984] presentan una implementación de esta política con complejidad $\Theta(N^2)$ cuando se prefija de antemano la dimensionalidad de los datos.

La elección de este algoritmo por parte de los sistemas de desambiguación se debe a su simplicidad y el hecho de que no requiera conocer de antemano el número de *clusters*. No obstante, la principal desventaja de HAC es que se trata de un método muy sensible con respecto a la elección del umbral de similitud que determina el agrupamiento generado finalmente. Artiles et al. [2009b] muestran que una elección óptima de este umbral para cada nombre de persona es capaz de conseguir mejoras significativas con respecto a emplear el mismo umbral obtenido mediante datos de entrenamiento para todos los nombres de persona, como hacen la mayoría de los sistemas de desambiguación que

Algoritmo 2.1 HAC($\mathcal{O}, sim_C$).

Entrada: Conjunto de objetos $\mathcal{O} = \{O_1, O_2, \ldots, O_N\}$, medida de similitud sim_C

Salida: Lista ordenada de agrupaciones $\mathcal{A}$.

1: $\mathcal{A} := []$
2: **para** $i = 1$ **to** N **hacer**
3: $C_i = \{O_i\}$
4: **fin para**
5: $\mathcal{C} = \bigcup_{i=1}^{N} C_i$
6: **mientras** $|\mathcal{C}| \neq 1$ **hacer**
7: **para** $i = 1$ **to** $|\mathcal{C}|$ **hacer**
8: **para** $j = 1$ **to** $|\mathcal{C}|$ **hacer**
9: $Sim[i][j] = sim_C(C_i, C_j)$
10: **fin para**
11: **fin para**
12: $< i, j >= arg\,\text{máx}_{i,j \leq N}\{Sim[i][j] | i \neq j\}$
13: $C_{ij} = C_i \cup C_j$
14: $\mathcal{C} = \mathcal{C} \setminus \{C_i, C_j\}$
15: $\mathcal{C} = \mathcal{C} \cup \{C_{ij}\}$
16: $\mathcal{A}$.añadir$(< C_i, C_j, Sim[i, j] >)$
17: **fin mientras**
18: **devolver** $\mathcal{A}$

utilizan HAC. Por su parte, Long y Shi [2010] y Xu et al. [2015] muestran que pequeñas variaciones del valor de umbral pueden provocar grandes diferencias en los resultados obtenidos por este algoritmo.

2.3.2. *Algoritmos de partición*

Algunos sistemas de desambiguación han empleado algoritmos de partición para agrupar los resultados de búsqueda. Estos algoritmos tienen la ventaja de ser muy eficientes, presentando habitualmente una complejidad en $\mathcal{O}(N)$, siendo N el número de objetos que reciben como entrada. No obstante, normalmente tienen algunos incovenientes tales como que son indeterministas o requieren información a priori.

Podemos clasificar los algoritmos de partición empleados por los sistemas del estado del arte según requieran conocer a priori el número de *clusters* en los que se deben agrupar los objetos o no:

- **Requieren conocer el número de clusters:** los algoritmos de partición de este tipo empleados son el método de las *k*-medias (*k-means*) introducido por Stein-

haus [1956] y popularizado por MacQueen [1967], y el método de los *k*-vecinos (*k-medoids*), presentado por Kaufman y Rousseeuw [1987].

El método de las *k*-medias ha sido empleado por [Kozareva et al., 2007; Rao et al., 2007; Lan et al., 2009]. En particular, Rao et al. [2007] muestran que los resultados de este algoritmo dependen por completo de la elección del número *k* de *clusters* en los que se particionan los resultados de búsqueda. Por su parte, Kozareva et al. [2007] fijan este número para todos los nombres de persona en base a datos de entrenamiento, mientras que Lan et al. [2009] lo prefijan a un valor arbitrario para cada nombre de persona.

Por otro lado, Lana-Serrano et al. [2010] utilizan el método de los *k*-vecinos estimando el valor de *k* mediante una función de coste que tiene en cuenta algunas propiedades de los *clusters* anotados para los datos de entrenamiento.

Todas estas propuestas presentan un bajo rendimiento en comparación con los sistemas que utilizan HAC, debido a que no existe una manera de estimar a priori la distribución de individuos que aparecen en los resultados de búsqueda [Artiles, 2009].

- **No requieren conocer el número de clusters:** los algoritmos de este tipo empleados son *Quality Threshold* (QT) [Heyer et al., 1999] y *Single Pass Clustering* (SPC) [Hill, 1968]. Ambos tienen en común con HAC que requieren el valor de un umbral de similitud para decidir cuando se agrupan los resultados de búsqueda.

  Romano et al. [2009] emplean una adaptación del algoritmo QT que obtiene peores resultados con respecto a los mejores sistemas que usan HAC, pero mejora significativamene los resultados obtenidos por los sistemas que requieren conocer el número de *clusters*.

  Por su parte, Balog et al. [2009] comparan los resultados de SPC con respecto a pLSA y HAC llegando a la conclusión de que SPC mejora los resultados de pLSA, pero obtiene peores resultados que HAC. Por otro lado, en los resultados de la segunda campaña de evaluación WePS ([Artiles et al., 2009b]) se puede ver que los resultados de SPC mostrados por Balog et al. [2009] son mejores con respecto al método de las *k*-medias empleado por Lan et al. [2009].

En resumen, podemos dividir los algoritmos de partición entre aquellos que requieren conocer el número de *clusters* a priori y aquellos que no necesitan esa información. Los algoritmos del primer tipo obtienen pobres resultados debido a que no aplican estimaciones adecuadas para obtener el número de *clusters*. Por su parte, los algoritmos del segundo tipo mejoran los resultados de los primeros, pero no logran alcanzar los resultados obtenidos por los mejores sistemas que emplean HAC.

2.3.3. Algoritmos basados en grafos

Los sistemas de desambiguación que representan los documentos mediante una estructura de grafos, emplean algoritmos sobre esta estructura a la hora de agrupar los resultados de búsqueda. No obstante, Manning et al. [2008] explica que estos métodos están relacionados con HAC porque este último puede verse como un algoritmo de obtención de componentes conexas del grafo resultante de identificar cada *cluster* como un nodo y pesar las aristas entre ellos mediante sus valores de similitud. En particular, los algoritmos de este tipo más empleados son *Random Walks* [Iria et al., 2007; Smirnova et al., 2010], y algoritmos de extracción de componentes conexas o cliques. En particular, Dornescu et al. [2010] emplean este último tipo de algoritmos y destacan que, pese a su simplicidad, mejoran significativamente los resultados de otros métodos sobre grafos más complejos basados en procesos estocásticos.

La aplicación de este tipo de algoritmos suele generar altos valores de precisión y bajos valores de cobertura. Por este motivo, algunos sistemas [Jiang et al., 2009; Smirnova et al., 2010; Nuray-Turan et al., 2012] operan en dos fases: en la primera fase se aplica un algoritmo basado en grafos para obtener *clusters* iniciales muy cohesivos, mientras que en la segunda fase se aplica otro algoritmo, habitualmente HAC, con el objetivo de fusionar los *clusters* iniciales y de esta manera mejorar los valores de cobertura. Esta estrategia de agrupamiento obtiene resultados muy competitivos que, en algunos casos, son mejores con respecto a los obtenidos por los sistemas de desambiguación que emplean solamente HAC.

2.3.4. *Fuzzy Ants*

Desde los años 90 del siglo XX se han presentado varios algoritmos de optimización denominados *algoritmos de la colonia de hormigas* basados en varios patrones de comportamiento auto-organizativo que se han observado en estos insectos y que, en particular, han sido adaptados al *clustering*. Lefever et al. [2009] y Venkateshan [2009] han empleado un algoritmo de este tipo, denominado *Fuzzy Ants* [Schockaert et al., 2007], para distinguir a los individuos mencionados en un ranking de resultados de búsqueda. *Fuzzy Ants* se trata de una adaptación del algoritmo de Monmarché [2000] basada en el empleo de reglas *fuzzy* de la forma *IF-THEN* y ha sido empleado en tareas de *clustering* de páginas web. Las principales ventajas del algoritmo *Fuzzy Ants* son las siguientes:

- No requiere conocer el número de *clusters*.
- Calcula automáticamente un valor de umbral de similitud que decida cuando deben realizarse agrupaciones.

- Puede adaptarse al *soft clustering*, de modo que es capaz de devolver *clusters* solapados.

A pesar de ello, Lefever et al. [2009] concluyen que HAC mejora significativamente los resultados obtenidos por *Fuzzy Ants* y propone utilizar ambos métodos de manera combinada como trabajo futuro.

2.4. Tratamiento de las redes sociales

Debido a la popularización de las redes sociales en los últimos años, es frecuente que este tipo de páginas web aparezcan como resultados de búsqueda cuando se realiza una consulta en un buscador. Esta situación es especialmente común cuando se consultan nombres de persona, dado que muchos usuarios de estas plataformas las emplean para compartir información personal y laboral. Pese a ello, el rol de estos portales web no ha sido tenido en cuenta en la desambiguación de nombres de personas hasta 2012, ya que o bien las primeras colecciones de evaluación fueron recopiladas cuando el impacto de estas páginas web era mucho menor, o bien porque los organizadores las descartaron por ser especialmente ambiguas [Artiles et al., 2009b]. En lo sucesivo, denominaremos *página social* a una página web perteneciente a alguna red social, mientras que llamaremos *página no social* a cualquier otro tipo de resultado de búsqueda.

En un trabajo posterior a las campañas WePS, Berendsen et al. [2012] presentaron una colección de desambiguación compuesta por nombres de persona de origen neerlandés junto con sus resultados de búsqueda escritos en ese idioma y obtenidos mediante varios buscadores (Google, Yahoo! y Bing). Entre esos resultados de búsqueda incluyeron intencionadamente un amplio número de páginas sociales con el objetivo de analizar su impacto dentro del problema. Para ello, los autores aplicaron HAC representando los resultados de búsqueda como vectores pesados mediante TF-IDF por tratarse de una estrategia común de los mejores sistemas de desambiguación de nombres de persona. Posteriormente, aplicaron el algoritmo sobre páginas sociales y no sociales por separado, de modo que observaron que los resultados sobre páginas sociales eran al menos un 45 % peores. De este modo, concluyeron que la aparición de los perfiles de redes sociales pueden implicar que el rendimiento de muchos de los sistemas de desambiguación propuestos se vea afectado negativamente y, por tanto, las páginas sociales deben tratarse de manera diferenciada.

Además, Berendsen et al. [2012] proponen una estrategia dual para tratar el problema teniendo en cuenta las redes sociales. En primer lugar dividen las páginas web entre páginas sociales y páginas no sociales comprobando si su dominio se corresponde con alguna de las redes sociales. Posteriormente, desambiguan cada grupo de páginas web

por separado. En el caso de las páginas no sociales aplican HAC representando cada una de ellas mediante un vector de palabras pesadas con TF-IDF. En el caso de las páginas sociales exploran diferentes estrategias, llegando a la conclusión de que puede suponerse que las páginas sociales suelen corresponderse con individuos distintos, por lo que se obtienen mejores resultados cuando no se las agrupa entre sí. Finalmente, proponen dos métodos para mezclar los *clusters* de ambos tipos de páginas web:

- Devolver la unión de los *clusters* de ambos tipos de páginas: si $\mathcal{C}_s$ y $\mathcal{C}_{ns}$ son los *clusters* sociales y no sociales respectivamente, este método devuelve $\mathcal{C} = \mathcal{C}_s \cup \mathcal{C}_{ns}$. Dado que los *clusters* sociales no se agrupan entre sí, este método de mezcla devuelve cada página social dentro de un *cluster* unitario.
- Aplicar un algoritmo iterativo de mezcla de los *clusters* de ambos tipos de páginas, a los que se denominan *clusters* sociales y *clusters* no sociales. El algoritmo asume que los *clusters* sociales son aquellos que contienen solamente páginas sociales, mientras que los *clusters* no sociales son aquellos que contienen alguna página no social. En cada iteración, el algoritmo de mezcla opera de la siguiente manera:
  - Para cada *cluster* social, se calcula el *cluster* no social más similar mediante la comparación de ambos *clusters* con respecto a un umbral prefijado manualmente. Tras este paso, cada *cluster* no social tiene asociada una lista de *clusters sociales candidatos* a ser mezclados entre sí.
  - Para cada *cluster* no social, se calcula su *cluster social candidato* que tenga mayor similitud y se agrupan ambos entre sí.

  El algoritmo penaliza a los *clusters* no sociales que contengan alguna página social debido a alguna fusión efectuada en iteraciones anteriores. Dicha penalización consiste en dividir el valor de similitud de estos *clusters* y cualquier *cluster* social con respecto al número de páginas sociales que contenga el *cluster* no social y un *parámetro de penalización* prefijado manualmente. Finalmente, el algoritmo termina cuando no existe ningún *cluster* social que sea suficientemente similar con respecto a algún *cluster* no social.

Aunque la hipótesis de que las páginas sociales suelen corresponderse a individuos diferentes parece ser razonable, dada su sencillez tiene algunas limitaciones. La más evidente es que su aplicación asume que cada individuo solo puede tener un perfil en una única red social, lo cual no tiene por qué ser cierto. Por tanto, dado que estas páginas web aparecen habitualmente como resultados de búsqueda, se deben refinar aún más las estrategias para tratarlas de modo que se correspondan más con un escenario real.

No se han encontrado más trabajos en el estado del arte que traten de forma diferenciada las páginas sociales.

2.5. Tratamiento del multilingüismo

La mayoría de los sistemas de desambiguación de nombres de persona en la Web han empleado colecciones formadas por páginas web escritas en el mismo idioma. Por ejemplo, las colecciones proporcionadas por las campañas WePS contienen páginas web escritas en inglés, mientras que la colección presentada por Berendsen et al. [2012] contiene páginas web escritas en neerlandés. No obstante, los motores de búsqueda son capaces de devolver enlaces a páginas web escritas en diferentes idiomas, de modo que debe estudiarse el papel del multilingüismo para poder desarrollar un sistema de desambiguación que sea útil en un escenario real. Además, de acuerdo con Pimienta et al. [2009], en los últimos años ha aumentado considerablemente el número de páginas web escritas en distintos idiomas debido a la popularización de Internet en países de habla no inglesa.

Algunos autores Mann y Yarowsky [2003]; Chen y Martin [2007b]; Kozareva y Ravi [2011] han empleado colecciones que contienen páginas web escritas en distintos idiomas. Pese a ello, todas estas colecciones tienen en común que asumen un escenario monolingüe, de modo que todas las páginas web asociadas a un nombre de persona están escritas en el mismo idioma.

Mann y Yarowsky [2003] proponen desambiguar a los individuos a partir de información personal como, por ejemplo, las fechas y lugares de nacimiento. Para ello, proponen extraer cada uno de estos datos a partir de patrones obtenidos mediante entrenamiento usando textos escritos en distintos idiomas. De este modo, para un cierto dato biográfico y un cierto idioma, manejan una lista de patrones. El problema de esta metodología reside en que se necesita una amplia colección de datos de entrenamiento para obtener los patrones: en particular, una colección independiente para cada dato biográfico e idioma. Los autores emplean una colección formada por cuatro nombres de personas y sus páginas web asociadas, pero no indican en qué idiomas están escritas.

Por su parte, Chen y Martin [2007b] recopilaron una pequeña colección denominada *Boulder Name Corpus* compuesta por cuatro nombres de origen anglosajón y otros cuatro de origen chino, donde todas las páginas web asociadas a cada individuo están escritas en su idioma de origen. No obstante, este trabajo se centra en el estudio de distintos tipos de rasgos para representar los documentos, de modo que no proponen ninguna técnica para tratar el multilingüismo en el problema.

Finalmente, Kozareva y Ravi [2011] emplean una colección de datos formada por documentos que hablan de distintos nombres de personas, localidades y organizaciones. Cada una de las entidades tiene asociados varios documentos escritos en uno de los siguientes idiomas: inglés, castellano, rumano y búlgaro. Los autores se centran en estudiar la aplicación de LDA en la desambiguación de entidades, de modo que no ex-

ploran técnicas de tratamiento del multilingüismo. No obstante, concluyen que LDA es una técnica válida para desambiguar cualquier tipo de entidad en documentos escritos en cualquier idioma.

2.6. Clasificación de los sistemas de desambiguación de nombres de persona

Con el fin de analizar mejor los sistemas de desambiguación de nombres de persona del estado del arte, podemos clasificarlos de acuerdo con los siguientes aspectos:

- Los rasgos empleados para representar los resultados de búsqueda.
- El modelo de representación empleado.
- El algoritmo de *clustering* usado para agrupar los resultados de búsqueda.
- Requerimiento de datos de entrenamiento.
- Tratamiento de las redes sociales.
- Tratamiento del multilingüismo.

La Tabla 2.1 muestra estas características para los mejores sistemas de desambiguación de nombres de persona de la siguiente manera:

- La columna *Rasgos* indica los tipos de rasgos empleados por cada sistema de desambiguación empleando la siguiente nomenclatura:
  - Los rasgos lingüísticos se denotan con *L*.
  - Los rasgos no lingüísticos se denotan con *NL*.
  - Los rasgos web se denotan con *W*.
  - Los rasgos de información personal se denotan con *IP*.
  - Los rasgos externos se denotan con *E*.
- La columna *Modelo* indica el tipo de modelo de representación de documentos empleado por el sistema de desambiguación. La mayoría de los sistemas emplean el modelo VSM, mientras que otros sistemas emplean un modelo híbrido que combinan una representación mediante grafos y VSM (marcados como *G+VSM*), y solamente el sistema propuesta por Xu et al. [2015] es probabilístico (marcado como *Prob.*).

- La columna *AC* indica el algoritmo de *clustering* empleado por los sistemas de desambiguación. La mayoría de los sistemas emplean HAC o variaciones de este algoritmo. Aquellos sistemas que no indican qué política de enlace emplean se marcan como *HAC*. Si el sistema emplea política de enlace simple se marca con *HAC-S*, mientras que si emplea política de enlace promedio se marca como *HAC-P*. Otros sistemas se basan en algoritmos que extraen componentes conexas sobre grafos (marcados como *CC*) y pueden funcionar en dos fases junto con HAC (marcado como *CC+HAC*), como es el caso del sistema propuesto por Jiang et al. [2009]. Finalmente Xu et al. [2015] emplea un algoritmo jerárquico de correferencias probabilístico denominado $HIER_{coref}$, propuesto por Wick et al. [2012].
- La columna *Ent.* indica si el sistema de desambiguación requiere datos de entrenamiento o no.
- La columna *Social* indica si el sistema de desambiguación realiza un tratamiento especial sobre las redes sociales.
- La columna *ML* indica si el sistema de desambiguación trata el multilingüismo.

Sistema	Rasgos	Modelo	AC	Ent.	Social	ML
Chen y Martin [2007a]	L+NL+W+IP	VSM	HAC-S	SI	NO	NO
Popescu y Magnini [2007]	L+NL	VSM	HAC	SI	NO	NO
Balog et al. [2009]	NL	VSM	HAC-S	SI	NO	NO
Jiang et al. [2009]	L+IP	G+VSM	CC + HAC-S	SI	NO	NO
Long y Shi [2010]	L+NL+E	VSM	HAC	SI	NO	NO
Smirnova et al. [2010]	NL+W+E	G+VSM	CC + HAC-S	SI	NO	NO
Yoshida et al. [2010]	NL+W+E	VSM	HAC-P	SI	NO	NO
Liu et al. [2011]	NL+W	VSM	HAC	SI	NO	NO
Nuray-Turan et al. [2012]	L+NL+W+IP+E	G+VSM	CC + HAC	SI	NO	NO
Berendsen [2015]	NL	VSM	HAC-S	SI	SI	NO
Xu et al. [2015]	L+NL+W+IP+E	Prob.	$HIER_{coref}$	NO	NO	NO

Tabla 2.1: Clasificación de los mejores sistemas de desambiguación de nombres de persona del estado del arte.

La tabla resume el contenido de lo que se ha ido viendo en este capítulo de la tesis:

- Los rasgos más empleados son los lingüísticos (L) y los no lingüísticos (NL), siendo estos últimos los más utilizados por los métodos de desambiguación más competitivos. En particular, Balog et al. [2009] únicamente emplea rasgos NL y sus resultados son comparables a los de otros métodos que utilizan una representación más compleja, como el propuesto por Xu et al. [2015]. Los rasgos web también

son ampliamente utilizados, destacando el empleo de *links* para determinar qué páginas webs están enlazadas entre sí. Por último, el empleo de rasgos externos es más común en los sistemas de desambiguación más actuales. Los sistemas han empleado cada tipo de rasgo de distinto modo a la hora de comparar los resultados de búsqueda entre sí:

- Los rasgos lingüísticos más utilizados son las NEs. En general, la mayoría de los sistemas las emplean para comparar los resultados de búsqueda mediante medidas de similitud sobre los vectores que las contienen y representan en cada documento. No obstante, algunos sistemas realizan un tratamiento especial sobre ellas. En particular, Yoshida et al. [2010] comparan los documentos teniendo en cuenta su similitud para cada categoría de NE, asignando mayor peso a cada una de ellas por este orden: nombres de persona, organizaciones y lugares. Por su parte, Nuray-Turan et al. [2012] evitan usar determinadas NEs incluidas en una lista, debido a que son muy ambiguas y pueden dar lugar a agrupaciones incorrectas. Este es el caso del nombre de países o grandes ciudades.
- Los rasgos no lingüísticos más empleados son las palabras. Todos los sistemas las emplean para comparar los documentos a partir de medidas de similitud. Algunos autores [Chen y Martin, 2007a; Yoshida et al., 2010] asignan un mayor peso a aquellas palabras que se encuentran más cercanas al nombre de persona consultado, al asumir que tienen una mayor relación con cada uno de los individuos.
- Los rasgos web más empleados son los *links*. Los sistemas de desambiguación agrupan las páginas web enlazadas entre sí bajo la asunción de que esto indica que se refieren al mismo individuo.
- Los rasgos correspondientes a información personal son empleados de una manera similar a los rasgos web, de modo que los sistemas de desambiguación agrupan páginas web que se refieran a individuos que hayan nacido el mismo año, o tengan la misma profesión. Por su parte, Nuray-Turan et al. [2012] también emplean estos rasgos como pistas para decidir cuando dos páginas web no hablan del mismo individuo. Por ejemplo, su sistema de desambiguación evita agrupar los resultados de búsqueda que hablan de individuos con distinto *middle name*.
- Los rasgos externos se emplean para enriquecer la representación de los documentos. En particular, los rasgos de este tipo empleados por los mejores sistemas consisten en información extraída de Wikipedia [Long y Shi, 2010; Xu et al., 2015] y rasgos extraídos de páginas web obtenidas mediante consultas adicionales [Smirnova et al., 2010; Nuray-Turan et al., 2012].

- VSM es el modelo de representación que emplean la gran mayoría de estos sistemas de desambiguación. La única excepción es el método probabilístico propuesto por Xu et al. [2015]. Algunos de los métodos más actuales [Jiang et al., 2009; Smirnova et al., 2010; Nuray-Turan et al., 2012] usan VSM junto con una representación adicional basada en grafos. En particular, los mejores sistemas suelen utilizar la función de pesado TF-IDF y la similitud coseno.

- El algoritmo de *clustering* más empleado es HAC. En particular, la política de enlace simple (HAC-S) es la más popular porque el efecto de encadenamiento que provoca se ajusta mejor a la distribución de nombres de personas en la Web. En el caso de Liu et al. [2011], se propone una política de comparación de *clusters* mediante *hit lists* compuestas por los rasgos compartidos por los documentos de cada *cluster*. Por su parte, Yoshida et al. [2010] proponen un método de dos fases basado en HAC, en donde la primera fase obtiene *clusters* muy precisos y la segunda fase los fusiona. Esta misma filosofía la emplean los métodos basados en grafos, también divididos habitualmente en dos fases. En la primera fase aplican un algoritmo de extracción de componentes conexas (CC) y en la segunda etapa emplean HAC. En el caso de Nuray-Turan et al. [2012], su etapa final de fusión de *clusters* solo se aplica sobre *clusters* unitarios. De nuevo, la excepción es el método propuesto por Xu et al. [2015], que aplica un algoritmo de resolución de correferencias $HIER_{coref}$.

- La mayoría de los sistemas requieren de algún tipo de dato de entrenamiento para estimar parámetros. En particular, todos estos métodos requieren, como mínimo, aprender por entrenamiento el valor de umbral que necesita HAC para devolver el conjunto de *clusters* finales. En este sentido, nuevamente el sistema propuesto por Xu et al. [2015] es una excepción.

- El sistema propuesto por Berendsen [2015] es el único método que realiza un tratamiento especial sobre las páginas sociales. Esto se debe a que las colecciones empleadas por el resto de sistemas del estado del arte contienen un número muy pequeño de páginas web de este tipo.

- Ninguno de los mejores sistemas de desambiguación encontrados trata el multilingüismo, porque han sido evaluados sobre colecciones donde se asume que todos los documentos asociados a un nombre de persona están escritos en el mismo idioma.

2.7. Conclusiones

Podemos resumir el contenido de este capítulo mediante los siguientes puntos:

- La desambiguación de nombres de persona en la Web se ha planteado como un problema de *clustering* compuesto por dos fases principales. La primera etapa tiene como objetivo representar las páginas web mediante rasgos que sean de utilidad a la hora de poder distinguir a diferentes individuos que comparten el mismo nombre. La segunda etapa tiene como objetivo agrupar los resultados de búsqueda de acuerdo al individuo al que se refieren mediante un algoritmo de *clustering*.

- Los sistemas de desambiguación aplican inicialmente una etapa de preprocesamiento cuyo objetivo es extraer el texto plano de las páginas web, realizar procesamientos linguísticos que permitan seleccionar información apropiada que pueda ser de ayuda para representar los documentos e identificar y descartar información ruidosa. Las técnicas más habituales durante la etapa de preprocesamiento son el análisis léxico, la eliminación de *stop words* y palabras poco frecuentes y la *stemmización* o lematización de las palabras contenidas en los documentos.

- Tras finalizar la etapa de preprocesamiento, los sistemas de desambiguación llevan a cabo la extracción de rasgos. Para ello, se emplean diferentes recursos de acuerdo con el tipo de los rasgos seleccionados: los rasgos no lingüísticos se extraen mediante herramientas de *tokenización*, los rasgos lingüísticos se extraen mediante sistemas NER y herramientas de etiquetado gramatical, los rasgos web se extraen mediante *parsers HTML*, los rasgos de información personal se extraen mediante el uso de técnicas de AE, y los rasgos externos se extraen de recursos como WordNet y Wikipedia o páginas web obtenidas mediante consultas adicionales a motores de búsquedas.

- Los principales sistemas de desambiguación han representado las páginas web mediante una combinación de rasgos de naturaleza lingüística, normalmente NEs, junto con el contenido del documento extraído sin utilizar recursos lingüísticos, generalmente palabras y n-gramas. Esta selección de rasgos parece suficiente para poder obtener resultados prometedores. Se ha reportado que la coocurrencia entre distintas páginas web de rasgos como los *links*, las URLs o los datos personales de los individuos, generalmente implica que se refieran al mismo individuo y son útiles para generar *clusters* iniciales precisos. Por último, el enriquecimiento de la representación mediante rasgos extraídos de recursos externos, como Wikipedia o *queries* adicionales en motores de búsqueda, mejora significativamente los resultados, pero implica un mayor coste computacional que no es deseable en problemas que se deben resolver en tiempo real.

- VSM es el modelo de representación más común entre los sistemas de desambiguación más competitivos. La representación de las páginas web mediante grafos suele ser muy efectiva para obtener *clusters* iniciales muy precisos, pero requiere

de otra etapa posterior que refine los agrupamientos. Finalmente, los sistemas que representan los documentos mediante técnicas basadas en la extracción de tópicos tienen resultados muy pobres.

- El algoritmo HAC es una buena elección a la hora de agrupar los resultados de búsqueda y ha sido frecuentemente utilizado por los métodos de desambiguación que presentan mejor rendimiento, frente a otros algoritmos como SPC, *Quality Threshold*, *Fuzzy Ants*, *k-means* o *k-medoids*.

- La desambiguación de nombres de personas en la Web puede efectuarse de manera competitiva a partir de una simple representación de documentos y el algoritmo HAC. No obstante, HAC tiene el inconveniente de que requiere datos de entrenamiento para aprender un umbral de similitud que es decisivo a la hora de obtener los *clusters* resultantes. La mayoría de métodos que han tratado de estimar automáticamente el número de *clusters* obtienen resultados muy pobres, por lo que continua siendo un reto. Solamente Xu et al. [2015] propone un método competitivo que no requiere utilizar datos de entrenamiento.

- Los principales sistemas de desambiguación no han tenido en cuenta el auge actual de las redes sociales, puesto que han sido evaluados con colecciones recopiladas cuando estas plataformas aún no tenían un gran impacto en Internet. Hoy en día es extraño que no aparezca ninguna página web de este tipo cuando se consulta un nombre de persona en un buscador. Berendsen [2015] concluyó que estas páginas web pueden tener un impacto negativo en los sistemas de desambiguación y deben ser tratadas de manera diferenciada.

- Pese a que los motores de búsqueda son capaces de devolver enlaces a páginas web escritas en diferentes idiomas, por el momento no se ha planteado un escenario multilingüe en el problema, asumiendo que todos los resultados de búsqueda se escriben en el mismo idioma. El tratamiento del multilingüismo permite dar con soluciones que puedan ser utilizadas en un escenario de búsqueda real.

Tras el estudio del estado del arte, podemos identificar tres problemas que continúan abiertos en la desambiguación de nombres de personas en la Web y que serán abordados en esta tesis doctoral:

- En primer lugar, el requerimiento de datos de entrenamiento por parte de los mejores sistemas de desambiguación. Artiles [2009] destaca que los resultados son muy sensibles a los parámetros aprendidos por entrenamiento debido a las distintas características que puede tener cada nombre de persona. En particular, según sea el umbral utilizado por HAC, los resultados pueden estar sesgados a favor

de nombres muy ambiguos o poco ambiguos. Por este motivo, resulta de interés investigar nuevas propuestas que no tengan esta limitación.

- Por otro lado, el impacto de las redes sociales no ha sido profundamente estudiado pese a que estas páginas web han cobrado un importante protagonismo en los últimos años. Berendsen [2015] propone una heurística sencilla para tratarlas, pero no se ajusta a situaciones frecuentes como, por ejemplo, que una persona tenga varios perfiles en diferentes redes sociales. Por tanto, debe analizarse de manera más precisa su papel y se deben proponer métodos que se correspondan más al escenario actual.

- Por último, pese a que el contenido en la Web tiende a ser cada vez más multilingüe [Pimienta et al., 2009], ningún trabajo ha asumido este factor, de modo que es necesario investigar nuevas técnicas que lo tengan en cuenta para poder adaptarse a un contexto actual de búsqueda en la Red.

3
Marco de experimentación

"La informática es la ciencia de la abstracción - crear el modelo correcto para un problema y diseñar técnicas mecanizables apropiadas para resolverlo."

— Alfred Aho & Jeffrey Ullman —

En este capítulo se explica el marco de experimentación que será empleado en los sucesivos capítulos de la presente tesis. Por un lado, se detallan algunos elementos relativos a la representación de los documentos, incluyendo el preprocesamiento, el modelo de representación empleado y diferentes funciones de pesado de términos y medidas de similitud utilizadas para comparar los documentos. Posteriormente, se presentan dos algoritmos de clustering del estado del arte que se usarán para comparar los resultados obtenidos por las propuestas presentadas en este trabajo. A continuación, se describirán las colecciones de evaluación de desambiguación de nombres de personas en la Web empleadas, haciendo especial hincapié en destacar sus características y sus principales diferencias. Finalmente, se definirán las métricas de evaluación de los resultados y el test de significancia estadística empleados para comparar el rendimiento entre diferentes sistemas de desambiguación de nombres de personas.

3.1. Representación de los documentos

Este apartado describe algunos detalles relativos a la representación de los documentos adoptados por las propuestas presentadas en esta tesis doctoral. En particular, se describirá el preprocesamiento llevado a cabo sobre las páginas web devueltas por el motor de búsqueda al introducir un nombre de persona. Posteriormente, se explicará brevemente el modelo de espacio vectorial empleado para representar matemáticamente los documentos. A continuación, se describirán las funciones de pesado de términos para medir la importancia de cada rasgo. Finalmente, se definirán las medidas de similitud utilizadas para comparar diferentes documentos.

3.1.1. Preprocesamiento

El preprocesamiento de las páginas web se ha realizado mediante la aplicación de los siguientes pasos:

- **Extracción del texto plano:** se ha empleado el *parser* HTML de la librería *TiKa Apache*[1] para extraer el texto plano de los resultados de búsqueda. Además, esta herramienta permite extraer otro tipo de información de las páginas web, como por ejemplo, los metadatos o los hipervínculos.

- **División del texto en frases:** se divide cada texto en las oraciones que lo componen a partir de los separadores ortográficos: puntos, saltos de línea, etc.

- ***Stemming*:** se aplica un algoritmo de *stemming* a cada palabra, sin incluir signos de puntuación, pero manteniendo los acentos. En particular, se ha utilizado el algoritmo de Porter [Porter, 1980] empleando el código fuente compartido por el propio autor[2].

- **Normalización del texto:** durante esta etapa se sustituyen las letras acentuadas por sus equivalentes sin acentuar. Por otro lado, se eliminan los signos ortográficos como las comas, puntos y comas, guiones, etc.

- **Eliminación de palabras vacías:** se eliminan las palabras vacías puesto que no ofrecen información semántica relevante y tienen un bajo valor discriminativo dado que suelen aparecer en muchos documentos. Además, los términos de la consulta (nombre y apellido/s) se tratan como palabras vacías porque se asume que todos los resultados de búsqueda los incluyen.

Tras llevar a cabo el preprocesamiento anterior, se efectua la extracción de los rasgos de cada una de las frases. En los siguientes capítulos se detallará qué clases de rasgos se han considerado oportunos para representar el contenido de los documentos. Posteriormente, se eliminan aquellos rasgos que aparecen en un único documento. Su eliminación se justifica porque no tienen poder discriminante y su aparición dificulta la comparación entre diferentes documentos. Además, de acuerdo con Aggarwal y Zhai [2012], este tipo de rasgos suele corresponderse con información ruidosa en el dominio Web.

[1] `https://tika.apache.org/`

[2] `https://tartarus.org/martin/PorterStemmer/index.html`

3.1.2. Modelo de espacio vectorial (VSM)

El modelo de representación de documentos que se empleará en esta tesis es VSM, propuesto por Salton et al. [1975]. Como vimos en el capítulo anterior, la mayoría de los mejores sistemas del estado del arte utilizan este modelo. Por otro lado, dada su sencillez, resulta adecuado para emplearlo en tareas que deben resolverse en tiempo real como la que se trata en esta tesis.

La representación con VSM se realiza en base a los *términos* que contienen los documentos. Habitualmente, los términos suelen corresponderse con las palabras, pero pueden ser otro tipo de rasgos como por ejemplo los n-gramas, las frases, etc. Este modelo supone el *principio de independencia* por el cual los términos que aparecen en un determinado texto no tienen relación entre sí. Por otra parte, VSM no tiene en cuenta el orden de aparición de los términos en el texto, por lo que asume que la semántica de un texto se reduce a la suma de los significados de cada uno de los términos que contiene. Pese a que estas asunciones no son correctas, permiten simplificar la representación de un documento como un vector de rasgos.

En lo sucesivo, diremos que un *corpus* es una colección de documentos, denotada como $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$. Cada documento $D_i \in \mathcal{D}$ puede verse como una secuencia formada por los términos que contiene por orden de aparición $D_i = s_1^i s_2^i \ldots s_{n_i}^i$, donde $n_i \in \mathbb{N} \cup \{0\}$ indica el número de términos que contiene el documento. El vocabulario de un documento consiste en el conjunto de todos los términos que contiene: $Voc(D_i) = \bigcup_{j=1}^{n_i} \{s_j^i\}$. Nótese que durante la etapa de preprocesamiento, algunos de los *tokens* originales del documento han sido eliminados como, por ejemplo, las palabras vacías.

Dado que se ha tomado el criterio de eliminar aquellos términos que solamente aparecen en un único documento, se considera que el vocabulario del corpus contiene aquellos términos que aparecen en varios documentos y se puede definir de la siguiente manera:

$$Voc(\mathcal{D}) = \{s \in \bigcup_{i=1}^{N} Voc(D_i) | \exists D_k, D_l \in \mathcal{D} : k \neq l \wedge s \in Voc(D_k) \cap Voc(D_l)\} \tag{3.1}$$

siendo $V_\mathcal{D} = |Voc(\mathcal{D})|$ el *tamaño del vocabulario* del corpus. Por simplicidad, denotaremos como $Voc(\mathcal{D}) = \{s_1, s_2, \ldots, s_{V_\mathcal{D}}\}$ a los términos del vocabulario del corpus $\mathcal{D}$.

La representación mediante VSM consiste en representar cada documento $D_i \in \mathcal{C}$ como un vector perteneciente al espacio vectorial sobre el cuerpo de los números reales $\mathbb{R}$ de dimensión $V_\mathcal{D}$, esto es: $\vec{d}_i = (w_1^i, w_2^i, \ldots, w_{V_\mathcal{D}}^i)$, donde el j-ésimo término del vector se corresponde con el término $s_j \in Voc(\mathcal{D})$, y $w_j^i \in \mathbb{R}$ es el valor de importancia (o peso)

de dicho término en el documento D_i. Generalmente, $w_j^i = 0$ indica que el término s_j no aparece en el documento D_i, mientras que $w_j^i > 0$ indica que el término aparece en el documento. Finalmente, el corpus $\mathcal{D}$ se representa mediante una matriz denominada *matriz de rasgo-documento* de dimensión $N \times V_{\mathcal{D}}$, que contiene el peso de cada rasgo en cada uno de los documentos del corpus:

$$\mathcal{M}_{N\times V_{\mathcal{D}}}(\mathcal{D}) = \begin{pmatrix} w_1^1 & w_2^1 & \cdots & w_{V_{\mathcal{D}}}^1 \\ w_1^2 & w_2^2 & \cdots & w_{V_{\mathcal{D}}}^2 \\ \vdots & \vdots & \ddots & \vdots \\ w_1^N & w_2^N & \cdots & w_{V_{\mathcal{D}}}^N \end{pmatrix} \tag{3.2}$$

Los pesos asociados a cada término se obtienen mediante la aplicación de alguna función de pesado de términos. En el apartado 3.1.3 definiremos las que serán empleadas en esta tesis doctoral. Por otro lado, esta representación permite comparar documentos entre sí mediante medidas de similitud basadas en operaciones entre vectores. El apartado 3.1.4 presenta las medidas de similitud que emplearemos a lo largo de este trabajo.

3.1.3. Funciones de pesado de términos

La representación mediante VSM implica que dos documentos $D_i, D_j \in \mathcal{D}$ serán considerados distintos si $\vec{d}_i \neq \vec{d}_j$, lo cual sucede si $\exists k \in \{1, 2, \ldots, V_{\mathcal{D}}\} : w_k^i \neq w_k^j$. Por tanto, la distinción entre documentos está determinada por la manera en la que se asignen pesos a cada uno de los términos incluidos en el vocabulario.

Las funciones de pesado de términos asignan un valor de importancia a cada término del vocabulario del corpus en un cierto documento. Estas funciones pueden clasificarse como *locales* o *globales*. Las funciones locales asignan pesos teniendo en cuenta exclusivamente las características de cada documento, mientras que las funciones globales además toman en consideración las características de toda la colección de documentos.

Se han seleccionado cuatro funciones de pesado de términos empleadas habitualmente en problemas de *clustering* con el objetivo de analizar su impacto en los resultados. En particular, se han escogido funciones locales y globales, de modo que en los estudios presentados posteriormente se puede concluir qué tipo de funciones es más adecuado para pesar los rasgos. A continuación, se presentan las funciones utilizadas en esta tesis: por un lado, las funciones locales binaria y TF (*term frequency*), y, por otro lado, las funciones globales TF-IDF (*Term Frequency - Inverse Document Frequency*) y *z-score* (o puntuación estándar). Además, se justifica el empleo de cada una de ellas.

Pesado binario

La función de pesado binario indica la pertenencia de un término a un determinado documento, de modo que la representación de los documentos queda limitada al *conjunto de términos* que contiene. Se define como $Bin : Voc(\mathcal{D}) \times \mathcal{D} \rightarrow \{0,1\}$, donde:

$$Bin(s, D_i) = \begin{cases} 1 & si \quad s \in Voc(D_i) \\ \\ 0 & si \quad s \notin Voc(D_i) \end{cases} \tag{3.3}$$

Esta función se corresponde con la *función característica* del conjunto $Voc(D_i)$, de modo que dos documentos $D_i, D_j \in \mathcal{D}$ se representan de la misma manera cuando $Voc(D_i) = Voc(D_j)$. Se trata de la función de pesado empleada por los modelos de representación booleanos de IR.

El empleo de la función binaria se justifica por dos motivos: por un lado, se trata de la función de pesado más sencilla y, por otro lado, es útil para analizar si no es necesario asignar distintos valores de importancia a los términos empleados para representar los resultados de búsqueda.

Pesado TF

Esta asignación de pesos de términos nace de la asunción de Luhn [1957] que establece que la importancia de un término es propocional a su frecuencia de aparición en un determinado documento. De este modo, el pesado TF asigna como valor para cada término su *frecuencia absoluta* en un documento, i.e.: el número de veces que aparece en el documento. Al emplear esta función de pesado, cada documento puede verse como un *multiconjunto* (o *bolsa*) formado por los términos que contiene, de manera que la frecuencia absoluta se corresponde con la multiplicidad de cada elemento del multiconjunto.

El pesado TF se define como $TF : Voc(\mathcal{D}) \times \mathcal{D} \rightarrow \mathbb{N} \cup \{0\}$, donde:

$$TF(s, D_i) = |\{s_j^i \in D_i | s = s_j^i\}| \tag{3.4}$$

siendo $D_i = s_i^1 s_i^2 \ldots s_i^{n_i}$ la representación del documento como la secuencia de los términos que contiene por orden de aparición.

Habitualmente, se suelen tomar las *frecuencias relativas* en lugar de las absolutas para evitar predisposiciones hacia documentos largos frente a otros más cortos. De este modo, se asegura que el peso de un término en cualquier documento consiste en un valor perteneciente al intervalo $[0,1] \subset \mathbb{R}$. La frecuencia relativa puede definirse mediante la función $TF_{rel} : Voc(\mathcal{D}) \times \mathcal{D} \rightarrow [0,1]$, donde:

$$TF_{rel}(s, D_i) = \frac{TF(s, D_i)}{\sum_{s' \in Voc(\mathcal{D})} TF(s', D_i)} \tag{3.5}$$

Las frecuencias relativas son valores normalizados de las frecuencias absolutas de los términos con respecto al número total de apariciones de todos los términos del documento. Por ello, estas frecuencias nos indican en qué proporción aparece cada término en el documento al multiplicarlas por 100. Por esta razón, para cualquier documento $D_i \in \mathcal{D}$ se cumple que $\sum_{s \in Voc(D_i)} TF_{rel}(s, D_i) = 1$.

La justificación de emplear el pesado TF se debe a que se trata de una función de pesado local que sí asigna diferentes valores de importancia a cada uno de los términos, a diferencia de la función binaria. Además, ha sido empleada por algunos sistemas del estado del arte [Saggion, 2007; Smirnova et al., 2010] y, de acuerdo con Monz y Weerkamp [2009], se trata de una función de pesado adecuada para la desambiguación de nombres de personas.

Pesado TF-IDF

Jones [1972] estudió cómo pesar los términos siguiendo la asunción de Luhn [1957] sobre la frecuencia de los términos en los documentos, pero teniendo además en cuenta la especificidad de cada término dentro de un corpus de documentos. Siguiendo estas ideas, Salton y Buckley [1988] propusieron el esquema de pesado TF-IDF, donde se toma en cuenta tanto la frecuencia de cada término en el documento (TF) como su especificidad obtenida mediante la *frecuencia inversa de documento* (IDF) en el corpus.

La función IDF tiene el perfil $IDF : Voc(\mathcal{D}) \times \mathcal{P}(\mathcal{D}) \rightarrow \mathbb{R}$, donde $\mathcal{P}(\mathcal{D})$ es el *conjunto potencia* de $\mathcal{D}$ que incluye todos sus subconjuntos, es decir, $\mathcal{D}' \in \mathcal{P}(\mathcal{D}) \Leftrightarrow \mathcal{D}' \subseteq \mathcal{D}$. En particular, la función se aplica sobre toda la colección $\mathcal{D} \in \mathcal{P}(\mathcal{D})$ y viene definida como sigue:

$$IDF(s, \mathcal{D}) = \log\left(\frac{|\mathcal{D}|}{|\{D_i \in \mathcal{D} | s \in Voc(D_i)\}|}\right) \tag{3.6}$$

Puesto que $|\{D_i \in \mathcal{D} | s \in Voc(D_i)\}| \leq |\mathcal{D}|$, se tiene que $\frac{|\mathcal{D}|}{|\{D_i \in \mathcal{D} | s \in Voc(D_i)\}|} \geq 1$, lo cual asegura que no se obtienen valores negativos al tomar logaritmos. Esta función asigna mayor peso a aquellos términos que aparecen en un menor número de documentos bajo la asunción de que son términos con un mayor poder discriminante. Pese a que su aplicación ha sido vista como una heurística empírica, Hiemstra [2000] da una justificación estadística en la que IDF se entiende como la probabilidad condicional de que un término aparezca en un determinado documento de la colección, mientras que Aizawa [2003] y Robertson [2004] presentan una perspectiva de esta función desde el punto de vista de la teoría de la información de Shannon [Shannon, 1948].

Finalmente, la función de pesado TF-IDF tiene un perfil $TF - IDF : Voc(\mathcal{D}) \times \mathcal{D} \times \mathcal{P}(\mathcal{D}) \rightarrow \mathbb{R}$ y viene definida como sigue:

$$TF - IDF(s, D_i, \mathcal{D}) = TF(s, D_i)\dot{I}DF(s, \mathcal{D}) \tag{3.7}$$

Nótese que los pesos obtenidos mediante TF-IDF son valores no negativos porque también lo son tanto las frecuencias como las frecuencias inversas de documento. Como se comentó con anterioridad, es habitual tomar las frecuencias relativas de cada término $TF_{rel}(s, D_i)$ en vez de las absolutas. Según el esquema TF-IDF, los términos con mayor peso dentro de un documento serán aquellos que sean muy frecuentes dentro de ese documento, pero que aparecen en un número reducido de documentos pertenecientes a la colección. Por esta razón, esta función se ha empleado habitualmente para identificar tanto palabras vacías como palabras clave (*keywords*).

El empleo de la función TF-IDF se debe a que se trata de la función de pesado de términos empleada por la mayoría de los sistemas del estado del arte [Artiles, 2009].

Pesado *z-score*

Los valores *z-score*, también conocidos como puntuación estándar, son una medida estadística que permite medir la distancia de cada valor perteneciente a una cierta población con respecto a la media de la misma en términos de desviaciones estándar.

Por un lado, se requiere obtener la frecuencia media en la que aparece cada término en el documento, lo cuál viene dado por la función $\mu : Voc(\mathcal{D}) \times \mathcal{P}(\mathcal{D}) \rightarrow \mathbb{R}$, donde

$$\mu(s, \mathcal{D}) = \sum_{i=1}^{N} \frac{TF(s, D_i)}{|\mathcal{D}|} \tag{3.8}$$

Por otra parte, se calcula la desviación típica de cada palabra obtenida mediante la función $\sigma : Voc(\mathcal{D}) \times \mathcal{P}(\mathcal{D}) \rightarrow \mathbb{R}$, donde:

$$\sigma(s, \mathcal{D}) = \sum_{i=1}^{N} \frac{\sqrt{(TF(s, D_i) - \mu(s, \mathcal{D}))^2}}{|\mathcal{D}|} \tag{3.9}$$

Finalmente, el valor *z-score* asociado a cada término en un cierto documento viene determinado mediante la función $z : Voc(\mathcal{D}) \times \mathcal{D} \times \mathcal{P}(\mathcal{D}) \rightarrow \mathbb{R}$, donde:

$$z(s, D_i, \mathcal{D}) = \frac{TF(s, D_i) - \mu(s, \mathcal{D})}{\sigma(s, \mathcal{D})} \tag{3.10}$$

A diferencia de las funciones presentadas anteriormente, los pesos obtenidos mediante *z-score* pueden ser negativos. Esto puede ser problemático puesto que, como

veremos posteriormente, algunas medidas de similitud operan sobre valores no negativos. Para evitar esta situación, se realiza un desplazamiento de los valores de la siguiente manera: en primer lugar, se calcula el menor valor *zscore* de toda la colección $min_z(\mathcal{D}) = min\{z(s, D_i, \mathcal{D}) | D_i \in \mathcal{D} \wedge s \in Voc(D_i)\}$ y, posteriormente, se reasignan los valores *z-score* como $z(s, D_i, \mathcal{D}) := z(s, D_i, \mathcal{D}) + (min_z(\mathcal{D})) + 1$, de modo que se asegura que el menor valor *z-score* de la colección vale 1.

La justificación de emplear z-score se debe a dos motivos: por un lado, varios autores [Andrade y Valencia, 1998; Cummins, 2013] han concluido que se trata de una función adecuada para pesar los términos que aparecen en los documentos y, por otro lado, nos permite comparar la función TF-IDF con respecto a otra función de pesado de términos global.

3.1.4. Medidas de similitud

Las medidas de similitud son funciones matemáticas que, como su nombre indica, nos permiten cuantificar la similitud entre dos objetos. Habitualmente, suelen verse como inversiones de las *funciones de distancia*, de modo que la similitud se interpreta como una medida de la proximidad o cercanía entre objetos. En particular, los algoritmos de *clustering* emplean estas funciones para comparar los objetos, de manera que los *clusters* generados contengan objetos lo más similares posibles entre sí y lo más distintos posibles con respecto a los objetos contenidos en el resto de *clusters*. La definición formal de este concepto se encuentra en la Definición 3.1:

Definición 3.1. Una **medida de similitud** sobre un conjunto X es una función sim: $X \times X \rightarrow [0, 1]$ tal que cumple las siguientes propiedades:

- 1. $\forall x, x' \in X : sim(x, x') = 1 \Leftrightarrow x = x'$
- 2. $\forall x, x' \in X : sim(x, x') = sim(x', x)$

Se trata de funciones cuyo *codominio* es el intervalo $[0, 1] \subset \mathbb{R}$, por lo que los valores que devuelven están restringidos a ese rango. La primera propiedad establece que el máximo grado de similitud de un objeto lo tiene consigo mismo, de modo que tendrá un grado de similitud menor con cualquier otro objeto diferente, i.e. $\forall x, x' \in X : 0 \leq sim(x, x') \leq sim(x, x) = sim(x', x') = 1$. En segundo lugar, se establece que las medidas de similitud son funciones que cumplen la *propiedad simétrica*. Por otro lado, se habla de *medida de similitud métrica* cuando adicionalmente cumple la *propiedad triangular*: $\forall x, x', x'' \in X : sim(x, x'') \leq sim(x, x') + sim(x', x'')$. Por último, dada una medida de similitud sim, se puede definir una función de distancia (o *disimilitud*) como $dist : X \times X \rightarrow [0, 1]$, donde $dist(x, x') = 1 - sim(x, x')$.

Strehl et al. [2000] estudiaron qué medidas de similitud y distancia son más eficientes a la hora de aplicar técnicas de *clustering* en páginas web. Por un lado, llegaron a la conclusión de que las medidas de distancia más empleadas no son adecuadas en dominios dispersos como las páginas web. En concreto, comprobaron esta afirmación empleando la distancia de Minkowski y sus casos particulares, la distancia euclídea y la distancia de Manhattan. Por otro lado, también concluyeron que las medidas de similitud que obtienen mejores resultados son el coseno (o *separación angular*) y aquellas basadas en el *indice de Jaccard* [Jaccard, 1901], que calcula la proporción de términos comunes en dos documentos con respecto al número de total de términos distintos que aparecen en ambos. Posteriormente, Huang [2008] corroboró estas conclusiones sobre siete colecciones de documentos diferentes.

En base a las conclusiones de Strehl et al. [2000] y Huang [2008], se han seleccionado las medidas de similitud coseno y *Jaccard pesado* (*weighted Jaccard*). Además, la selección de la similitud coseno también se justifica porque ha sido empleada por la mayoría de los sistemas de desambiguación de nombres de persona. Por otro lado, se ha escogido la similitud Jaccard pesado porque tiene en cuenta el peso de los términos, a diferencia del índice de Jaccard, de modo que tiene sentido emplear las funciones de pesado de términos vistas anteriormente.

A continuación, definiremos las dos medidas de similitud que emplearemos a lo largo de esta tesis doctoral.

Similitud coseno

Esta medida de similitud se obtiene mediante el coseno formado por los vectores que representan los objetos que se quieren comparar. De este modo, la similitud se establece a partir de la orientación de ambos vectores, de forma que será máxima cuando estos sean paralelos y mínima en el caso de que sean ortogonales. La similitud coseno puede calcularse para espacios vectoriales normados que tengan definida una operación de producto escalar. En particular, esta condición se cumple para VSM, puesto que representa los documentos como vectores pertenecientes al espacio vectorial euclídeo $\mathbb{R}^p$ que garantiza la existencia de estas operaciones. Denotaremos y definiremos la medida de similitud coseno como $cos : \mathbb{R}^p \times \mathbb{R}^p \rightarrow [0,1]$ donde:

$$cos(X,Y) = \frac{X \cdot Y}{\|\mathbf{X}\|\|\mathbf{Y}\|} = \frac{\sum_{i=1}^{p} X_i Y_i}{\sqrt{\sum_{i=1}^{p} X_i^2}\sqrt{\sum_{i=1}^{p} Y_i^2}} \tag{3.11}$$

Como los pesos de cada término son valores no negativos, se garantiza que los valores de similitud obtenidos por la función coseno se encuentran en el intervalo $[0,1]$. Nótese que el coseno puede calcularse mediante el producto escalar de ambos vectores cuando estos han sido previamente normalizados, puesto que en dicho caso se tiene que

$\|\mathbf{X}\| = \|\mathbf{Y}\| = 1$. La presencia de la norma en el denominador garantiza que esta función de similitud es independiente con respecto a la longitud o tamaño de los documentos.

Similitud Jaccard pesado (*weighted Jaccard*)

La similitud Jaccard pesado se define como una función que recibe dos vectores con valores no negativos y tiene el perfil $WJ : \mathbb{R}^p \times \mathbb{R}^p \rightarrow [0, 1]$, donde:

$$WJ(X,Y) = \begin{cases} \sum_{i=1}^{p} \frac{min(X_i,Y_i)}{max(X_i,Y_i)} & si \quad X \neq \vec{0} \vee Y \neq \vec{0} \\ \\ 1 & si \qquad X = Y = \vec{0} \end{cases} \tag{3.12}$$

Esta función es una generalización del índice de Jaccard [Jaccard, 1901], que mide la similitud entre dos conjuntos A y B como $J(A,B) = \frac{|A \cap B|}{|A \cup B|}$. En particular, la similitud Jaccard pesado equivale al índice de Jaccard cuando los documentos se pesan mediante la función binaria.

3.2. Algoritmos de *clustering*

En este trabajo se han empleado dos algoritmos de *clustering* del estado del arte para verificar las hipótesis sobre la representación de los documentos que explicaremos en el próximo capítulo. En particular, se han escogido los algoritmos HAC y *Affinity Propagation* (AP) por los siguientes motivos:

- HAC es el algoritmo de *clustering* más empleado por parte de los mejores sistemas de desambiguación de nombres de persona. Este algoritmo requiere conocer el número de *clusters* o un umbral de similitud, normalmente obtenido mediante datos de entrenamiento, para poder generar los *clusters* resultantes. No obstante, los sistemas de desambiguación emplean la segunda opción con el fin de evitar una estimación a priori del número del número de individuos diferentes que son mencionados en los resultados de búsqueda.

- AP tiene en común con las propuestas presentadas en esta tesis doctoral que no necesita conocer ningún tipo de información a priori para generar los *clusters* resultantes.

A continuación, los apartados 3.2.1 y 3.2.2 describen con mayor detalle la configuración empleada por ambos algoritmos en los experimentos llevados a cabo en los siguientes capítulos.

3.2.1. HAC

La descripción del algoritmo HAC puede encontrarse en el apartado 2.3.1. Los experimentos llevados a cabo con HAC utilizan la política de enlace simple, dado que ha sido la más empleada por la mayoría de los sistemas de desambiguación de nombres de persona. Además, algunos autores [Balog et al., 2009; Dornescu et al., 2010] han concluido que esta política de enlace es la más adecuada para la desambiguación de nombres de personas en la Web.

Se ha escogido el criterio de seleccionar un umbral de similitud para generar los *clusters* resultantes de la misma forma que los sistemas del estado del arte. En particular, el umbral escogido se obtiene de la siguiente manera:

- Aplicar HAC sobre los rankings de resultados de búsqueda de todos los nombres de persona de la colección.
- Cortar los dendrogramas devueltos por HAC tomando los valores de umbral de similitud 0.00, 0.01, . . . , 0.98, 0.99, 1.00.
- Obtener los resultados obtenidos por cada valor de umbral para todos los nombres de persona. En particular, se ha empleado la métrica oficial de las campañas WePS, descrita en la sección 3.4.
- Seleccionar el valor de umbral que obtiene el mejor resultado promedio en la colección.

Los sistemas de desambiguación obtienen el valor de umbral de similitud mediante el procedimiento anterior sobre una colección de entrenamiento y, posteriormente, aplican dicho valor para todos los nombres de persona contenidos en una colección de test. En cambio, en los experimentos llevados a cabo en esta tesis se empleará el mejor valor de umbral sobre la propia colección que se evalúa, lo cual puede ser visto como una cota superior de la política de obtención del umbral de similitud empleada por los sistemas del estado del arte. De esta manera, esta decisión permite comparar los resultados obtenidos en el caso ideal del criterio de selección del umbral de similitud de los mejores sistemas del estado del arte con respecto a los que obtienen otros métodos que no requieren datos de entrenamiento, como los que se presentarán posteriormente en esta memoria.

Por otro lado, Artiles [2009] señala que el procedimiento de obtención del umbral de similitud descrito anteriormente no presenta los mejores resultados posibles que puede obtener HAC. Esto se debe a que los mejores resultados posibles para cada nombre de persona normalmente se obtienen empleando valores de umbral de similitud diferentes,

que habitualmente dependen del grado de ambigüedad del nombre de persona. Por este motivo, la cota superior de HAC consiste en tomar los resultados obtenidos por el mejor umbral para cada nombre de persona.

3.2.2. *Affinity Propagation* (AP)

Frey y Dueck [2007] propusieron este algoritmo de *clustering* basado en grafos que no requiere conocer a priori el número de *clusters* en los que se va a dividir el conjunto de objetos. AP interpreta la colección de objetos como una red en la que cada nodo se corresponde con un objeto y las aristas contienen valores de *afinidad* entre nodos inicialmente obtenidos a partir de sus similitudes. El algoritmo asigna un *ejemplar* a cada objeto consistente en otro objeto de la colección que intuitivamente puede ser visto como un punto céntrico de la red de nodos, o el punto céntrico de un *cluster* de acuerdo a su similitud con el resto de sus vecinos de la red. Este proceso continua hasta que se encuentra un conjunto de ejemplares que maximizan la suma de similitudes de todos ellos con respecto a cada uno de sus nodos asociados.

AP recibe como entrada la *matriz de similitudes* formada por los valores de similitud entre cada par de objetos y un cierto número de iteraciones T. Inicialmente, se asigna a cada objeto un valor, denominado *preferencia*, cuyo objetivo consiste en identificar *candidatos* proclives a ser escogidos como ejemplares. Los autores recomiendan calcular los valores de preferencia de cada objeto como la mediana de las similitudes entre el objeto en cuestión con respecto al resto. Una vez identificados los posibles candidatos a ejemplares, se aplica un proceso iterativo por el cual cada nodo transmite *mensajes de afinidad* al resto de nodos con el objetivo de mejorar la elección de los ejemplares. Estos mensajes de afinidad se dividen en dos clases:

- **Mensajes de responsabilidad:** cada nodo n informa a cada candidato c sobre su disposición para elegirle como ejemplar. Para ello, se toma en cuenta tanto el valor de afinidad entre ambos, acumulado en las anteriores iteraciones, como los valores de afinidad entre el nodo n y el resto de candidatos.
- **Mensajes de disponibilidad:** cada candidato c informa a cada nodo n sobre cómo es de apropiado para ser su ejemplar. Para ello, se toma en cuenta tanto el valor de afinidad entre ambos, acumulado en las anteriores iteraciones, como los valores de afinidad entre el candidato c y el resto de nodos.

En cada iteración se actualizan los mensajes de ambos tipos de forma que al combinarlos entre sí se pueda elegir de manera más precisa el conjunto de ejemplares final. El proceso termina cuando estos valores convergen o cuando se han efectuado el número de iteraciones T dado por parámetro.

Este algoritmo tiene un coste en $\mathcal{O}(N^2T)$ siendo N el número de nodos. No obstante, Fujiwara et al. [2011] presentaron una versión más eficiente de este algoritmo basada en identificar y podar mensajes innecesarios entre los nodos de la red, de modo que la complejidad temporal se encuentra en $\mathcal{O}(MT)$, donde M es el número de aristas de la red que, normalmente, suele ser mucho menor que N.

La justificación del uso de este algoritmo en los experimentos se debe a que tiene dos características comunes con las propuestas presentadas en esta tesis: no requiere conocer el número de *clusters* ni tampoco necesita aprender ningún tipo de parámetro mediante datos de entrenamiento. En particular, se ha utilizado la implementación de AP proporcionada por APRO[3]. Los valores iniciales de preferencia se han calculado mediante la mediana de los valores de similitud de cada nodo con el resto de objetos, y se ha prefijado el número de iteraciones en $T = 100$ tal y como recomiendan los autores originales del algoritmo [Frey y Dueck, 2007].

3.3. Colecciones de evaluación

En este apartado se describen las colecciones de evaluación de desambiguación de nombres de persona en la Web que se emplearán a lo largo de esta tesis para evaluar las propuestas presentadas y compararlas con los sistemas de desambiguación del estado del arte. Todas ellas tienen en común que se componen de nombres de persona junto con sus respectivos rankings de sitios web devueltos por algún buscador comercial. En particular, se han utilizado las tres colecciones proporcionadas por las campañas de evaluación WePS, porque son colecciones de referencia para este problema y han sido ampliamente utilizadas para evaluar el rendimiento de los sistemas de desambiguación de nombres de personas. Por otro lado, se ha empleado la colección ECIR 2012 [Berendsen et al., 2012], recopilada para estudiar el impacto de las redes sociales en este contexto. Además, también se ha usado la colección MC4WePS [Montalvo et al., 2016] porque presenta un escenario multilingüe para este problema, a diferencia del resto de colecciones. Finalmente, compararemos todas estas colecciones de evaluación y destacaremos las principales características de cada una de ellas.

3.3.1. WePS-1

Pese a que la desambiguación de nombres de personas había sido tratada con anterioridad, no existían colecciones de evaluación adecuadas para este escenario, puesto que las existentes por aquel entonces eran demasiado pequeñas [Mann y Yarowsky, 2003; Pedersen et al., 2005; Bollegala et al., 2006] o estaban sesgadas de alguna manera,

[3] http://www.apro.u-psud.fr/

como la presentada por Wan et al. [2005], donde hay una alta presencia de nombres de celebridades.

Artiles et al. [2007] presentan la colección WePS-1 utilizada en la primera campaña de evaluación WePS, y que se encuentra disponible libremente para la comunidad científica[4]. Esta colección se divide en datos de entrenamiento y datos de test, los cuales pasamos a describir a continuación.

Datos de entrenamiento

Los datos de entrenamiento de WePS-1 consisten en 49 nombres de persona de origen anglosajón junto con sus respectivos resultados de búsqueda devueltos por Yahoo! escritos en lengua inglesa. Las páginas web fueron descargadas completamente, por lo que los sistemas podían hacer uso de información adicional como, por ejemplo, las imágenes incluidas. Adicionalmente, para cada resultado de búsqueda se almacenaron otros datos como la URL, la posición en el ranking, el título y el *snippet* generado por el motor de búsqueda. Este conjunto de nombres de persona pueden dividirse en tres grupos de acuerdo al criterio escogido para su elección:

- **Censo de EEUU:** se reutilizó la colección *Web03* [Mann, 2006] formada por 32 nombres de personas extraídos aleatoriamente del Censo de los EEUU. Cada nombre de persona tiene asociado un número diferente de resultados de búsqueda que varía entre 2 (ej. *Cathy Ely*) y 405 (ej. *Cynthia Voigt*). Según Artiles [2009], esto se debe a que varios de estos nombres seleccionados aleatoriamente son muy poco frecuentes, de modo que el motor de búsqueda devolvió un número muy reducido de resultados al consultarlos.
- **Wikipedia:** los organizadores de WePS seleccionaron otros 7 nombres de personas bajo el criterio de que tuviesen alguna entrada en la edición en inglés de Wikipedia. Para cada uno de estos nombres se extrajeron alrededor de 100 resultados de búsqueda.
- **ECDL'06:** los organizadores de WePS seleccionaron aleatoriamente 10 nombres de personas que aparecían en la lista del Comité de Programa de la conferencia *European Conference on Digital Libraries* (ECDL), celebrada en el año 2006. Para cada uno de estos nombres se extrajeron alrededor de 100 resultados de búsqueda.

Datos de test

Los datos de test de WePS-1 fueron recopilados siguiendo la misma metodología que los datos de entrenamiento. En particular, se extrajeron 30 nombres de persona de origen anglosajón junto con sus respectivos resultados de búsqueda devueltos por Yahoo!

[4] `http://nlp.uned.es/weps/weps-1/weps1-data`

escritos en lengua inglesa. Del mismo modo que en la colección de entrenamiento, las páginas web fueron descargadas completamente y se almacenó para cada resultado de búsqueda su URL, posición en el ranking, título y *snippet* asociado. En particular, se recopilaron alrededor de 100 páginas web para cada uno de los nombres de persona, a diferencia de los datos de entrenamiento. Los organizadores dividieron los nombres de persona seleccionados de la misma manera que los datos de entrenamiento:

- **Censo de EEUU:** A diferencia de los datos de entrenamiento, los organizadores de WePS seleccionaron 10 nombres que consideraron relativamente comunes para evitar que hubiese nombres de persona que no tuviesen asociados al menos 100 resultados de búsqueda.
- **Wikipedia:** los organizadores de WePS seleccionaron otros 10 nombres de personas que tuviesen una entrada en la edición en inglés de Wikipedia.
- **ACL'06:** los organizadores de WePS seleccionaron aleatoriamente 10 nombres de personas que aparecían en la lista del Comité de Programa de la conferencia *Association for Computational Linguistics* (ACL), celebrada en el año 2006.

Tanto los datos de entrenamiento como los de test fueron anotados manualmente por expertos. Los anotadores se fijaron en que algunos resultados de búsqueda no ofrecían información suficiente para identificar al individuo al que hacían referencia o, simplemente, no contenían ninguna mención de los individuos consultados. Estos sitios web fueron identificados como no relevantes por los anotadores, de forma que se decidió que no se tomasen en cuenta durante el proceso de evaluación aunque no se eliminasen de la colección. Por otro lado, la consistencia de las anotaciones de la colección de test fue verificada tras comparar dos anotaciones realizadas de manera independiente por dos personas diferentes.

3.3.2. WePS-2

En la segunda edición de las campañas WePS [Artiles et al., 2009b], solamente se publicó una colección de test, dado que el entrenamiento podía realizarse utilizando los nombres de personas de la colección WePS-1. Esta colección también se encuentra disponible libremente para la comunidad científica[5].

La colección WePS-2 contiene 30 nombres de persona recopilados bajo el criterio empleado durante la campaña WePS-1. No obstante, hay algunas diferencias. Por un lado, los organizadores proporcionaron solamente el fichero HTML de cada página web y,

[5] `http://nlp.uned.es/weps/weps-2/weps2-data`

por otro lado, se tomaron en cuenta los 150 primeros resultados de búsqueda devueltos por Yahoo!. La selección de los nombres de persona siguió criterios similares a los de la colección de test de WePS-1, salvo pequeños detalles:

- **Censo de EEUU:** los organizadores de WePS seleccionaron 10 nombres de persona a partir del Censo de EEUU. En este caso, la selección se realizó tomando combinaciones de nombres y apellidos por separado. Posteriormente, se calculó la probabilidad de seleccionar cada nombre y apellido de acuerdo a su frecuencia en el Censo con el objetivo de que estos nombres tuviesen diferente grado de ambigüedad entre sí.
- **Wikipedia:** se seleccionaron aleatoriamente 10 nombres de personas extraídos de una lista de biografías de la edición en inglés de Wikipedia.
- **ACL'08:** se seleccionaron aleatoriamente 10 nombres de personas que aparecían en la lista del Comité de Programa de la conferencia ACL celebrada en 2008.

Al igual que en WePS-1, esta colección fue anotada por un par de expertos que discutieron entre sí sus anotaciones con el fin de alcanzar una anotación acordada. Por otra parte, los criterios para identificar resultados de búsqueda no relevantes fueron mucho más exigentes que en WePS-1, ya que se incluían redes sociales, páginas de genealogías y páginas web escritas en otros idiomas. La presencia de éstas últimas se debe a que unos pocos nombres de personas incluidos en WePS-2 no son de origen anglosajón. No obstante, representan un número reducido de páginas web porque en la configuración de las búsquedas se pidió exclusivamente páginas web escritas en inglés.

3.3.3. WePS-3

Artiles et al. [2010] describen la colección de test empleada en la tercera edición de las campañas WePS, que también se distribuye libremente en Internet[6]. Esta colección contiene 300 nombres de personas, mayoritariamente de origen anglosajón. Para cada uno de ellos se proporcionan alrededor de 200 páginas web escritas en inglés devueltas por Yahoo!, junto con sus URLs, posiciones en el ranking y *snippets*. En este caso, la clasificación de los nombres de persona incluye nuevos tipos según el perfil profesional de algunos de los individuos:

- **Censo de EEUU:** se seleccionaron aleatoriamente 50 nombres de persona presentes en el Censo de EEUU.

[6] `http://nlp.uned.es/weps/weps-3/data`

- **Wikipedia:** se seleccionaron aleatoriamente 50 nombres de personas extraídos de una lista de biografías de la edición en inglés de Wikipedia.
- **Conferencia:** se seleccionaron aleatoriamente 50 nombres de personas que aparecían en la lista del Comité de Programa de un congreso de informática.
- **Abogado:** se seleccionaron 50 nombres de personas para los que al menos un individuo es abogado.
- **Ejecutivo:** se seleccionaron 50 nombres de personas para los que al menos un individuo tiene un puesto ejecutivo en una empresa.
- **Agente inmobiliario:** se seleccionaron 50 nombres de personas para los que al menos un individuo es agente inmobiliario.

La principal diferencia de esta colección con respecto a las anteriores es que únicamente se anotaron los resultados de búsqueda correspondientes a dos individuos por nombre de persona, excepto en el caso de los nombres de tipo Conferencia, para los que solamente se anotaron las páginas web correspondientes a un único individuo. De este modo, la evaluación en esta colección solamente tuvo en cuenta a lo sumo dos *clusters* por nombre de persona, en lugar de considerar todos los *clusters*.

3.3.4. ECIR 2012

Berendsen et al. [2012] recopilaron una nueva colección de desambiguación de nombres de personas a la que se denominará ECIR 2012 y que se encuentra disponible para la comunidad científica[7]. ECIR 2012 fue recopilada con el objetivo de estudiar el impacto de las redes sociales en el problema. La colección contiene 33 nombres de persona de origen neerlandés, junto con sus respectivos rankings de búsqueda formados por páginas web escritas en ese idioma. Al igual que en la colección de entrenamiento de WePS-1, el número de resultados de búsqueda para cada nombre de persona es diferente. En particular, el rango se encuentra entre 27 páginas web (*Kim de Vogel*) y 164 (*Marieke de Jong*).

Los nombres de persona fueron seleccionados a partir del registro de consultas de un motor de búsqueda de personas neerlandés. Este buscador de personas incluye resultados de búsqueda recuperados por Google, Yahoo! y Bing, por lo que las páginas web de cada nombre de persona no consisten en un ranking de resultados de búsqueda real devuelto por un buscador convencional. Weerkamp et al. [2011] describe el registro de consultas utilizado en el que se incluye información de identificación de los usuarios

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

del buscador de personas junto con los clicks que realizaron tras introducir cada consulta y las marcas temporales (*timestamps*) de cada acción. La selección de los nombres se hizo en base al número de usuarios que los buscaron y al número de páginas web sociales devueltas. Para cada resultado de búsqueda solamente se aclara en cuáles de los tres buscadores aparecen y la posición del ranking en cada uno de ellos.

El criterio de selección de resultados de búsqueda no relevantes incluye páginas web que no contienen el nombre de persona consultado y páginas correspondientes a buscadores verticales especializados en nombres de personas, que habitualmente contienen enlaces a perfiles sociales de diferentes individuos denominados de la misma manera.

3.3.5. MC4WePS

Recientemente, Montalvo et al. [2016] recopilaron una nueva colección de desambiguación de nombres de personas denominada MC4WePS que también está disponible para la comunidad científica[8]. La colección MC4WePS fue recopilada con el objetivo de presentar un escenario multilingüe en el problema. Esta colección contiene 100 nombres de persona de distinto origen, aunque mayoritariamente anglosajón e hispano. Para cada uno de ellos, se recopilaron los 110 primeros resultados de búsqueda devueltos por Google o Yahoo! junto con sus posiciones en el ranking, sus URLs y los idiomas en los que están escritos. Todos estos datos fueron anotados por lingüístas de la *Universidad Autónoma de Madrid*.

MC4WePS se distingue del resto de colecciones porque incluye resultados de búsqueda escritos en diferentes idiomas para algunos de los nombres de persona. Esto se debe a que durante la recopilación de la colección se utilizó navegación privada mediante Firefox y la opción de búsqueda avanzada de Google y Yahoo! que permiten devolver sitios web en cualquier idioma. En particular, los anotadores identificaron 30 idiomas diferentes en esta colección, aunque prevalecen el inglés y el castellano, puesto que un 96.08 % de los resultados de búsqueda están escritos en alguno de estos dos idiomas.

Por otro lado, MC4WePS también se diferencia del resto de colecciones porque incluye resultados de búsqueda que no son páginas web sino documentos en otros formatos (ej. pdf, doc, etc.). Pese a que los motores de búsqueda son capaces de devolver este tipo de documentos, el resto de colecciones no los tuvieron en consideración y no se incluyeron en el conjunto de datos. Finalmente, MC4WePS también incluye páginas sociales debido al éxito de estas plataformas en el momento en el que la colección fue recopilada.

La selección de nombres de personas llevada a cabo en MC4WePS se basa en distintos criterios de acuerdo a su grado de ambigüedad y la existencia de multilingüismo en

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

los resultados de búsqueda y a nivel de *cluster*. Por un lado, se considera que un *nombre de persona* es *muy ambiguo* (MA) si es compartido por más de 10 individuos, mientras que en caso contrario es considerado como *poco ambiguo* (PA). En particular, aquellos nombres para los que todos los resultados de búsqueda obtenidos se corresponden con un único individuo se denominan *no ambiguos* (NA). Por otro lado, los nombres de persona se dividen entre monolingües (MN) y multilingües (ML), de acuerdo a los idiomas en los que están escritos los resultados de búsqueda. Además, se realiza otra distinción según haya *clusters* monolingües o multilingües: aquellos nombres de persona para los que todos los *clusters* son monolingües se denominan *monolingües a nivel de cluster* (MN Clust.), mientras que aquellos en los que no sucede esta situación se denominan *multilingües a nivel de cluster* (ML Clust.).

La Tabla 3.1 muestra los porcentajes de cada tipo de nombre de persona de acuerdo a la clasificación anterior.

Grado Amb.	**MN** (21 %)	**ML** (79 %)	**MN Clust.** (37 %) **/ ML Clust.** (63 %)
NA (4 %)	1 %	3 %	1 % / 3 %
PA (45 %)	9 %	36 %	15 % / 30 %
MA (51 %)	11 %	40 %	21 % / 30 %

Tabla 3.1: Porcentajes según grado de ambigüedad y multilingüismo de los nombres de persona incluidos en MC4WePS.

La tabla muestra que MC4WePS equilibra el número de nombres de persona muy ambiguos y poco ambiguos, mientras que, por otro lado, hay una mayor proporción de nombres de persona multilingües con respecto a monolingües. Además, hay 37 nombres monolingües a nivel de cluster, donde 21 de ellos lo son por el hecho de ser monolingües. Por tanto, únicamente para 16 nombres de persona multilingües se cumple que cada individuo (*cluster*) tiene asociados resultados de búsqueda escritos en el mismo idioma. Esto indica que el idioma en el que están escritos los resultados de búsqueda no es una pista de utilidad a la hora de identificar a los diferentes individuos.

3.3.6. Comparativa entre colecciones

La Tabla 3.2 resume las principales características de las colecciones de evaluación que hemos descrito anteriormente. En particular, se muestra el número de nombres de personas que contiene cada colección (#N), el número de resultados de búsqueda total (#W), el número de resultados de búsqueda por nombre de persona (#W/N), los porcentajes de los tipos de nombres de persona según su grado de ambigüedad tomando el criterio de MC4WePS [Montalvo et al., 2016], el porcentaje de páginas sociales (S %),

el idioma de los resultados de búsqueda (EN: inglés, NL: neerlandés) y, finalmente, si la colección es monolingüe (MN) o multilingüe (ML).

Colección	#N	#W	#W/N	NA %	PA %	MA %	S %	IDIOMA	MN/ML
WePS-1 (entr.)	49	3477	2 - 405	12.24 %	53.06 %	34.70 %	0.6 %	EN	MN
WePS-1 (test)	30	2968	$\sim$ 100	3.33 %	3.33 %	93.34 %	1.52 %	EN	MN
WePS-2	30	3444	$\sim$ 150	10.00 %	33.33 %	56.67 %	2.41 %	EN	MN
WePS-3	300	57357	$\sim$ 200	(*)	(*)	(*)	2.76 %	EN	MN
ECIR 2012	33	3487	27 - 164	0.00 %	18.18 %	81.82 %	34.73 %	NL	MN
MC4WePS	100	10432	$\sim$ 100	4.00 %	45.00 %	51.00 %	8.36 %	Varios	ML

Tabla 3.2: Comparativa de las colecciones de evaluación de desambiguación de nombres de personas en la Web.

En cuanto a la ambigüedad de los nombres de persona, la colección de entrenamiento de WePS-1 es la que presenta un mayor porcentaje de nombres poco ambiguos. Esto se debe a que varios de ellos tienen asociado un número reducido de páginas web porque son nombres muy poco frecuentes, de modo que el motor de búsqueda extrajo muy pocos resultados. Por su parte, la colección de test de WePS-1 y ECIR2012 contienen un porcentaje muy elevado de nombres ambiguos. De acuerdo con Artiles [2009], en el caso de WePS-1, esto indica que aunque intuitivamente se pueda pensar que aquellos nombres de persona que tengan asociada una entrada en Wikipedia sean menos ambiguos, esto no tiene por qué ser así. En el caso de ECIR2012, Weerkamp et al. [2011] observaron que los apellidos de los nombres de persona seleccionados se encuentran entre los más frecuentes en los Países Bajos, lo cual explica que haya tantos nombres muy ambiguos. Por otro lado, las colecciones WePS-2 y MC4WePS contienen porcentajes más equilibrados de cada tipo de nombre de persona. Finalmente, no podemos estudiar la ambigüedad de los nombres de persona incluidos en WePS-3 (marcado con (*) en la tabla) puesto que no se anotaron todos los documentos, de modo que no se conoce el número de individuos mencionados en los resultados de búsqueda de cada nombre de persona.

En cuanto a los porcentajes de páginas sociales, las colecciones WePS incluyen un número muy reducido de este tipo de páginas web. Esto se explica porque este tipo de plataformas no eran tan populares en el momento en el que se recopilaron estas colecciones. ECIR 2012 incluye un alto número de este tipo de páginas porque la selección de nombres de persona y de resultados de búsqueda tuvo por objeto incluir este tipo de páginas web aunque no formasen parte de un ranking de resultados real devuelto por un buscador. Finalmente, en la colección MC4WePS el porcentaje de páginas sociales es superior al de las colecciones WePS debido a la creciente popularidad de las redes sociales en los últimos años. Por otra parte, MC4WePS contiene un menor porcentaje de

páginas web sociales que ECIR 2012 porque fueron extraídas de rankings de resultados reales, sin que se añadiesen de manera artificial.

En cuanto al multilingüismo, MC4WePS es la única colección que contiene rankings de resultados de búsqueda escritos en varios idiomas. En cambio, las colecciones WePS contienen páginas web escritas en inglés, mientras que ECIR 2012 contiene resultados de búsqueda escritos en neerlandés.

La mayoría de estas colecciones son de utilidad para evaluar la desambiguación de nombres de persona desde distintos puntos de vista:

- **WePS-1 (entrenamiento):** permite evaluar el rendimiento de los sistemas de desambiguación de personas para nombres poco ambiguos y teniendo en cuenta un número dispar de resultados de búsqueda.
- **WePS-1 (test):** permite evaluar el rendimiento de los sistemas de desambiguación para nombres de personas muy ambiguos.
- **WePS-2:** permite evaluar el rendimiento de los sistemas de desambiguación sobre un conjunto de nombres de persona equilibrado con respecto al grado de ambigüedad.
- **ECIR 2012:** permite estudiar el impacto de las redes sociales en el problema. Además, nos permite evaluar el rendimiento de los sistemas de desambiguación de personas en un idioma diferente al de las colecciones WePS.
- **MC4WePS:** permite estudiar el impacto tanto de las redes sociales como del multilingüismo en el problema. En esta colección también hay un equilibrio de los nombres de persona de acuerdo a su grado de ambigüedad. Además, al tratarse de una colección recopilada recientemente, presenta un escenario de búsqueda más actual.

En el caso de la colección WePS-3, solamente se evalúan uno o dos *clusters* para cada nombre de persona en lugar de evaluar todo el agrupamiento, por lo que no se dispone de información sobre la naturaleza de los *clusters* anotados ni el grado de ambigüedad de cada uno de los nombres de personas.

3.4. Métricas de evaluación

Las métricas de evaluación nos permiten evaluar el rendimiento de un sistema para un determinado problema. En particular, las métricas de evaluación en *clustering* pueden clasificarse en dos tipos: intrínsecas y extrínsecas. Las *métricas de evaluación intrínsecas* se

basan en medir cómo de similares son los objetos que pertenecen a un mismo *cluster* y cómo de distintos son los objetos pertenecientes a distintos *clusters*, por lo que requieren de una cierta medida de similitud para poder calcularse. Por su parte, las *métricas de evaluación extrínsecas* son aquellas que se basan en comparaciones entre la salida generadas por los sistemas de *clustering* y un determinado *gold standard* que incluye la solución del agrupamiento, habitualmente construida manualmente por anotadores humanos. Estas últimas son las más comunes en problemas de *clustering*. En particular, todas las colecciones de desambiguación de nombres de personas cuentan con dicho *gold standard*, de modo que los sistemas han sido evaluados mediante métricas de evaluación extrínsecas.

Los organizadores de la campaña WePS-1 [Artiles et al., 2007] emplearon las métricas de evaluación *pureza* (*purity*), *pureza inversa* (*inverse purity*) y la medida-F de ambas métricas para medir el rendimiento de los sistemas de desambiguación. No obstante, posteriormente estos autores [Artiles et al., 2009b] se dieron cuenta de que puede construirse un sistema de desambiguación tramposo con un rendimiento competitivo con estas métricas, consistente en devolver el siguiente agrupamiento solapado:

- Cada resultado de búsqueda conforma un *cluster* diferente.
- Se añade un *cluster* adicional que contiene a todos los resultados de búsqueda.

Por este motivo, en las campañas WePS-2 y WePS-3 se utilizaron las *métricas B-Cubed* propuestas por Bagga y Baldwin [1998], de modo que la mayoría de los sistemas del estado del arte han sido evaluados con ellas. Estas métricas consisten en la *precisión B-Cubed* (BP), la *cobertura B-Cubed* (BR) y la medida-F de las dos métricas anteriores. Su elección se justifica porque cumplen una serie de restricciones formales adecuadas para la evaluación del *clustering* [Amigó et al., 2009], a diferencia de otros tipos de métricas extrínsecas. Además, las métricas *B-Cubed* también han sido empleadas para evaluar experimentos sobre otras colecciones más recientes, como ECIR 2012 [Berendsen et al., 2012] o MC4WePS [Montalvo et al., 2016], y otros problemas relacionados, como la desambiguación de nombres de personas en noticias [Bentivogli et al., 2013].

Las *métricas B-Cubed* se pueden definir formalmente de la siguiente manera: sean $\mathcal{C} = \{D_1, D_2, \ldots, D_N\}$ el conjunto de documentos que se quiere dividir en grupos, $\mathcal{S} = \{C_1^s, C_2^s, \ldots, C_l^s\}$ el agrupamiento de $\mathcal{C}$ generado por un cierto sistema de *clustering* y $\mathcal{G} = \{C_1^g, C_2^g, \ldots, C_k^g\}$ el *gold standard* asociado. La *precisión B-Cubed* (BP) y la *cobertura B-Cubed* (BR) se definen como:

$$BP = \frac{1}{N} \sum_{i=1}^{l} \frac{1}{|C_i^s|} \sum_{x_i \in C_i^s} \sum_{x_j \in C_i^s} g(x_i, x_j) \tag{3.13}$$

$$BR = \frac{1}{N} \sum_{i=1}^{k} \frac{1}{|C_i^g|} \sum_{x_i \in C_i^g} \sum_{x_j \in C_i^g} s(x_i, x_j) \tag{3.14}$$

donde:

$$g(x_i, x_j) = \begin{cases} 1 \quad si \quad \exists m \in \{1, 2, \ldots, k\} : x_i \in C_m^g \wedge x_j \in C_m^g \\ \\ 0 \quad si \quad \nexists m \in \{1, 2, \ldots, k\} : x_i \in C_m^g \wedge x_j \in C_m^g \end{cases} \tag{3.15}$$

$$s(x_i, x_j) = \begin{cases} 1 \quad si \quad \exists m \in \{1, 2, \ldots, l\} : x_i \in C_m^s \wedge x_j \in C_m^s \\ \\ 0 \quad si \quad \nexists m \in \{1, 2, \ldots, k\} : x_i \in C_m^s \wedge x_j \in C_m^s \end{cases} \tag{3.16}$$

Nótese que el cálculo de BP y BR se basa en determinar el número de pares de objetos que pertenecen al mismo *cluster* tanto en el *gold standard* (ver fórmula 3.15) como en la solución generada por el sistema (ver fórmula 3.16).

La manera de combinar ambas métricas viene dada por la *medida-F* de *van Rijsbergen* [van Rijsbergen, 1974], definida como sigue:

$$F_\alpha = F_\alpha(BP, BR) = \frac{1}{\alpha \frac{1}{BP} + (1 - \alpha) \frac{1}{BR}} \tag{3.17}$$

donde $\alpha \in [0, 1] \subset \mathbb{R}$

En esta tesis emplearemos la métrica $F_{0,5}$, debido a que es la métrica oficial de las campañas WePS y, por tanto, permite comparar nuestras propuestas con los sistemas del estado del arte. Este valor se corresponde con la media armónica de BP y BR, por lo que pesa a ambas por igual. No obstante, Artiles et al. [2009b] toman también en consideración el valor $F_{0,2}$, que pesa más el valor de cobertura BR, bajo la intuición de que al usuario le es menos perjudicial tener que realizar un pequeño filtrado entre las páginas web de los *clusters* generados en lugar de perder resultados de búsqueda correspondientes al individuo del que le interesa obtener información.

Finalmente, la métrica oficial utilizada en las campañas de evaluación WePS para rankear los sistemas de desambiguación consiste en la media aritmética de los valores $F_{0,5}$ de todos los nombres de persona incluidos en la colección de evaluación correspondiente. Esta métrica será la que se empleará en la experimentación de esta tesis doctoral.

3.5. Estudio de la significancia estadística

Los *test de significancia estadística* sirven para cuantificar la probabilidad de que la diferencia entre dos muestras no sea producto del azar. En particular, son habitualmente empleados para determinar si existen diferencias significativas entre los resultados obtenidos por distintos experimentos. Estos tests pueden clasificarse en dos tipos: *paramétricos* y *no paramétricos*. Los primeros asumen que la muestra sigue una distribución normal, mientras que los segundos no realizan ninguna asunción sobre la distribución de los datos de la población.

En esta tesis, se ha empleado el test no paramétrico sobre muestras dependientes denominado *prueba de los rangos con signo de Wilcoxon* o *test de Wilcoxon* [Wilcoxon, 1945]. Esto se justifica porque la muestra de datos tomada para comparar los experimentos son los resultados $F_{\alpha=0,5}$ obtenidos por los sistemas de desambiguación para cada nombre de persona. Por un lado, no podemos asumir que estos valores sigan una distribución normal, por lo que se ha de seleccionar un test no paramétrico. Por otro lado, se trata de una comparativa entre muestras dependientes. En particular, esta prueba estadística ha sido empleada en varios problemas de *clustering* (ej. [Zhao et al., 2012; Montalvo et al., 2015b]).

El test de Wilcoxon recibe dos pares de observaciones $X = (x_1, x_2, \ldots, x_N)$, $Y = (y_1, y_2, \ldots, y_N)$ donde x_i e y_i son los valores de la métrica $F_{\alpha=0,5}$ obtenidos por dos sistemas de desambiguación para el individuo i. Esta prueba consiste en verificar la *hipótesis nula* $H_0 : \mu_X = \mu_Y$ que indica que ambos resultados son equivalentes. Para ello, el test computa el denominado p-valor que cuantifica la evidencia de que H_0 se cumpla. Por otro lado, este valor se compara con un umbral α denominado *significatividad*. La significatividad define dos regiones de valores de aceptación o rechazo de la hipótesis nula. En particular, se fijará este valor en $\alpha = 0.05$, lo cual significa que habrá una diferencia significativa entre los resultados de dos sistemas de desambiguación con una probabilidad del 95 %. A partir de estos valores, el test rechaza H_0 si $p < \alpha$ y la acepta en caso contrario.

3.6. Conclusiones

En este capítulo se han detallado algunos elementos del marco de experimentación empleado para evaluar las propuestas presentadas en esta tesis doctoral. En particular:

- Se representarán los resultados de búsqueda siguiendo el modelo de espacio vectorial y se explorará el uso de diferentes funciones de pesado de términos y medidas de similitud.

- Se utilizarán dos algoritmos de *clustering* del estado del arte para verificar las hipótesis planteadas y comparar los resultados de las propuestas. Por un lado, se usará el algoritmo HAC porque ha sido utilizado por los mejores sistemas de desambiguación de nombres de persona. Por otro lado, se empleará el algoritmo AP porque cuenta con dos características comunes con las propuestas que serán presentadas posteriormente en esta memoria: no requiere conocer el número de *clusters* ni tampoco necesita datos de entrenamiento.

- Se emplearán varias colecciones de evaluación en desambiguación de nombres de personas en la Web. Cada una de ellas cuenta con distintas características relacionadas con el grado de ambigüedad de los nombres de personas, la presencia de las redes sociales y el multilingüismo en los resultados de búsqueda. Esto permitirá evaluar las propuestas presentadas desde diferentes perspectivas que pueden aparecer en un escenario real de búsqueda en la Web.

- Se ha descrito la manera en la que se compararán los resultados de diferentes sistemas de desambiguación de nombres de persona. Por un lado, se medirá su rendimiento mediante las métricas de evaluación *B-Cubed*, mientras que, por otro lado, se determinará la significancia estadística entre ellos mediante el test de Wilcoxon.

4

Primera aproximación a la desambiguación de nombres de persona en la Web

"Un algoritmo debe ser visto para ser creído."

— Donald Knuth —

Este capítulo presenta la primera propuesta en esta tesis para desambiguación de nombres de personas en la Web. La propuesta se divide en dos fases: una etapa de representación de los resultados de búsqueda y otra etapa de aplicación de un algoritmo de clustering para agruparlos de acuerdo al individuo que mencionan. En cuanto a la representación de los documentos, se enunciarán hipótesis que describirán qué clase de rasgos suponemos adecuados para representar el contenido de los resultados de búsqueda y se verificarán a través de experimentos preliminares. Con respecto a la agrupación de los resultados de búsqueda, este capítulo presenta un nuevo algoritmo de clustering. Este algoritmo se basa en el concepto de umbral adaptativo, cuyo uso evita tanto la necesidad de conocer a priori el número de clusters como el requerimiento de datos de entrenamiento para aprender el valor de algún parámetro. Posteriormente, se discutirán los resultados del algoritmo propuesto con respecto a otras aproximaciones del estado del arte y se analizarán sus ventajas e inconvenientes. Finalmente, se enumerarán las principales conclusiones que resumen el contenido de este capítulo.

4.1. Representación de los documentos

En esta sección se enuncian hipótesis sobre qué clase de rasgos pueden ser de utilidad para representar los documentos a la hora de distinguir entre diferentes individuos que comparten el mismo nombre. Posteriormente, se llevarán a cabo experimentos que permitirán verificar cada una de ellas.

4.1.1. Hipótesis

Como se explicó en el Capítulo 2, la representación de los sitios web normalmente se ha llevado a cabo mediante una amplia variedad de rasgos de distinto tipo de

acuerdo a la manera en la que son extraídos. Dado que el problema se enmarca en un escenario de búsqueda en la Web en el que los usuarios esperan obtener resultados rápidamente, resulta conveniente que las fases del proceso de desambiguación no tengan un coste computacional alto. El enfoque defendido en este trabajo consiste en aprovechar lo máximo posible la información contenida en los resultados de búsqueda evitando el empleo de recursos externos. En particular, las siguientes hipótesis, denotadas como RD (Representación de Documentos), se refieren a ciertos tipos de rasgos contenidos en los resultados de búsqueda:

- *(RD1) Dado un ranking de resultados de búsqueda obtenido tras consultar un nombre de persona, la coaparición de n-gramas es un criterio eficaz a la hora de decidir si dos sitios web del ranking hablan del mismo individuo. En concreto: la probabilidad de que dos resultados de búsqueda que comparten n-gramas hablen del mismo individuo es mayor a medida que el número de palabras n es mayor.*
- *(RD2) En los idiomas que emplean el alfabeto latino, las palabras que comienzan por una letra mayúscula aportan información especialmente útil a la hora de caracterizar individuos y poder distinguirlos entre sí.*

A continuación, se explica más detalladamente el por qué de las hipótesis anteriores:

n-gramas

Un documento puede verse como una secuencia ordenada de las frases que contiene $D = S_1 S_2 \dots S_k$, siendo $k \in \mathbb{N}$ el número de frases del documento. A su vez, cada frase $S_i \in D$ puede ser vista como una secuencia ordenada de palabras $S_i = w_1^i w_2^i \dots w_{m_i}^i$, donde $m_i \in \mathbb{N}$ es el número de palabras de la frase. Un n-grama es una secuencia de $n > 0$ palabras escritas consecutivamente en una frase, de modo que el conjunto de n-gramas de una frase S_i se define como:

$$\text{n-gramas}(n, S_i) = \{w_{j+1}^i \dots w_{j+n}^i | j \geq 0 \wedge j + n \leq m_i\} \tag{4.1}$$

Los n-gramas se componen de r-gramas donde $1 \leq r < n$, de modo que, por ejemplo, el 3-grama (*trigrama*) *Natural Language Processing* se descompone en los 2-gramas (*bigramas*) *Natural Language* y *Language Processing* y en los 1-gramas (*unigramas*) *Natural*, *Language* y *Processing*. Trivialmente, se cumple que si x es un n-grama de una frase S_i, entonces todos los r-gramas de los que se compone x son r-gramas de la frase S_i. Por otra parte, el número de n-gramas diferentes de una frase S_i que contiene $m_i \in \mathbb{N}$ palabras está acotado por $|\text{n-gramas}(n, S_i)| \leq m_i - (n-1)$ si $m_i \geq n$, y $|\text{n-gramas}(n, S_i)| = 0$ en caso contrario.

Un n-grama de un documento es un n-grama de alguna de las frases que contiene, de modo que el conjunto de n-gramas de un documento D se define formalmente como:

$$\text{n-gramas}(n, D) = \bigcup_{S_i \in D} \text{n-gramas}(n, S_i) \tag{4.2}$$

Por un lado, la hipótesis (RD1) afirma que la coaparición de n-gramas es un criterio eficaz a la hora de distinguir individuos diferentes. Algunos autores han corroborado esta afirmación. De acuerdo con Popescu y Magnini [2007], los n-gramas son capaces de capturar información que puede caracterizar a un determinado individuo. Por su parte, Nuray-Turan et al. [2012] afirman que los n-gramas sirven para capturar los tópicos de los que tratan los documentos porque tienen en cuenta el orden de aparición de las palabras.

Por otro lado, la hipótesis (RD1) también establece que cuanto mayor sea el número de palabras n de las que se componen los n-gramas compartidos por dos resultados de búsqueda, más probable es que ambos sitios web se refieran al mismo individuo. La justificación de esta afirmación viene del hecho de que la probabilidad de que n palabras aparezcan en la misma secuencia en dos documentos diferentes es muy baja cuando el valor de n es alto. En particular, los modelos de lenguaje basados en n-gramas (ej. [Caropreso et al., 2001]) se basan en esta idea puesto que la probabilidad que asignan a un n-grama es menor o igual a la probabilidad de cada uno de los $(n-1)$-gramas de los que se compone.

Palabras en mayúsculas

En los idiomas que emplean el alfabeto latino es habitual escribir los nombres propios comenzando por una letra en mayúscula. En lo sucesivo, nos referiremos a este tipo de rasgos como *palabras en mayúsculas*. Normalmente, estas palabras suelen corresponderse con NEs, por lo que ambos tipos de rasgos suelen estar relacionados. Las NEs han sido ampliamente usadas por los sistemas de desambiguación de personas porque pueden ofrecer información sobre lugares, organizaciones u otras personas relacionadas con cada individuo. No obstante, a continuación enumeramos una serie de razones que justifican el empleo de palabras en mayúsculas (hipótesis (RD2)) en lugar del uso de NEs:

- De acuerdo con Artiles et al. [2009a], el rendimiento de los sistemas de desambiguación que emplean NEs depende del sistema NER empleado para extraerlas. Esta situación no sucede con la extracción de rasgos escritos en mayúsculas.
- Los sistemas NER no siempre capturan información que suele escribirse en mayúsculas y que puede aportar pistas útiles a la hora de extraer información relacionada

con un cierto individuo. Por ejemplo, los títulos de películas, libros, videojuegos o programas de televisión pueden proporcionarnos información sobre las aficiones de una persona, y no siempre son reconocidos como NEs por parte de los sistemas NER.

Para verificar esta afirmación se ha realizado el siguiente experimento sobre la colección de entrenamiento de la campaña de evaluación WePS-1:

- Extraer las palabras escritas en mayúsculas de las páginas web tras llevar a cabo el preprocesamiento descrito en el apartado 3.1.1.
- Extraer las NEs de los resultados de búsqueda mediante el sistema NER de la Universidad de Stanford [Finkel et al., 2005]. Se ha empleado este sistema NER porque ha sido el más utilizado por los sistemas de desambiguación del estado del arte. En particular, se han extraído las NEs de las categorías PER, LOC, ORG y MISC, aunque los sistemas de desambiguación extraen solo NEs pertenecientes a las tres primeras categorías. La extracción de las NEs se ha llevado a cabo sobre los textos planos de las páginas web sin aplicar el resto de etapas de preprocesamiento, puesto que algunas de ellas (ej. *stemming* o eliminación de palabras vacías) pueden afectar en el rendimiento del sistema NER. No obstante, estas últimas etapas de preprocesamiento se han aplicado sobre cada una de las palabras de las NEs extraídas. Finalmente, se han dividido cada NE en las palabras que contiene.
- Calcular el porcentaje de palabras en mayúsculas que son capturadas por las NEs, y el porcentaje de palabras contenidas en las NEs que están escritas en mayúsculas. Este cálculo se ha realizado comparando los conjuntos de palabras calculados en los puntos anteriores de cada resultado de búsqueda de la colección, de modo que el porcentaje obtenido finalmente tiene en cuenta todos los resultados de búsqueda de la colección.

El resultado es que un 99,62 % de las palabras que forman parte de las NEs son palabras en mayúsculas, mientras que solamente un 37,63 % de las palabras en mayúsculas forman parte de las NEs. Esto significa que tomando palabras en mayúsculas se captura casi toda la información de las NEs de las categorías consideradas y, además, otros rasgos adicionales que no son identificados como NEs por el sistema de Stanford. Las palabras que forman parte de las NEs que no están escritas en mayúsculas se corresponden habitualmente con números (ej. años) o nombres de localidades escritos en minúsculas, que son identificados por el NER de Stanford mediante el uso de *gazetteers*. Una posible explicación al bajo porcentaje de palabras en mayúsculas que forman parte de las NEs es que, como afirman Dornescu et al. [2010], el rendimiento de los sistemas NER es peor en dominios con

información ruidosa, como el dominio Web, puesto que normalmente están entrenados con corpora compuestos por noticias, que normalmente suelen contener menos cantidad de ruido.

n-gramas en mayúsculas

Uniendo las hipótesis (RD1) y (RD2), se considera que la coaparición de n-gramas compuestos por palabras en mayúscula es un criterio eficaz a la hora de decidir si dos resultados de búsqueda hablan del mismo individuo. En lo sucesivo y por simplicidad, nos referiremos a este tipo de rasgos como *n-gramas en mayúsculas*. Además, la combinación de estos dos tipos de rasgos disminuye la probabilidad de obtener información que puede dar lugar a la agrupación de sitios web que hablan de distintos individuos que comparten el mismo nombre. Por ejemplo, de acuerdo con Nuray-Turan et al. [2012], algunas NEs compuestas por una única palabra son muy ambiguas y pueden dar lugar a agrupaciones incorrectas. Por ejemplo, varias personas con el mismo nombre pueden tener relación con otras personas llamadas *José*. Este tipo de rasgos ambiguos pueden evitarse tomando n-gramas compuestos por varias palabras.

La representación de los sitios web que utilizaremos en nuestra propuesta consistirá en vectores de números reales, donde cada componente del vector se corresponde con cada n-grama en mayúsculas que aparece en el conjunto de resultados de búsqueda.

4.1.2. Experimentos preliminares

En este apartado se describen una serie de experimentos preliminares que nos permitirán verificar las hipótesis sobre representación de documentos (RD1) y (RD2). Para ello, utilizaremos la colección de entrenamiento proporcionada en la campaña de evaluación WePS-1. Emplearemos diferentes tipos de rasgos que encajan con las hipótesis (RD1) y (RD2), y usaremos los algoritmos de *clustering* HAC y AP para agrupar los resultados de búsqueda. Por otro lado, los términos se pesan con TF-IDF y los documentos se comparan mediante la similitud coseno, debido a que se trata de la configuración más utilizada por los sistemas del estado del arte.

Rasgos empleados

Para verificar las hipótesis (RD1) y (RD2) se han seleccionado varios rasgos que cumplen algunas de las siguientes propiedades: (i) están compuestos por un número diferente de palabras; y (ii) están escritos en mayúsculas. A continuación, describiremos cada uno de ellos.

n-gramas

Emplearemos diferentes n-gramas en relación con el número de palabras de las que se componen para verificar la hipótesis (RD1). Además, para verificar la hipótesis (RD2) consideraremos tanto n-gramas en mayúsculas como n-gramas compuestos por cualquier tipo de palabra. Para este estudio preliminar, hemos tomado n-gramas compuestos entre 1 y 4 palabras extraídos tras aplicar el preprocesamiento detallado en el apartado 3.1.1.

k-skip-n-gramas:

Los k-skip-n-gramas son una generalización de los n-gramas consistentes en secuencias de n palabras en las que se permite que existan saltos entre ellas no superiores a $k \in \mathbb{N}$ palabras [Guthrie et al., 2006]. Por tanto, estos rasgos también encajan con la hipótesis (RD1). Al igual que con los n-gramas, consideraremos tanto los k-skip-n-gramas que estén compuestos por palabras escritas en mayúsculas, como aquellos compuestos por cualquier tipo de palabra con el fin de verificar la hipótesis (RD2).

El conjunto de k-skip-n-gramas de una frase $S_i = w_1^i w_2^i \dots w_{m_i}^i$ se define como:

$$\text{k-skip-n-gramas}(k, n, S_i) = \{w_{i_1}^i \dots w_{i_n}^i | \sum_{j=2}^{n} i_j - i_{j-1} \leq k+1\} \tag{4.3}$$

Se tiene que $\text{n-gramas}(n, D) \subseteq \text{k-skip-n-gramas}(k, n, D)$, porque los n-gramas son k-skip-n-gramas tomando $k = 0$. Por tanto, los k-skip-n-gramas generan un vocabulario mucho mayor que los n-gramas, consistente en información adicional que captura el contexto de cada una de las palabras. En particular, si $n = 1$, los k-skip-1-gramas coinciden con los 1-gramas para cualquier valor de k, puesto que ambos son el conjunto de palabras que aparecen en la frase, de modo que no hay saltos. Dado que los k-skip-n-gramas pueden incluir saltos no superiores a k, se cumple que si $k < l \Rightarrow \text{k-skip-n-gramas}(k, n, S_i) \subseteq \text{k-skip-n-gramas}(l, n, S_i)$, de modo que el vocabulario se extiende a medida que el valor k es mayor. Además, Guthrie et al. [2006] muestran que, a diferencia de los n-gramas, el número de k-skip-n-gramas de una frase se amplía cuando el número de palabras n es mayor. Esto se explica porque a medida que se toman en cuenta un mayor número de palabras, los k saltos pueden tomarse en un mayor número de posiciones.

Los k-skip-n-gramas de un documento $D = S_1 S_2 \dots S_k$ son k-skip-n-gramas que aparecen en alguna de sus frases, de modo que el conjunto de k-skip-n-gramas de un documento D se define de la siguiente manera:

$$\text{k-skip-n-gramas}(k, n, D) = \bigcup_{S_i \in D} \text{k-skip-n-gramas}(k, n, S_i) \tag{4.4}$$

Para este estudio preliminar, se han tomado k-skip-n-gramas con $k = 4$ posibles saltos entre palabras y variando el valor de n entre 1 y 4, igual que con los n-gramas. Estos rasgos se han extraído tras aplicar el preprocesamiento explicado en el apartado 3.1.1.

NEs:

Las NEs se ajustan a las hipótesis (RD1) y (RD2) porque habitualmente suelen estar compuestas por varias palabras escritas en mayúsculas. Por otro lado, han sido ampliamente utilizadas en el estado del arte para representar el contenido de los resultados de búsqueda. A diferencia de los rasgos anteriores, las NEs no se componen de un número n fijo de palabras y su extracción se lleva a cabo mediante un sistema NER. En particular, para este estudio, se han extraído las NEs de las categorías PER, LOC, ORG y MISC mediante el sistema NER de la Universidad de Stanford [Finkel et al., 2005]. Se han utilizado las NEs detectadas de varias categorías porque diferentes autores [Saggion, 2008; Artiles et al., 2009a] concluyen que el uso exclusivo de categorías concretas tiende a obtener pobres resultados de cobertura, debido a que no todos los resultados de búsqueda contienen NEs de una determinada categoría. El sistema NER se ha aplicado sobre el texto plano de los resultados de búsqueda sin que se hayan llevado a cabo las técnicas de preprocesamiento descritas en el apartado 3.1.1, puesto que pueden interferir en el rendimiento del sistema NER. No obstante, para cada NE identificada, se ha aplicado el algoritmo de Porter [Porter, 1980] a cada palabra que las compone y se han eliminado tanto las palabras vacías como las menciones del nombre y apellido del nombre de persona consultado.

Algoritmos

Como se explicó en la sección 3.2, se utilizarán los algoritmos de *clustering* HAC y AP para agrupar los resultados de búsqueda en los estudios preliminares. En ambos casos, los rasgos se pesan mediante la función TF-IDF y se usa la medida de similitud coseno para comparar los documentos, debido a que esta configuración ha sido ampliamente utilizada en el estado del arte cuando se representan los documentos mediante VSM.

Para ilustrar la importancia de la estimación del valor de umbral en HAC, en cada experimento se ha tomado el mejor valor de umbral de similitud promedio de la colección a través del procedimiento explicado en el apartado 3.2.1. Por su parte, el algoritmo AP no requiere conocer el valor de ningún parámetro, y se ha aplicado la configuración recomendada por sus autores originales [Frey y Dueck, 2007].

Criterios de valoración de los resultados

Antes de presentar los criterios por los que vamos a valorar los resultados de esta experimentación preliminar, presentaremos la notación empleada en este apartado. Una colección de evaluación de desambiguación de nombres de personas en la web es

un conjunto de rankings de resultados de búsqueda correspondientes a $M \in \mathbb{N}$ nombres de persona diferente, i.e. $\{\mathcal{W}(NP_1), \mathcal{W}(NP_2), \ldots, \mathcal{W}(NP_M)\}$, donde $\mathcal{W}(NP_i) = \{W_1(NP_i), W_2(NP_i) \ldots, W_{N_i}(NP_i)\}$ es el ranking de resultados devuelto por un motor de búsqueda al consultar el nombre de persona NP_i, conteniendo $N_i \in \mathbb{N}$ resultados de búsqueda. Además, $W_j(NP_i) \in \mathcal{W}(NP_i)$ es el j-ésimo resultado de búsqueda del ranking $\mathcal{W}(NP_i)$, y denotaremos mediante $F_j(NP_i)$ a su conjunto de rasgos.

Valoraremos la conveniencia de utilizar los rasgos empleados en este estudio teniendo en cuenta los siguientes criterios:

Porcentaje de resultados de búsqueda representados

Diremos que un *resultado de búsqueda* $W_j(NP_i)$ está *representado* si al menos contiene un rasgo, i.e., $F_j(NP_i) \neq \emptyset$. Los resultados de búsqueda representados pueden agruparse entre sí, de modo que es conveniente que la selección de rasgos empleada sea capaz de representar tantos resultados de búsqueda como sea posible. La manera de calcular este factor consiste en computar la proporción de resultados de búsqueda con representación de toda la colección, con respecto al número total de resultados de búsqueda de la colección:

$$R\,\% = 100 \cdot \frac{\sum_{i=1}^{M} |\{W_j(NP_i) \in \mathcal{W}(NP_i) | F_j(NP_i) \neq \emptyset\}|}{\sum_{i=1}^{M} |\mathcal{W}(NP_i)|}\,\% \tag{4.5}$$

Por otro lado, este factor influye en los resultados obtenidos por los algoritmos de *clustering* por el siguiente motivo: los resultados de búsqueda sin representación no pueden agruparse con ninguna otra página web, puesto que su similitud con el resto de documentos es nula. Por tanto, los algorimos de *clustering* los devolverán contenidos en *clusters* unitarios. Los *clusters* unitarios tienen un impacto positivo en los valores de precisión, pero un efecto perjudicial en los valores de cobertura. En particular, para los nombres de persona poco ambiguos, el impacto de los *clusters* unitarios es más perjudicial, puesto que los resultados de búsqueda hablan de un menor número de individuos (*clusters*). Por tanto, un valor bajo de $R\,\%$ tenderá a tener un impacto negativo en la colección de entrenamiento de WePS-1, dado que la mayoría de nombres de persona que contiene son de este tipo, como vimos en el apartado 3.3.6.

Tamaño del vocabulario

Un tamaño de vocabulario grande implica un mayor coste computacional en tiempo y espacio, puesto que se corresponde con la dimensión de los vectores que representan a los documentos en el modelo VSM. Por tanto, es preferible contar con un vocabulario pequeño en términos de eficiencia computacional. Para medir este factor, tomaremos la media aritmética de los tamaños de vocabulario de todos los rankings de resultados incluidos en la colección, i.e.:

$$Avg(|Voc|) = \frac{\sum_{i=1}^{M} |Voc(\mathcal{W}(NP_i))|}{M} \tag{4.6}$$

Métricas B-Cubed y estudio de la significancia estadística

Emplearemos las métricas *B-Cubed* (ver sección 3.4) para analizar el rendimiento de los algoritmos de acuerdo al tipo de rasgos empleados para representar los resultados de búsqueda. Se ha utilizado el evaluador distribuido por los organizadores de las campañas WePS para calcular estas métricas. Por otro lado, emplearemos el test de Wilcoxon (ver sección 3.5) para determinar cuándo existen mejoras significativas entre los diferentes experimentos llevados a cabo.

Grado de ambigüedad de los nombres de persona

Para decidir qué clase de rasgo es más adecuado, podemos examinar la sensibilidad de cada uno de ellos con respecto al grado de ambigüedad de los nombres de persona. En el contexto tratado en esta tesis, se entiende que el *grado de ambigüedad de un nombre de persona* está relacionado con el número de individuos diferentes presentes en los resultados de búsqueda asociados al nombre. De este modo, asumiendo que se toma un número $N \in \mathbb{N}$ fijo de resultados de búsqueda, se dice que un nombre de persona NP_i es más ambiguo que otro nombre de persona NP_j si el número de individuos distintos mencionados en el ranking $\mathcal{W}(NP_i)$ es mayor que el número de individuos mencionados en el ranking $\mathcal{W}(NP_j)$. Por ello, se tomarán nombres de persona con un número similar de resultados de búsqueda y distinto nivel de ambigüedad, y se analizará la variación de los valores de las métricas de evaluación entre ellos.

Resultados

La Tabla 4.1 muestra el valor de los factores anteriores para la colección de entrenamiento de la campaña de evaluación WePS-1, tomando los rasgos descritos anteriormente y aplicando los algoritmos de *clustering* AP y HAC. Nótese que los valores R % y $Avg(|Voc|)$ son independientes del algoritmo de *clustering* empleado. Como explicamos anteriormente, los k-skip-1-gramas se corresponden con los 1-gramas para cualquier valor de k, de modo que la tabla muestra únicamente los datos de éstos últimos cuando $n = 1$. Los n-gramas y k-skip-n-gramas compuestos exclusivamente por palabras en mayúsculas se distinguen con la marca *MAY* de aquellos compuestos por cualquier tipo de palabra. En cuanto a los algoritmos, la tabla muestra los valores promedio de las métricas *B-Cubed* y en el caso de HAC, además se muestra el valor del mejor umbral γ tomado para cortar el dendrograma para cada tipo de rasgo. Por otro lado, la tabla también muestra el resultado obtenido tras aplicar el test de significancia estadística entre distintos experimentos. El estudio de la significancia estadística se ha dividido en dos tipos: (i) significancia estadística empleando distintos tipos de rasgos y mismo algoritmo; y (ii) significancia estadística empleando el mismo tipo de rasgos y distinto

algoritmo. A continuación se explican los símbolos utilizados para cada tipo de estudio de la significancia estadística:

- **(i) Distinto tipo de rasgos / Mismo algoritmo:** este estudio nos permite evaluar los resultados con respecto al tipo de rasgos empleado para representar las páginas web. Para ello, hemos empleado la notación (k) donde $k \in \mathbb{N}$, de manera que aquellos experimentos marcados con (k) obtienen peores resultados con respecto a algún experimento marcado con (k') cuando $k' < k$ de acuerdo con el test de significancia estadística. En cambio, si $k = k'$, ambos experimentos obtienen resultados similares.
- **(ii) Mismo tipo de rasgos / Distinto algoritmo:** este estudio nos permite evaluar los resultados con respecto al algoritmo de *clustering* empleado para agrupar las páginas web. Si la columna correspondiente a uno de los algoritmos se marca con el símbolo † entonces obtiene mejoras significativas con respecto al otro algoritmo usando los mismos rasgos, de acuerdo al test de significancia estadística. Si no aparece esta marca en ninguno de los algoritmos entonces ambos obtienen resultados similares usando los mismos rasgos.

WePS-1 (entrenamiento)			**AP**			**HAC**			
Rasgos	**R %**	**Avg(|Voc|)**	**BP**	**BR**	**$F_{0,5}$**	γ	**BP**	**BR**	**$F_{0,5}$**
1-gramas	98.27 %	3850.10	0.80	0.53	0.55 (1)	0.13	0.78	0.87	0.79 (1) †
1-gramas *MAY*	97.87 %	1592.45	0.80	0.51	0.53 (1)	0.11	0.68	0.90	0.75 (2) †
2-gramas	95.46 %	3446.37	0.91	0.44	0.52 (1)	0.08	0.71	0.88	0.75 (2) †
2-gramas *MAY*	91.66 %	1375.59	0.91	0.43	0.51 (2)	0.07	0.68	0.92	0.75 (2) †
k-skip-2-gramas	96.17 %	14612.76	0.88	0.46	0.53 (1)	0.06	0.73	0.82	0.74 (2) †
k-skip-2-gramas *MAY*	92.87 %	3743.55	0.90	0.44	0.51 (2)	0.08	0.73	0.87	0.76 (2) †
3-gramas	77.88 %	1826.06	0.91	0.46	0.54 (1)	0.01	0.85	0.83	0.82 (1) †
3-gramas *MAY*	70.12 %	454.65	0.92	0.44	0.53 (1)	0.00	0.89	0.76	0.80 (1) †
k-skip-3-gramas	85.04 %	29549.02	0.90	0.45	0.52 (1)	0.03	0.82	0.82	0.81 (1) †
k-skip-3-gramas *MAY*	76.53 %	5536.0	0.91	0.45	0.53 (1)	0.01	0.86	0.82	0.82 (1) †
4-gramas	59.59 %	1126.37	0.92	0.43	0.52 (1)	0.00	0.94	0.62	0.72 (3) †
4-gramas *MAY*	49.67 %	239.08	0.93	0.44	0.53 (1)	0.00	0.96	0.53	0.65 (4) †
k-skip-4-gramas	68.91 %	52728.04	0.91	0.44	0.53 (1)	0.00	0.91	0.73	0.79 (1) †
k-skip-4-gramas *MAY*	58.96 %	9193.37	0.93	0.45	0.54 (1)	0.00	0.94	0.62	0.72 (3) †
NEs	91.45 %	872.38	0.88	0.44	0.50 (2)	0.15	0.76	0.82	0.76 (2) †

Tabla 4.1: Resultados obtenidos por los algoritmos AP y HAC para distintos tipos de rasgos en la colección de entrenamiento de la campaña de evaluación WePS-1.

Análisis de los resultados

A continuación, se presenta un análisis de los resultados desde diferentes puntos de vista:

- **Análisis con respecto a R % y Avg(|Voc|)**: los valores de $R\,\%$ y $Avg(|Voc|)$ de los k-skip-n-gramas siempre serán mayores o iguales con respecto a los de los n-gramas, puesto que éstos últimos son un subconjunto de los primeros. Esto significa que los n-gramas tienen la ventaja de que utilizan una representación menos costosa computacionalmente que los k-skip-n-gramas, pero cuentan con el inconveniente de que representan un menor número de resultados de búsqueda. En particular, la tabla muestra que el tamaño del vocabulario obtenido al emplear k-skip-n-gramas compuestos por cualquier palabra es, en la mayoría de los casos, un orden de magnitud superior al de los n-gramas.

  Análogamente, los n-gramas y k-skip-n-gramas compuestos por cualquier palabra son un superconjunto de aquellos compuestos sólamente por palabras en mayúsculas, de modo que los valores de $R\,\%$ y $Avg(|Voc|)$ de los primeros serán siempre mayores o iguales con respecto a los segundos. Esto significa que los rasgos compuestos por palabras en mayúsculas tienen la ventaja de que utilizan una representación menos costosa computacionalmente que los rasgos compuestos por cualquier palabra, pero cuentan con el inconveniente de que representan un menor número de resultados de búsqueda.

  El valor de $R\,\%$ disminuye en los n-gramas y k-skip-n-gramas a medida que se componen de un número n mayor de palabras. Esto se explica porque los rasgos compuestos por un menor número de palabras están contenidos en aquellos compuestos por un mayor número de palabras. Dado que los rasgos que aparecen únicamente en un documento se eliminan del vocabulario, esto implica que es menos probable que coaparezcan rasgos compuestos por un número mayor de palabras en dos resultados de búsqueda diferentes tal como asume la hipótesis (RD1). En particular, el valor de $R\,\%$ es superior al 90 % cuando se toman rasgos compuestos por 1 o 2 palabras. No obstante, este valor desciende drásticamente cuando se consideran rasgos compuestos por 3 o más palabras.

  Con respecto a $Avg(|Voc|)$, su valor desciende en el caso de los n-gramas a medida que crece el valor de n. Este dato también implica que es menos probable que coaparezcan rasgos compuestos por más palabras en distintos documentos, tal y como se asume en la hipótesis (RD1). No obstante, en el caso de los k-skip-n-gramas, el valor de $Avg(|Voc|)$ aumenta a medida que el valor de n es mayor debido a que hay un mayor número de combinaciones posibles en las que se pueden realizar los k saltos entre palabras. Este dato corrobora los resultados presen-

tados por Guthrie et al. [2006] y significa que la representación basada en k-skip-n-gramas es más costosa en tiempo y en espacio con respecto a la representación basada en n-gramas.

Finalmente, las NEs son capaces de representar un alto número de resultados de búsqueda y generar un tamaño de vocabulario pequeño. En particular, el tamaño de vocabulario generado por las NEs es menor con respecto al generado por los n-gramas compuestos por 1 o 2 palabras y los k-skip-n-gramas de cualquier longitud, aunque es mayor con respecto al generado por 3-gramas y 4-gramas compuestos por palabras en mayúsculas.

- **Algoritmos de *clustering*:** HAC mejora significativamente los resultados de AP para todos los rasgos considerados. La razón es que los resultados de HAC mostrados en la tabla se corresponden con el agrupamiento obtenido por este algoritmo usando el mejor umbral promedio de la colección, lo cuál puede ser visto como una cota superior de los resultados obtenidos por este algoritmo. No obstante, esto significa que aplicando HAC es posible obtener mejores agrupamientos de resultados de búsqueda con respecto a AP si se realiza una estimación conveniente del umbral. Estos datos corroboran las conclusiones de varios autores [Artiles, 2009; Balog et al., 2009; Berendsen, 2015] con respecto a que HAC es una elección adecuada a la hora de agrupar los resultados de búsqueda. El algoritmo AP no requiere datos de entrenamiento, pero obtiene pobres resultados al igual que otros algorimos empleados en este problema con características similares como por ejemplo *Fuzzy Ants* [Venkateshan, 2009] o el método de los k-vecinos estimando previamente de manera automáticamente el número de *clusters* [Lana-Serrano et al., 2010]. Por estos motivos, en lo sucesivo nos referiremos únicamente a los resultados obtenidos por HAC.

- **Valor del umbral γ:** la tabla muestra el mejor valor promedio del umbral γ para cada tipo de rasgo cuando se aplica HAC. En particular, los valores de los mejores umbrales para 1-gramas y NEs concuerdan con los obtenidos para esta colección por otros autores [Saggion, 2007; Elmacioglu et al., 2007] que han empleado este algoritmo. Los *clusters* resultantes se obtienen mediante el criterio $sim_C(C_i, C_j) > \gamma$, de modo que si γ es menor, entonces la condición de agrupamiento es más laxa. De acuerdo a los valores de la tabla, el mejor umbral γ desciende a medida que se consideran rasgos formados por un número mayor de palabras. Esto significa que cuanto mayor sea el valor de n, la coaparición de los rasgos implica que las páginas web hablan del mismo individuo con mayor posibilidad, lo cual concuerda con la hipótesis (RD1) y corrobora las conclusiones de algunos autores [Popescu y Magnini, 2007; Artiles et al., 2009a; Nuray-Turan et al., 2012]. En particular, para los 3-gramas en mayúsculas y para los rasgos compuestos por 4 palabras, los mejores

resultados se obtienen tomando el umbral $\gamma = 0$, lo cuál significa que la coaparición de un único rasgo es un criterio adecuado para decidir que las páginas web hablan del mismo individuo, dado que en dicho caso la similitud necesariamente será mayor que 0. Estos resultados indican que cuando los rasgos empleados se componen de un mayor número de palabras, el umbral que debe estimarse debe tener un valor más bajo. En particular, esta situación se cumple para los n-gramas e indica que la hipótesis (RD1) es acertada.

- **Valor de n:** a medida que se aumenta el valor de n, existe una mayor tendencia a que se obtengan mejores valores de precisión, pero peores valores de cobertura. Esto se explica porque el valor de R % desciende a medida que crece n, lo cual implica que los algoritmos generarán un mayor número de *clusters* unitarios. Como explicamos anteriormente, este tipo de *clusters* tienen un impacto positivo en los resultados de precisión, pero tienen un efecto negativo en los valores de cobertura. En particular, los resultados de cobertura descienden drásticamente cuando se toma el valor de $n = 4$, de modo que no es adecuado representar los resultados de búsqueda mediante rasgos compuestos por este número de palabras. Por otro lado, los 3-gramas obtienen mejores resultados que los 2-gramas y los 1-gramas, empleando un vocabulario más reducido, lo cual confirma las hipótesis (RD1) y (RD2). No obstante, el resultado de medida-F obtenido por los 1-gramas es comparable con respecto al obtenido por los 3-gramas compuestos por cualquier palabra o por palabras en mayúsculas. De acuerdo a los resultados de las métricas *B-Cubed*, podemos descartar los 2-gramas de los dos tipos y los 1-gramas en mayúsculas.

- **n-gramas y k-skip-n-gramas:** los k-skip-n-gramas tienen la desventaja de que generan un tamaño de vocabulario de al menos un orden de magnitud mayor con respecto a los n-gramas en la mayoría de los casos. Por otra parte, en la mayoría de los casos los k-skip-n-gramas obtienen resultados similares a los n-gramas, de modo que no son rasgos adecuados a la hora de representar los resultados de búsqueda. La única excepción se observa cuando $n = 4$, dado que los k-skip-4-gramas obtienen mejoras significativas con respecto a los 4-gramas. No obstante, los k-skip-4-gramas tienen resultados similares a los 3-gramas en mayúsculas pero estos últimos generan un tamaño de vocabulario mucho menor, de modo que son rasgos más adecuados. En conclusión, la representación mediante n-gramas es más eficiente con respecto a la representación mediante k-skip-n-gramas, de manera que en lo sucesivo no nos referiremos más a los resultados obtenidos por los k-skip-n-gramas.

- **Palabras en mayúsculas / Cualquier palabra:** las representaciones mediante cualquier tipo de palabra obtienen mejores resultados con respecto a aquellas basadas en palabras en mayúsculas, salvo en los casos de los n-gramas de longitudes 2 y

3. No obstante, anteriormente descartamos el uso de los 2-gramas de cualquier tipo y los 1-gramas en mayúscula debido a que obtienen peor rendimiento. En el caso de los 1-gramas de cualquier tipo de palabra, los resultados obtenidos son similares a los de los 3-gramas, por lo que no pueden descartarse. Por último, los 3-gramas en mayúsculas obtienen resultados similares a los obtenidos por los 3-gramas compuestos por cualquier tipo de palabra, de modo que el uso de los primeros es más eficiente computacionalmente. De este modo, descartamos los 3-gramas compuestos por cualquier tipo de palabra.

- **NEs:** pese a que las NEs obtienen peores resultados que los 3-gramas y la representación basada en 1-gramas, cuentan con la ventaja de representar un alto número de páginas web y al mismo tiempo generar un vocabulario pequeño, por lo que no pueden descartarse. Los resultados obtenidos mediante NEs se asemejan a los obtenidos por los *n*-gramas en mayúsculas compuestos por 1 o 2 palabras, lo cuál puede deberse a que un 82,28 % de estos rasgos se componen de este número de palabras, y la mayoría de ellas (99,62 %) se componen exclusivamente por palabras escritas en mayúsculas.

  En los anteriores análisis hemos descartado varios rasgos, quedándonos solamente con los 1-gramas, los 3-gramas en mayúsculas y las NEs. Los 1-gramas tienen la ventaja de que son los rasgos capaces de representar más resultados de búsqueda, pero cuentan con el inconveniente de que generan un vocabulario más amplio que los otros rasgos. En cambio, los 3-gramas en mayúsculas tienen la ventaja de que generan un vocabulario mucho más pequeño, pero representan un menor porcentaje de resultados de búsqueda. Por último, las NEs consiguen una buena relación entre páginas web representadas y tamaño de vocabulario, pero obtienen peores resultados con respecto a los 1-gramas y los 3-gramas en mayúsculas. A pesar de las posibles ventajas y desventajas de cada uno de estos rasgos, por el momento no se descartarán y se tendrán en cuenta en las siguientes propuestas que se presentarán en esta tesis doctoral.

- **Grado de ambigüedad de los resultados de búsqueda:** a continuación, estudiaremos la sensibilidad de los rasgos que no han sido descartados con respecto al grado de ambigüedad de los nombres de persona. Como se explicó anteriormente, solamente tiene sentido comparar el grado de ambigüedad entre los nombres de persona si se cuenta con un número de resultados de búsqueda similar para todos ellos. No obstante, la colección de entrenamiento WePS-1 se caracteriza porque muchos de sus nombres de personas tienen asociado un número diferente de resultados de búsqueda (ver apartado 3.3.1). En particular, esta colección se divide en 32 nombres de personas reutilizados del corpus Web03 [Mann, 2006] con un número diferente de resultados de búsqueda asociados y otros 17 nombres de

persona, recopilados por Artiles et al. [2007], que tienen asociado un número de resultados de búsqueda entre 98 y 100. Además, estos nombres de persona tienen diferentes grados de ambigüedad. En particular, las páginas web del nombre de persona *Allan Hanbury* solo se refieren a dos individuos diferentes, mientras que las del nombre de persona *Thomas Baker* se refieren a 60 individuos distintos. El resto de nombres de persona tienen un grado de ambigüedad entre los dos casos anteriores. Por tanto, se han seleccionado estos 17 nombres de persona para estudiar la sensibilidad con respecto al grado de ambigüedad de los 1-gramas, los 3-gramas en mayúsculas y las NEs.

La Figura 4.1 muestra la sensibilidad de los tres tipos de rasgos considerados con respecto al grado de ambigüedad de los nombres de persona seleccionados. En particular, el eje X representa el número de individuos distintos de cada nombre de persona, mientras que el eje Y muestra los valores de medida-F. Estos resultados son los obtenidos por el algoritmo HAC empleando el mejor umbral de similitud promedio para cada tipo de rasgo y empleando para todos ellos la misma función de pesado (TF-IDF) y la misma medida de similitud (coseno). Para medir la sensibilidad de los rasgos con respecto al grado de ambigüedad se ha calculado el *coeficiente de correlación de Pearson* entre el número de individuos diferentes y los resultados de medida-F de los tres tipos de rasgos, denotado como $\rho_{X,Y}$. En particular, esta correlación es $\rho_{X,Y} = -0.86$ y $\rho_{X,Y} = -0.84$ para los 1-gramas y las NEs respectivamente, lo cual indica que para estos dos tipos de rasgos existe una fuerte tendencia de que sus resultados varíen de acuerdo al grado de ambigüedad de los nombres de personas. En cambio, la correlación para los 3-gramas en mayúsculas es $\rho_{X,Y} = -0.39$, lo cual indica que existe una tendencia débil a que los resultados obtenidos por estos rasgos sean dependientes del grado de ambigüedad de los nombres de persona. En conclusión, los 3-gramas en mayúsculas son rasgos más adecuados a la hora de conseguir un sistema de desambiguación más robusto con respecto al grado de ambigüedad de los nombres de persona, lo cual supone una línea abierta en la desambiguación de los nombres de persona (ver sección 1.1.3) y un objetivo planteado en esta tesis doctoral.

Conclusiones

Las conclusiones que podemos extraer de los experimentos preliminares llevados a cabo en este apartado son las siguientes:

- Se ha corroborado que HAC es una buena elección para agrupar resultados de búsqueda en el problema de desambiguación de nombres de personas en la Web, tal y como han concluido varios autores [Artiles, 2009; Balog et al., 2009; Berendsen, 2015].

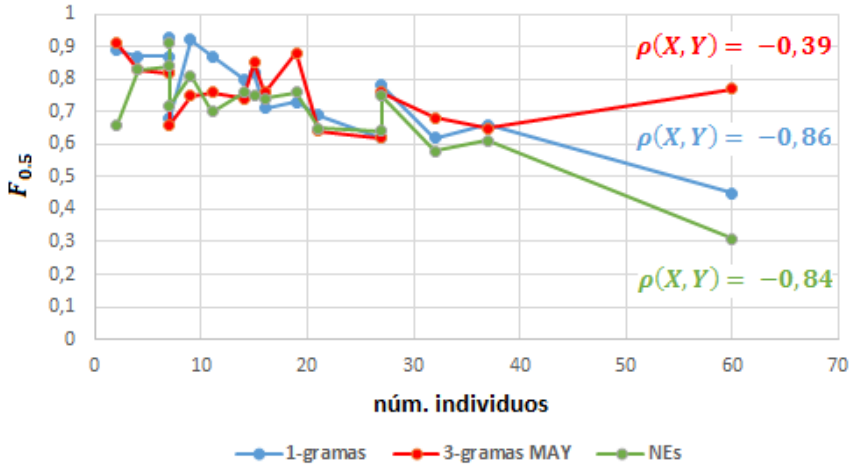

Figura 4.1: Sensibilidad de distintos tipos de rasgos con respecto al grado de ambigüedad de los nombres de persona.

- Los n-gramas son rasgos adecuados para representar el contenido de los resultados de búsqueda, dado que obtienen mejores resultados que otros tipos de rasgos como las NEs, y obtienen resultados similares a los k-skip-n-gramas sin generar un vocabulario tan extenso. Por un lado, este hecho confirma la hipótesis (RD1) y, por otro lado, determina que los k-skip-n-gramas son rasgos menos adecuados para representar los resultados de búsqueda.

- Se ha comprobado experimentalmente que es más probable que dos resultados de búsqueda hablen del mismo individuo cuando coaparecen entre ellos n-gramas compuestos por un mayor número de palabras entre sí, tal y como enuncia la hipótesis (RD1). En particular, cuando el valor de n crece, el mejor umbral promedio desciende, lo cuál significa que es necesario una menor coaparición de rasgos para decidir que dos resultados de búsqueda hablan de un mismo individuo.

- Las representaciones basadas en palabras en mayúsculas obtienen resultados similares con respecto a aquellas basadas en palabras de cualquier tipo para n-gramas compuestos por 2 y 3 palabras, a pesar de que las primeras generan un vocabulario más reducido. Esto significa que las palabras en mayúsculas capturan información relevante sobre los individuos tal y como se enuncia en la hipótesis (RD2).

- Los 3-gramas en mayúscula obtienen mejoras significativas con respecto a otros rasgos y son poco sensibles al grado de ambigüedad de los nombres de personas. Esto corrobora que la unión de las hipótesis (RD1) y (RD2) es acertada.

- Los 3-gramas en mayúscula tienen la desventaja de que no representan un alto porcentaje de resultados de búsqueda (concretamente, un 29,88 %). No obstante, esto significa que existe un margen de mejora si se emplean rasgos adicionales que

sean capaces de representar un mayor porcentaje de páginas web. En particular, varios de los mejores sistemas del estado del arte [Jiang et al., 2009; Yoshida et al., 2010] siguen una estrategia basada en dos fases, donde la primera consiste en obtener inicialmente *clusters* muy precisos a partir de rasgos poco comunes y, posteriormente, una segunda fase se encarga de refinarlos mediante rasgos más comunes. En particular, los 3-gramas en mayúsculas encajan como rasgos para obtener *clusters* iniciales precisos que pueden ser refinados con posterioridad.

Tras haber analizado que los 3-gramas en mayúsculas son adecuados para representar los resultados de búsqueda, la próxima sección presenta un nuevo algoritmo de *clustering* basado en el uso de estos rasgos. El algoritmo propuesto no requiere datos de entrenamiento con el fin de cumplir uno de los objetivos de esta tesis (ver sección 1.2). Por otro lado, este algoritmo puede verse como una fase inicial de agrupamiento cuyo objetivo consiste en obtener un conjunto inicial de *clusters* que obtengan un valor alto de precisión. Esto se debe a que los 3-gramas en mayúsculas se caracterizan por: (i) obtener resultados de medida-F competitivos y valores altos de precisión; y (ii) no ser capaces de representar un alto porcentaje de páginas web. Ambas razones implican que existe un margen de mejora tras el empleo de los 3-gramas en mayúsculas, puesto que las páginas web que no logran representar pueden representarse posteriormente mediante otros tipos de rasgos, de manera que puedan compararse con los *clusters* iniciales obtenidos a partir de una representación basada en 3-gramas en mayúsculas.

4.2. Algoritmo propuesto: *Unsupervised Person Name Disambiguator* (UPND)

Esta sección describe la primera propuesta de esta tesis doctoral para agrupar los resultados de búsqueda de acuerdo al individuo al que mencionan: el algoritmo de *clustering Unsupervised Person Name Disambiguator* (UPND). Este método se distingue de la mayoría de los sistemas de desambiguación de nombres de personas en la Web porque tiene las siguientes características: por un lado, es capaz de estimar el número de *clusters* de manera automática y, por otro lado, no necesita aprender ningún parámetro mediante datos de entrenamiento. UPND se basa en el concepto de *umbral adaptativo*, consistente en una función matemática que devuelve un cierto valor de umbral dependiente exclusivamente de las características de los sitios web que se comparan. En primer lugar, definiremos qué es un umbral adaptativo y, posteriormente, explicaremos en detalle el algoritmo de *clustering* UPND.

4.2.1. Umbrales adaptativos

La mayoría de los mejores sistemas del estado del arte agrupan los resultados de búsqueda mediante el algoritmo HAC. Como se explicó en el apartado 2.3.1, este algoritmo agrupa en cada paso el par de *clusters* con mayor similitud con respecto a una cierta política de enlace, de modo que en cada iteración se construye un nivel del dendrograma que genera en base al valor de similitud de los *clusters* agrupados. El número de *clusters* devuelto por HAC puede venir dado por varios criterios. Un posible criterio consiste en fijar el número de *clusters* k, de manera que se devuelve el nivel del dendrograma que contiene k *clusters*. No obstante, Heyl y Neumann [2007] son los únicos autores que han empleado este criterio, pero obtienen resultados bajos de medida-F. Salvo esta excepción, la mayoría de los métodos basados en HAC cortan el dendrograma a partir de un cierto valor de *umbral de similitud* $\gamma \in [0,1] \subset \mathbb{R}$. Este umbral define el mínimo nivel de similitud entre los *clusters* del dendrograma devuelto, de modo que los *clusters* resultantes se obtienen mediante la aplicación del criterio $sim_C(C_i, C_j) > \gamma$, siendo sim_C una cierta medida de similitud entre dos *clusters*. Los mejores sistemas de desambiguación obtienen un valor de umbral de similitud mediante datos de entrenamiento y, posteriormente, lo emplean para todos los nombres de persona incluidos en las colecciones de test.

Artiles [2009] concluyó que el rendimiento de estos sistemas depende fuertemente del grado de ambigüedad de los nombres de persona, puesto que el valor del umbral de similitud γ determina el número de *clusters* devuelto por el algoritmo. En particular, varios autores [Lefever et al., 2009; Long y Shi, 2010; Xu et al., 2015] muestran que pequeñas variaciones del valor γ implican diferencias significativas en los resultados obtenidos por HAC. Esto significa que el valor γ está sesgado por los nombres de persona pertenecientes a la colección de entrenamiento, y puede no ser adecuado para otros nombres de persona con distintas características. Para la ilustrar esta situación, la Figura 4.2 muestra cómo varían los valores promedios de las métricas *B-Cubed* en la colección de entrenamiento de WePS-1 con respecto al valor del umbral de similitud, empleando una representación mediante 1-gramas pesados con la función TF-IDF y utilizando la similitud coseno para comparar los resultados de búsqueda.

La gráfica muestra que el valor del umbral de similitud γ tiene un alto impacto en los resultados obtenidos por HAC. Cuando el valor es bajo, se obtienen valores altos de cobertura y pobres resultados de precisión, dado que la condición de agrupamiento es laxa y se agrupan más resultados de búsqueda. Por tanto, un valor pequeño de γ es adecuado para nombres de persona poco ambiguos, puesto que tienen asociados un menor número de *clusters*. En cambio, cuando el valor del umbral de similitud es alto, se obtienen altos valores de precisión y pobres resultados de cobertura, puesto que la

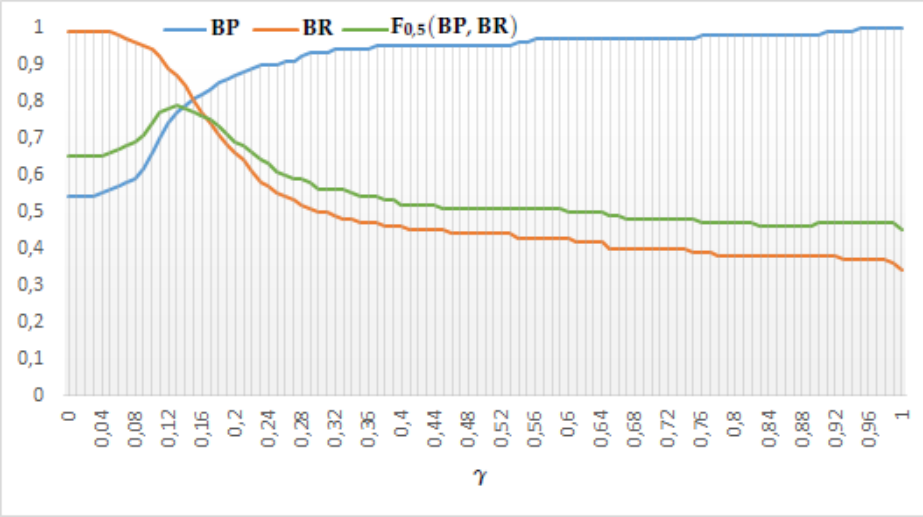

Figura 4.2: Variación de los valores de las métricas *B-Cubed* obtenidas por el algoritmo HAC con respecto al valor de umbral de similitud γ para la colección de entrenamiento de WePS-1.

condición de agrupamiento es estricta y se agrupa un menor número de resultados de búsqueda entre sí. En particular, un valor alto de γ es adecuado para nombres de persona muy ambiguos.

Por otra parte, de acuerdo con Artiles et al. [2009b], cuando se toma el mejor valor γ para cada nombre de persona, se obtienen mejoras significativas en los resultados con respecto a los mejores sistemas de desambiguación. Esto significa que existe un margen de mejora en los resultados con respecto a aplicar la política de los mejores sistemas del estado del arte, consistente en prefijar el mismo valor de umbral γ para cualquier nombre de persona. Para corroborar esta afirmación se ha tomado la colección de entrenamiento de WePS-1 comparando los resultados obtenidos por HAC cuando se aplica el mejor umbral promedio de la colección con respecto a emplear el mejor umbral para cada nombre de persona. Para ello, se han representado las páginas web mediante 1-gramas pesados mediante la función TF-IDF y se ha empleado la similitud coseno para comparar los resultados de búsqueda. La Tabla 4.2 muestra los resultados obtenidos por ambas políticas, donde el experimento marcado con • indica que obtiene mejoras significativas con respecto al otro.

Política HAC	γ	**BP**	**BR**	$\mathbf{F_{0,5}}$
HAC MEJOR γ PROMEDIO COLECCIÓN	0.13	0.78	0.87	0.79
HAC MEJOR γ POR NOMBRE DE PERSONA	-	0.89	0.91	0.90 •

Tabla 4.2: Resultados obtenidos por HAC aplicando el mejor umbral promedio y el mejor umbral para cada nombre de persona sobre la colección de entrenamiento de WePS-1.

La tabla muestra que se obtienen mejoras significativas cuando se toma el mejor

umbral de cada nombre de persona en lugar de tomar el mejor umbral promedio de toda la colección. Esto significa que resulta adecuado estimar un valor de umbral distinto para cada nombre de persona en lugar de aplicar el mismo umbral para todos ellos.

Siguiendo la idea de estimar diferentes umbrales para cada nombre de persona, en este trabajo se propone estimar automáticamente el valor de estos umbrales para cada comparación entre los resultados de búsqueda, de modo que dependan exclusivamente de las características de las páginas web que se comparan. Esta idea se formaliza en el concepto de *umbral adaptativo* que se presenta en este apartado.

Antes de definir qué es un umbral adaptativo, presentaremos previamente algunas notaciones. Denotaremos mediante $x \uparrow\uparrow$ y $x \downarrow\downarrow$ que el valor del número x se incrementa y decrece respectivamente, y usaremos la notación $x \gg y$ para expresar que el valor del número x es mucho mayor que el del número y. Por otro lado, dado un corpus $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$, denotaremos por F_i^n al conjunto de rasgos del documento D_i donde $n \in \mathbb{N}$ denota el número de palabras de las que se compone cada rasgo de F_i^n.

La Definición 4.1 presenta el concepto de umbral adaptativo:

Definición 4.1. Sea $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$ un corpus. Diremos que un **umbral adaptativo** es una función $\gamma^n : \mathcal{D} \times \mathcal{D} \to [0, 1]$ tal que cumple las siguientes propiedades:

- (UA1) $\forall D_i, D_j \in \mathcal{D} : |F_i^n \cap F_j^n| \uparrow\uparrow \Rightarrow \gamma^n(D_i, D_j) \downarrow\downarrow$
- (UA2) $n \uparrow\uparrow \Rightarrow \gamma^n(D_i, D_j) \downarrow\downarrow$
- (UA3) $|F_i^n| \gg |F_j^n| \vee |F_j^n| \gg |F_i^n| \Rightarrow \gamma^n(D_i, D_j) \downarrow\downarrow$

Los umbrales adaptativos son funciones matemáticas que comparan documentos entre sí con el fin de obtener un valor de umbral en el intervalo $[0, 1] \subset \mathbb{R}$ que dependa exclusivamente de las características de los documentos que se comparan. En particular, estas características consisten en el número de rasgos que comparten entre sí (propiedad UA1), el número de palabras de las que se componen dichos rasgos (propiedad UA2), y el número de rasgos de cada documento (propiedad UA3). De este modo, la condición de agrupamiento entre documentos definido por el umbral adaptativo viene dado por $sim^n(D_i, D_j) > \gamma^n(D_i, D_j)$, siendo sim^n una medida de similitud sobre los rasgos F_i^n.

Los umbrales adaptativos asumen que los documentos se representan mediante rasgos que se componen de un mismo número de palabras n. La razón es que se sabe de antemano que dos rasgos que no estén compuestos por el mismo número de palabras son distintos, por lo que no tiene sentido comparar documentos representados mediante rasgos con diferentes valores de n. La propiedad (UA1) exige que el umbral adaptativo será menor cuantos más rasgos compartan los documentos. Dicho de otro modo, la función de umbral adaptativo es monótona decreciente con respecto al número de rasgos

compartidos por los documentos que se comparan. La propiedad (UA2) cumple formalmente la hipótesis (RD1), puesto los valores del umbral adaptativo serán menores según sea el número de palabras n de las que se componen los rasgos. Esto quiere decir que el umbral adaptativo es una función monótona decreciente con respecto a n. Por último, la propiedad (UA3) indica que el umbral adaptativo debe ser menor cuando hay diferencias entre los tamaños de los documentos. En estos casos, la representación mediante VSM penaliza las comparaciones entre documentos cortos con respecto a documentos grandes porque los valores de similitud son muy bajos. De esta manera, la función de umbral adaptativo trata de atenuar estas penalizaciones.

La función de umbral adaptativo que se propone es la siguiente:

$$\gamma^n(D_i, D_j) = \frac{\gamma^n_{min}(D_i, D_j) + \gamma^n_{max}(D_i, D_j)}{2 \cdot n} \tag{4.7}$$

donde γ^n_{min} y γ^n_{max} se definen de la siguiente manera:

$$\gamma^n_{min}(D_i, D_j) = \begin{cases} 1 & si \quad F^n_i \cap F^n_j = \emptyset \\ \frac{\text{mín}(|F^n_i|, |F^n_j|) - |F^n_i \cap F^n_j|}{\text{mín}(|F^n_i|, |F^n_j|)} & si \quad F^n_i \cap F^n_j \neq \emptyset \end{cases} \tag{4.8}$$

$$\gamma^n_{max}(D_i, D_j) = \begin{cases} 1 & si \quad F^n_i \cap F^n_j = \emptyset \\ \frac{\text{mín}(|F^n_i|, |F^n_j|) - |F^n_i \cap F^n_j|}{\text{máx}(|F^n_i|, |F^n_j|)} & si \quad F^n_i \cap F^n_j \neq \emptyset \end{cases} \tag{4.9}$$

La función γ^n cumple la propiedad (UA1), dado que en las funciones γ^n_{min} y γ^n_{max} aparece restando en el numerador el número de rasgos compartidos entre los documentos $|F^n_i \cap F^n_j|$. γ^n cumple trivialmente la propiedad (UA2), puesto que el número de palabras n de la que se componen los rasgos aparece en el denominador. Finalmente, la función cumple la propiedad (UA3), ya que el numerador de las funciones γ^n_{min} y γ^n_{max} depende del tamaño menor de los documentos y, por otro lado, en la función γ^n_{max} aparece el máximo de los tamaños en el denominador, de modo que cuanto mayor sea la diferencia entre los tamaños, menor será el factor $\frac{\text{mín}(|F^n_i|, |F^n_j|)}{\text{máx}(|F^n_i|, |F^n_j|)}$ y, por tanto, menor será el valor obtenido por el umbral adaptativo γ^n.

4.2.2. Algoritmo UPND

A continuación, se describirá la primera propuesta de esta tesis en relación al agrupamiento de resultados de búsqueda de acuerdo a los individuos que mencionan: el algoritmo de *clustering* UPND. Este algoritmo emplea un umbral adaptativo para comparar los resultados de búsqueda, de modo que no necesita datos de entrenamiento para

prefijar el valor de ningún parámetro. Además, UPND es capaz de estimar el número de *clusters*, por lo que tampoco requiere conocer este dato a priori.

Formalización del problema

El problema de la desambiguación de nombres de personas en el dominio Web puede formalizarse de la siguiente manera: sea $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$ el ranking de resultados devuelto por un motor de búsqueda al consultar un nombre de persona NP. Los resultados de búsqueda contienen un conjunto de menciones a individuos que comparten el nombre NP, denotado como $E = \{e_1, e_2, \ldots, e_k\}$. A su vez, cada resultado de búsqueda $W_i \in \mathcal{W}$ puede mencionar a un subconjunto de estos individuos denotado $E_i = \{e_1^i, e_2^i, \ldots, e_{n_i}^i\} \subseteq E$. Nótese que $E_i = \emptyset$ implica que el resultado de búsqueda W_i no habla de ningún individuo de nombre NP. El objetivo consiste en obtener un agrupamiento $\mathcal{C}$ de los resultados de búsqueda de tal manera que cada grupo contiene todos los resultados de búsqueda en los que se menciona a un cierto individuo, y no contiene ningún otro resultado de búsqueda donde no se le mencione. Formalmente, se pide obtener $\mathcal{C} \in \mathcal{P}(\mathcal{W})$ tal que cumpla $\forall C_i \in \mathcal{C}\ \exists e_j \in E\ \forall W_p \in C_i : e_j \in E_p \wedge \nexists W_r \in \mathcal{C} \setminus C_j : e_j \in E_r$. Esto es, todos los resultados de búsqueda de un cierto grupo (*cluster*) mencionan a un mismo individuo llamado NP, y no existe ningún resultado de búsqueda fuera del grupo en el que también se le mencione.

De acuerdo a la descripción anterior, pueden existir grupos distintos $C_i, C_j \in \mathcal{C}$ tales que $C_i \cap C_j \neq \emptyset$. Esta situación se presenta más comúnmente cuando hay páginas web consistentes en listados de individuos con el mismo nombre (ej. páginas de desambiguación de Wikipedia), páginas web que tratan la genealogía de una familia en la que varios miembros se llaman de la misma manera, o en casos en los que varias celebridades relacionadas entre sí comparten el mismo nombre y se les menciona en varias páginas web, como sucede con los ex-presidentes de EEUU *George H. W. Bush* y *George W. Bush*. Esta situación fue tomada en cuenta por los organizadores de las campañas WePS, de modo que aquellas páginas web donde se hablase de varios individuos con el mismo nombre fueron anotadas en los *clusters* correspondientes a los diferentes individuos. No obstante, de acuerdo con Artiles [2009], este fenómeno no juega un gran papel en el problema.

En general, los sistemas del estado del arte han simplificado el problema asumiendo el criterio de que *cada resultado de búsqueda menciona a un único individuo llamado NP* (*one person per document*) de un modo análogo a la política *one sense per discourse* (*un sentido por discurso*) en el problema WSD [Gale et al., 1992]. Por otro lado, bajo este criterio también se asume que cualquier página web devuelta por el buscador mencionará a algún individuo llamado como la consulta, aunque esta situación no tiene por qué ser cierta. Por ejemplo, si se consulta el nombre de persona *Javier Martínez*, algunos posibles resultados devueltos podrían hablar de unos individuos llamados *Javier Pérez* y *Luis*

Martínez, pero no mencionar a ningún individuo llamado *Javier Martínez*.

Partición del ranking de resultados de búsqueda

El algoritmo UPND también asume el criterio *one person per document*, lo cual permite formalizar matemáticamente la desambiguación de nombres de personas en la Web como la obtención de una partición del ranking de resultados devueltos por el motor de búsqueda. A continuación, se presentan una serie de conceptos matemáticos para explicar el algoritmo UPND:

Definición 4.2. Se dice que R es una **relación n-ária** sobre los conjuntos $A_1, A_2, \ldots, A_n$ si $R \subseteq A_1 \times A_2 \times \ldots \times A_n$. Si $A = A_1 = A_2 = \ldots = A_n$, se dice que R es una **relación n-ária sobre el conjunto A**.

Dado un conjunto A, se dice que $R \subseteq A \times A$ es una *relación binaria* sobre el conjunto A. Dados dos elementos $a, a' \in A$, se dice que a y a' *están relacionados* por la relación R si y sólo si $(a, a') \in R$, lo cual suele denotarse como $R(a, a')$ o aRa'. En particular, suelen resultar de interés las denominadas *relaciones de equivalencia*:

Definición 4.3. Sea $\sim \subseteq A \times A$ una relación binaria sobre un conjunto $A \neq \emptyset$. Se dice que $\sim$ es una **relación de equivalencia** si cumple las siguientes propiedades:

- Reflexividad: $\forall a \in A : a \sim a$
- Simetría: $\forall a, a' \in A : a \sim a' \Leftrightarrow a' \sim a$
- Transitividad: $\forall a, a', a'' \in A : a \sim a' \wedge a' \sim a'' \Rightarrow a \sim a''$

Una relación de equivalencia $\sim$ sobre un conjunto A define *clases de equivalencia módulo* $\sim$, de modo que para cualquier elemento $a \in A$, su clase de equivalencia se define y denota como $[a]_\sim = \{a' \in A | a' \sim a\}$, y se dice que a es un *representante de la clase de equivalencia* $[a]_\sim$. El conjunto de todas las clases de equivalencia módulo $\sim$ se denomina *conjunto cociente* y se denota y define como $A/\sim = \{[a]_\sim | a \in A\}$. El cardinal de este conjunto se denomina *orden de la relación de equivalencia*, $k = |A/\sim|$.

El concepto de *partición de un conjunto* es relevante en el problema, puesto que al asumir que cada resultado de búsqueda menciona a un único individuo llamado como la consulta, entonces el conjunto de *clusters* que debe obtenerse consiste en una partición del ranking de páginas web W devuelto por el motor de búsqueda tras consultar el nombre de persona.

Definición 4.4. Se dice que una **partición de un conjunto** A es una familia $P = \{A_i \mid i \in I\}$ de subconjuntos de A tal que cumple las siguientes propiedades:

- $\forall i \in I : A_i \subseteq A \wedge A_i \neq \emptyset$
- $\forall i, j \in I, i \neq j : A_i \cap A_j = \emptyset$
- $A = \bigcup_{i \in I} A_i$

donde I es un conjunto de índices que distinguen a cada conjunto A_i.

La primera propiedad exige que los conjuntos pertenecientes a una partición sean no vacíos. Por otro lado, la segunda propiedad exige que todos ellos sean disjuntos entre sí. Finalmente, la última propiedad establece que el conjunto A puede obtenerse mediante la unión de los subconjuntos pertenecientes a la partición, lo cual implica que cualquier elemento del conjunto A debe pertenecer a alguna de sus divisiones pertenecientes a la partición, i.e., $\forall a \in A \; \exists i \in I : a \in A$. Es más, dado que todas las divisiones son disjuntas entre sí, todo elemento $a \in A$ pertenecerá a una única división del conjunto. Por otro lado, dada una partición $P = \{A_i \mid i \in I\}$ sobre un conjunto A, puede definirse la siguiente relación binaria: $R_P = \{(a, a') \in A \times A | \exists i \in I : a, a' \in A_i\}$.

El *Teorema Fundamental de las Relaciones de Equivalencia* asocia los conceptos de relación de equivalencia y partición de un conjunto:

Teorema 4.1. ***(Teorema Fundamental de las Relaciones de Equivalencia)****: sea A un conjunto. Las siguientes afirmaciones son ciertas:*

1. *Si P es una partición de A entonces R_P es una relación de equivalencia, i.e., toda partición define una relación de equivalencia.*
2. *Si $\sim$ es una relación de equivalencia sobre A entonces $A/\sim$ es una partición de A, i.e., toda relación de equivalencia define una partición.*

La demostración de este resultado puede encontrarse en libros sobre teoría de conjuntos (ej. [Hrbacek y Jech, 1999]). El teorema implica que una manera de encontrar una partición de un conjunto consiste en definir una relación de equivalencia sobre él. El algoritmo UPND obtiene una partición del ranking de resultados de búsqueda a partir de la construcción de una relación de equivalencia entre las páginas web. En particular, UPND emplea la relación entre resultados de búsqueda definida por el criterio de agrupamiento mediante umbrales adaptativos:

$$R^n(W_i, W_j) \equiv sim^n(W_i, W_j) > \gamma^n(W_i, W_j) \tag{4.10}$$

donde *sim* es una medida de similitud y γ es una función de umbral adaptativo.

A continuación, se comprueba si $R^n \subseteq \mathcal{W} \times \mathcal{W}$ es una relación de equivalencia: R^n es reflexiva puesto que, por un lado, para cualesquiera $n \in \mathbb{N}$ y $W_i \in \mathcal{W}$ se cumple

que $sim^n(W_i, W_i) = 1$ por la definición de medida de similitud (ver Definición 3.1) y, por otro lado, $\gamma^n(W_i, W_i) = 0$ puesto que mín$(|F_i^n|, |F_i^n|) - |F_i^n \cap F_i^n| = |F_i^n| - |F_i^n| = 0$, tomando la función de umbral adaptativo presentada anteriormente. Por otro lado, R^n también cumple la propiedad simétrica puesto que, por un lado, cualquier medida de similitud *sim* la cumple por definición y, por otro lado, la función de umbral adaptativo también es simétrica porque los operadores *min*, *max* son simétricos y las operaciónes $\cap$ y $\cdot$ (producto de escales) son conmutativas. No obstante, no se puede asegurar que la relación R^n sea transitiva dado que esta propiedad no la garantiza ninguna medida de similitud ni ninguna función de umbral adaptativo, por lo que no está garantizado que R^n sea una relación de equivalencia.

Para poder obtener una relación de equivalencia a partir de R^n, se toma el concepto de *cierre transitivo de una relación* que pasamos a definir a continuación:

Definición 4.5. Se dice que el **cierre transitivo (o clausura transitiva) de una relación** binaria $R \subseteq A \times A$ sobre un conjunto A es una relación $R_C \subseteq A \times A$ tal que cumple las siguientes propiedades:

1. $R \subseteq R_C$
2. R_C es transitiva
3. $\forall R' \subset A \times A$ tal que R' es transitiva: $R \subseteq R' \Rightarrow R_C \subseteq R'$

La idea intuitiva es que el cierre transitivo de una relación R es la menor relación transitiva que contiene a R. La tercera propiedad expresa que cualquier otra relación transitiva que contenga a R contiene al menos los mismos elementos que su cierre transitivo. La existencia del cierre transitivo de cualquier relación R está garantizada y puede caracterizarse de la siguiente manera:

$$R_C(a, a') \equiv \exists b_1, b_2, \ldots b_m \in A : aR_Cb_1 \wedge b_1R_Cb_2 \wedge \ldots \wedge b_mR_Ca' \tag{4.11}$$

Dada una relación reflexiva y simétrica R, su cierre transitivo R_C es también reflexivo y simétrico, puesto que por definición $R \subseteq R_C$, y además, también por definición, R_C es transitivo, por lo que se trata de una relación de equivalencia. El algoritmo UPND obtiene una partición del ranking de resultados $\mathcal{W}$ devuelto por el motor de búsqueda a partir del cierre transitivo de la relación R^n, y se define de la siguiente manera:

$$W_i \sim_{UPND} W_j \equiv \exists W_1', W_2', \ldots W_m' \in \mathcal{W} : R^n(W_i, W_1') \wedge R^n(W_1', W_2') \wedge \ldots \wedge R^n(W_m', W_j) \tag{4.12}$$

donde $0 \leq m \leq |\mathcal{W}|$.

El algoritmo UPND calcula la partición del ranking de páginas web $\mathcal{W}$ a partir de la relación de equivalencia $\sim_{UPND}$, esto es $\mathcal{W}/\sim_{UPND}$. Cada clase de equivalencia obtenida por UPND se corresponderá con un subconjunto de resultados de búsqueda que idealmente mencionarán al mismo individuo, y el número de *clusters* devuelto por el algoritmo se corresponderá con el orden de la relación de equivalencia $\sim_{UPND}$, $k = |\mathcal{W}/\sim_{UPND}|$, por lo que UPND no necesita conocer este dato de antemano, a diferencia de otros algoritmos de partición como los métodos de las *k*-medias y los *k*-vecinos.

Pseudocódigo

El Algoritmo 4.2 muestra el pseudocódigo del algoritmo UPND. El algoritmo recibe como entrada el ranking de resultados $\mathcal{W}$ devuelto por el motor de búsqueda al consultar un nombre de persona NP, una medida de similitud sim, una función de umbral adaptativo γ, y un parámetro n que indica el número de palabras contenidas por los n-gramas con los que se representan los resultados de búsqueda. UPND devuelve como salida un conjunto de *clusters* $\mathcal{C}$ formados por elementos de $\mathcal{W}$. Inicialmente, UPND asigna un *cluster* a cada resultado de búsqueda (líneas 1-4), por lo que se trata de un algoritmo de *clustering* aglomerativo. A continuación, se comparan aquellos pares de resultados de búsqueda que no pertenecen a un mismo *cluster* mediante el criterio de agrupamiento $sim^n(W_i, W_j) > \gamma^n(W_i, W_j)$, de modo que en caso de cumplirse esta condición, se agrupan los *clusters* a los que pertenecen ambos documentos (líneas 5-14). Para detectar si dos resultados de búsqueda pertenecen a un mismo *cluster* se emplea el predicado booleano $sameCluster(W_i, W_j) \equiv \exists C_i \in \mathcal{C} : W_i, W_j \in C_i$. Nótese que el bucle de la línea 6 no recorre todos los documentos para evitar comparaciones innecesarias, debido a que tanto la medida de similitud como el umbral adaptativo son funciones simétricas. Finalmente, el algoritmo devuelve el conjunto de *clusters* resultantes $\mathcal{C}$ (línea 15).

El algoritmo UPND cumple las siguientes propiedades:

- **Determinismo:** UPND devuelve el cierre transitivo de una relación, que por definición es único, por lo que se asegura que el algoritmo computa siempre la misma partición para los mismos datos de entrada, i.e., es determinista. Además, dado que, en particular, la relación $\sim_{UPND}$ es simétrica, se garantiza que UPND siempre devuelve la misma partición con independencia del orden de las páginas web en el ranking de resultados de búsqueda.

- **Complejidad:** dado que los bucles de las líneas 6 y 7 recorren el ranking de resultados de búsqueda, el algoritmo UPND tiene una complejidad del orden de $\mathcal{O}(N^2)$, al igual que HAC bajo determinadas condiciones.

Algoritmo 4.2 UPND($\mathcal{W}, sim, \gamma, n, m$).

Entrada: Ranking de resultados de búsqueda $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$, medida de similitud sim, umbral adaptativo γ, $n \in \mathbb{N}$.

Salida: Conjunto de *clusters* $\mathcal{C}$.

para $i = 1$ **hasta** N **hacer**
  $C_i = \{W_i\}$
fin para
$\mathcal{C} = \bigcup_{i=1}^{N} C_i$
para $i = 1$ **hasta** N **hacer**
  para $j = i + 1$ **hasta** N **hacer**
    si $\neg sameCluster(W_i, W_j)$
      si $sim^n(W_i, W_j) > \gamma^n(W_i, W_j)$
        $C_i = C_i \cup C_j$
        $\mathcal{C} = \mathcal{C} \setminus \{C_j\}$
      fin si
    fin si
  fin para
fin para
devolver $\mathcal{C}$

4.3. Resultados y discusión

En este apartado presentaremos los resultados obtenidos por el algoritmo UPND utilizando la representación mediante 3-gramas en mayúsculas de las páginas web. En primer lugar, compararemos los resultados de UPND con respecto al algoritmo HAC. Posteriormente, analizaremos el impacto de distintas funciones de pesado de términos y medidas de similitud en los resultados obtenidos por UPND. Por último, compararemos sus resultados con respecto a los resultados de los sistemas del estado del arte en las tres colecciones de test proporcionadas por las campañas de evaluación WePS.

4.3.1. Comparativa entre HAC y UPND

La Tabla 4.3 muestra los resultados de UPND con respecto a HAC sobre la colección de entrenamiento de la campaña de evaluación WePS-1 usando 3-gramas en mayúsculas. Se han seleccionado estos rasgos tras haber concluido que son adecuados en el apartado 4.1.2. Al igual que en los experimentos preliminares, los rasgos se han pesado mediante la función TF-IDF y la proximidad entre los resultados de búsqueda se calcula mediante la medida de similitud coseno. Se han descartado los resultados obtenidos por AP debido a que presenta un peor rendimiento con respecto a HAC. En el caso de HAC,

los resultados se corresponden con los obtenidos mediante el mejor umbral de similitud promedio, $\gamma = 0$ en el caso de los 3-gramas en mayúsculas. Este umbral se ha estimado previamente, de modo que este experimento puede verse como una cota superior de la política empleada por los mejores sistemas del estado del arte que emplean HAC. En cambio, los resultados de UPND son siempre los mismos dado que se basa en la comparación de documentos mediante un umbral adaptativo.

Algoritmo	**BP**	**BR**	$\mathbf{F_{0,5}}$
UPND	0.92	0.72	0.79
HAC	0.89	0.76	0.80

Tabla 4.3: Resultados obtenidos por los algoritmos HAC y UPND para la colección de entrenamiento de WePS-1 utilizando 3-gramas en mayúsculas.

De acuerdo al test de significancia estadística, los resultados de UPND y HAC son similares, a pesar de que HAC emplea el mejor umbral de similitud promedio. Esto corrobora que las propiedades de los umbrales adaptativos son adecuadas, y pueden dar lugar a resultados similares a los obtenidos por el mejor umbral promedio del algoritmo HAC. Por otro lado, UPND presenta mejores resultados de precisión y peores resultados de cobertura con respecto a HAC. Esto se explica porque el umbral de similitud empleado por HAC es $\gamma = 0$, lo cuál quiere decir que se agrupan todas las páginas web que compartan al menos un rasgo, dado que, en dicho caso, la similitud entre ambas será superior a 0. En cambio, las agrupaciones de UPND vienen determinadas por el umbral adaptativo (ver fórmula 4.7), el cuál depende del número de rasgos que coaparecen entre los resultados de búsqueda que se comparan.

Por otro lado, dado que en este experimento los algoritmos HAC y UPND emplean la misma configuración, se puede estudiar cuál es la sensibilidad de ambos con respecto al grado de ambigüedad de los nombres de persona. Para ello, se han tomado los 17 nombres de personas de la colección de entrenamiento de WePS-1 que se emplearon para hacer este mismo estudio para distintos tipos de rasgos. La Figura 4.3 muestra la variación de los resultados de los dos algoritmos con respecto al grado de ambigüedad de los nombres de persona. El comportamiento de los dos algoritmos es similar para los nombres de persona seleccionados, a pesar de que HAC emplea el mejor umbral de similitud promedio y UPND no necesita aprender dicho parámetro. La sensibilidad de ambos algoritmos con respecto al grado de ambigüedad se ha calculado mediante el coeficiente de correlación de Pearson. Esta correlación es $\rho_{X,Y} = -0.39$ en el caso de HAC, mientras que es $\rho_{X,Y} = -0.30$ en el caso de UPND, lo cual significa que UPND es más independiente del grado de ambigüedad de los nombres de persona que HAC.

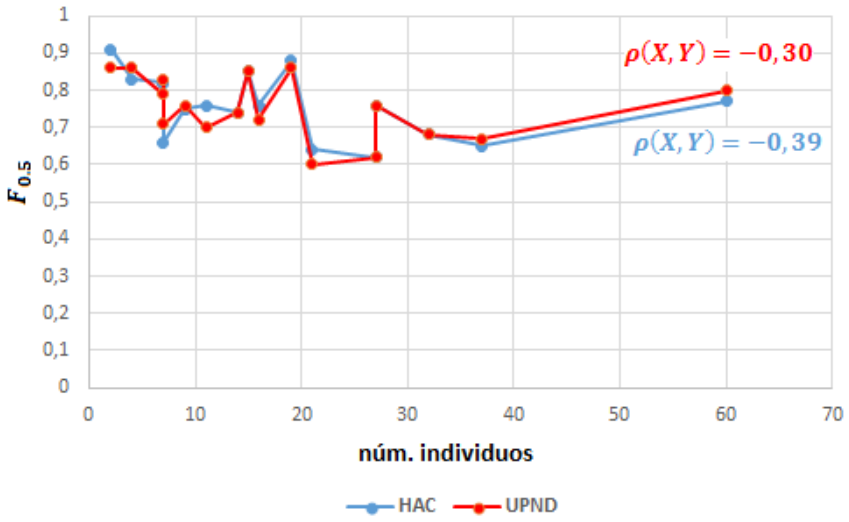

Figura 4.3: Sensibilidad de los algoritmos HAC y UPND con respecto al grado de ambigüedad de los nombres de persona.

4.3.2. Estudio de la configuración de UPND

En este apartado se evaluará UPND utilizando diferentes funciones de pesado y medidas de similitud con la colección de entrenamiento de WePS-1.

La Tabla 4.4 muestra los resultados de UPND empleando las funciones descritas en el apartado 3.1.3: pesado binario (*Bin*), TF, TF-IDF y *z*-score. En cuanto a las medidas de similitud, se emplean coseno (*Cos*) y Jaccard pesado (*WJ*), descritas en el apartado 3.1.4. El estudio de la significancia estadística se ha dividido en dos partes:

- **Misma función de pesado / Distinta medida de similitud:** si un experimento contiene la marca † significa que el empleo de la correspondiente medida de similitud obtiene mejoras significativas con respecto a usar la otra medida de similitud.
- **Distinta función de pesado / Misma medida de similitud:** de acuerdo con el test de significancia estadística, cuando se emplea la misma medida de similitud no hay mejoras significativas a pesar de que se empleen diferentes funciones de pesado de términos. Por esta razón, la tabla no incluye ningún símbolo al respecto.

La tabla muestra que utilizando la similitud coseno se obtienen mejoras significativas con respecto a usar Jaccard pesado. En particular, empleando Jaccard pesado se obtiene una ligera mejora del resultado de precisión, pero empeora los resultados de cobertura, lo cual sucede porque UPND es capaz de agrupar menos resultados de búsqueda cuando se emplea esta función. Los motivos son los siguientes:

- La similitud coseno normaliza los pesos en el denominador, de modo que evita

WePS-1 (entrenamiento)	Cos			WJ		
Función de pesado	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
Bin	0.91	0.73	0.80 †	0.92	0.69	0.77
TF	0.92	0.72	0.79 †	0.92	0.69	0.77
TF-IDF	0.92	0.72	0.79 †	0.92	0.68	0.76
z-score	0.91	0.72	0.79 †	0.92	0.66	0.75

Tabla 4.4: Resultados obtenidos por el algoritmo UPND sobre la colección de entrenamiento de WePS-1 empleando distintas funciones de pesado de términos y medidas de similitud.

predisposiciones hacia documentos representados con un mayor número de rasgos con respecto a Jaccard pesado.

- La función Jaccard pesado se basa en la intersección de rasgos entre los documentos, al igual que la función del umbral adaptativo (ver ecuaciones 4.8 y 4.9), de modo que los valores de la similitud y el umbral adaptativo tienden a ser similares, lo cuál produce que se cumpla en menor medida la condición de agrupamiento entre resultados de búsqueda.

Por tanto, la similitud coseno es una opción adecuada a la hora de medir la cercanía entre resultados de búsqueda para el algoritmo UPND y en lo sucesivo se utilizará en los experimentos llevados a cabo sobre UPND en este trabajo.

Por otro lado, la función de pesado de términos no tiene impacto en los resultados cuando se emplean 3-gramas en mayúsculas. Esto se debe a que la representación mediante estos rasgos genera un vocabulario muy pequeño (ver Tabla 4.1), de modo que la mera coaparición de pocos rasgos de este tipo entre resultados de búsqueda diferentes tiene como consecuencia que UPND los agrupe en el mismo *cluster*, pese a que tengan asignados diferentes pesos. Esto significa que las funciones de pesado locales (*Bin* y TF) son una buena alternativa para pesar los 3-gramas en mayúsculas, dado que son menos costosas computacionalmente, porque su cómputo solo depende de las apariciones de los rasgos en cada documento, a diferencia de las funciones globales (TF-IDF y z-score). En lo sucesivo, emplearemos la función binaria para pesar este tipo de rasgos puesto que no requiere hacer un recuento de apariciones de cada rasgo a diferencia de TF.

4.3.3. Comparativa con otros sistemas

En este apartado se presentan los resultados obtenidos por UPND con las colecciones de test proporcionadas por las tres campañas de evaluación WePS [Artiles et al., 2007, 2009b, 2010] empleando la configuración explicada anteriormente: 3-gramas en mayúsculas, función de pesado binaria y similitud coseno.

Los resultados del algoritmo UPND se compararán con respecto a los obtenidos por los participantes de estas campañas y con los sistemas de desambiguación de nombres de personas más recientes. En particular, nos centraremos en los sistemas que obtienen los mejores resultados, y en aquellos métodos que, al igual que UPND, no requieren datos de entrenamiento. Adicionalmente, se compararán los resultados de UPND con respecto a algunos *baselines*. Los *baselines* son sistemas cuyos resultados sirven como referencia a la hora de medir el rendimiento de nuevas propuestas con respecto a un determinado problema. En el caso particular de la desambiguación de nombres de persona, emplearemos los *baselines* propuestos por los organizadores de las campañas WePS, basados en casos extremos de agrupamiento de los resultados de búsqueda e independientes de su representación:

- **ONE IN ONE:** devuelve cada resultado de búsqueda en un *cluster* unitario. ONE IN ONE habitualmente garantiza el máximo valor de precisión, debido a que trivialmente se cumple que cada resultado de búsqueda debe ser miembro de su propio *cluster*. No obstante, esta política no obtiene el máximo valor de precisión en el caso extremo en el que el número de individuos mencionados sea superior al número de resultados de búsqueda: todos los resultados de búsqueda hablan de individuos distintos, y además, algunos de ellos mencionan a varios individuos distintos llamados igual que no son mencionados por el resto de páginas web. No obstante, esta situación no es habitual y, en particular, no sucede en ninguna de las colecciones de desambiguación de nombres de persona.
- **ALL IN ONE:** devuelve un único *cluster* compuesto por todos los resultados de búsqueda. ALL IN ONE habitualmente garantiza el máximo valor de cobertura, puesto que todos los resultados de búsqueda que hablan de cualquiera de los individuos están incluidos en el mismo *cluster*. No obstante, esta política no garantiza el máximo valor de cobertura para aquellos nombres de persona en los que existe solapamiento entre *clusters* dado que, por definición, ALL IN ONE no contiene ningún resultado de búsqueda que forme parte de distintos *clusters*. Como explicamos anteriormente (ver apartado 4.2.2), esta situación es más habitual que el caso extremo descrito para ONE IN ONE, pero, considerando las colecciones WePS, solamente se presenta en la colección de test de la campaña WePS-1.

Ambos *baselines* sirven para determinar el sesgo de una colección de datos con respecto al grado de ambigüedad de los individuos que contienen. En caso de que el valor de cobertura obtenido por ONE IN ONE sea elevado, significa que la colección de datos se compone principalmente de individuos muy ambiguos. En cambio, si el valor de precisión obtenido por ALL IN ONE es elevado, significa que la colección de datos se compone principalmente de individuos poco ambiguos.

Además, el algoritmo AP será empleado como *baseline* adicional, puesto que se trata de un método del estado del arte que tiene las mismas características que el algoritmo UPND: no requiere conocer previamente el número de *clusters* ni tampoco necesita datos de entrenamiento para aprender el valor de algún parámetro. En particular, emplearemos la configuración de este algoritmo con mejores resultados en los experimentos preliminares: uso de 1-gramas para representar los documentos, términos pesados mediante TF-IDF y similitud coseno.

Designaremos mediante P a los sistemas participantes en las campañas WePS, mientras que usaremos TP para designar a trabajos presentados con posterioridad y B para identificar a los *baselines*.

Los estudios de la significancia estadística de este apartado consisten en comparar los resultados de UPND con cada sistema. Los símbolos que emplearemos para ello son los siguientes:

- $\uparrow\uparrow$: Indica que UPND obtiene mejoras significativas con respecto al sistema.
- $\downarrow\downarrow$: Indica que el sistema obtiene mejoras significativas con respecto a UPND.
- $=$: Indica que el sistema obtiene resultados similares a los de UPND.
- **?**: Indica que no se ha podido realizar la comparación con el test de significancia estadística puesto que no se ha tenido acceso a los resultados obtenidos por el sistema para cada nombre de persona, a pesar de tener acceso a los resultados promedios de las métricas de evaluación.

Las Tablas 4.5, 4.6 y 4.7 muestran los resultados promedios de las métricas *B-Cubed* obtenidos por UPND (resaltados en color rojo), por otros sistemas del estado del arte y por los *baselines* para las colecciones de test de la campañas de evaluación WePS-1, WePS-2 y WePS-3, respectivamente. Los resultados aparecen ordenados con respecto al valor promedio $F_{0,5}$, empleado como métrica oficial de las campañas WePS. Además, en las tablas se indica también el tipo de sistema (B, P o TP), si el sistema necesita datos de entrenamiento o no (columna *Entr.*) y el estudio de significancia estadística mediante el test de Wilcoxon siguiendo la simbología explicada anteriormente.

Los resultados de las tablas nos indican lo siguiente:

- UPND obtiene mejoras significativas con respecto a los tres *baselines* empleados en las tres colecciones WePS. En particular, ONE IN ONE y ALL IN ONE son independientes de la representación usada y siempre devuelven el mismo número de *clusters*. Por otro lado, UPND mejora los resultados del algoritmo de *clustering* AP que cuenta con sus mismas características.

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Liu et al. [2011]	TP	SI	0.79	0.85	0.81	⇓⇓
Jiang et al. [2009]	TP	SI	0.80	0.79	0.78	*?*
UPND	TP	NO	0.85	0.70	0.76	=
Chen y Martin [2007a]	P	SI	0.61	0.83	0.70	⇑⇑
Elmacioglu et al. [2007]	P	SI	0.68	0.73	0.70	⇑⇑
Popescu y Magnini [2007]	P	SI	0.68	0.71	0.69	⇑⇑
Saggion [2007]	P	SI	0.54	0.74	0.62	⇑⇑
Balog et al. [2007]	P	SI	0.79	0.50	0.61	⇑⇑
ONE IN ONE	B	NO	1.00	0.43	0.57	⇑⇑
Ellman y Emery [2007]	P	SI	0.59	0.63	0.57	⇑⇑
AP	B	NO	0.80	0.53	0.55	⇑⇑
Kalmar y Blume [2007]	P	SI	0.43	0.84	0.53	⇑⇑
Lefever et al. [2007]	P	SI	0.42	0.80	0.51	⇑⇑
Kozareva et al. [2007]	P	NO	0.54	0.53	0.49	⇑⇑
Rao et al. [2007]	P	NO	0.36	0.73	0.43	⇑⇑
Iria et al. [2007]	P	NO	0.28	0.88	0.39	⇑⇑
Sugiyama y Okumura [2007]	P	SI	0.29	0.82	0.38	⇑⇑
Heyl y Neumann [2007]	P	SI	0.30	0.74	0.38	⇑⇑
del Valle-Agudo et al. [2007]	P	SI	0.26	0.91	0.36	⇑⇑
ALL IN ONE	B	NO	0.18	0.98	0.28	⇑⇑

Tabla 4.5: Resultados obtenidos por el algoritmo UPND, los sistemas del estado del arte y los *baselines* sobre la colección de test de WePS-1.

- UPND obtiene mejoras significativas con respecto a los sistemas del estado del arte que no requieren datos de entrenamiento, salvo en el caso del método propuesto por Xu et al. [2015] en la campaña WePS-2, que será comentado más adelante. La mayoría de estos métodos prefijan de antemano el número de *clusters* de cada nombre de persona (ej. [Kozareva et al., 2007; Lan et al., 2009]), lo cual tiene un efecto negativo en los resultados, dado que cada nombre de persona puede tener asociado un número distinto de individuos. No obstante, UPND también obtiene mejoras significativas sobre otros sistemas que estiman automáticamente el número de *clusters* y que están basados en otros algoritmos de *clustering* del estado del arte como *Fuzzy Ants* [Venkateshan, 2009] y el método de los *k*-vecinos [Lana-Serrano et al., 2010] estimando previamente el número de *clusters*.

- UPND obtiene mejoras significativas con respecto a la mayoría de los participantes de las tres campañas y obtiene resultados similares con respecto a algunos de los

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Yoshida et al. [2010]	TP	SI	0.89	0.82	0.85	*?*
Jiang et al. [2009]	TP	SI	0.85	0.83	0.83	*?*
Chen et al. [2009]	P	SI	0.87	0.79	0.82	⇊
Balog et al. [2009]	P	SI	0.85	0.80	0.81	=
Ikeda et al. [2009]	P	SI	0.93	0.73	0.81	=
Xu et al. [2015]	TP	NO	0.88	0.78	0.81	*?*
UPND	TP	NO	0.92	0.70	0.79	=
Romano et al. [2009]	P	SI	0.82	0.66	0.72	⇈
Kalmar y Freitag [2009]	P	SI	0.85	0.62	0.70	⇈
Gong y Oard [2009]	P	SI	0.94	0.60	0.70	⇈
Song et al. [2009]	P	SI	0.54	0.93	0.63	⇈
Han y Zhao [2009]	P	SI	0.65	0.75	0.63	⇈
Lefever et al. [2009]	P	SI	0.73	0.58	0.57	⇈
González et al. [2009]	P	SI	0.60	0.66	0.56	⇈
ALL IN ONE	B	NO	0.43	1.00	0.53	⇈
AP	B	NO	0.82	0.39	0.44	⇈
Lan et al. [2009]	P	NO	0.50	0.55	0.41	⇈
Martínez-Romo y Araujo [2009]	P	NO	0.66	0.39	0.40	⇈
Venkateshan [2009]	P	NO	0.61	0.38	0.39	⇈
ONE IN ONE	B	NO	1.00	0.24	0.34	⇈
Pinto et al. [2009]	P	NO	0.89	0.25	0.33	⇈

Tabla 4.6: Resultados obtenidos por el algoritmo UPND, los sistemas del estado del arte y los *baselines* sobre la colección WePS-2.

mejores participantes como Balog et al. [2009] o Ikeda et al. [2009] en WePS-2.

- En el caso de WePS-1, UPND mejora significativamente los resultados de todos los sistemas participantes en la campaña y solamente obtiene peores resultados que dos sistemas presentados con posterioridad que sí requieren datos de entrenamiento. En particular, el método propuesto por Jiang et al. [2009] hace uso de heurísticas *ad hoc* para el dominio Web, como por ejemplo, filtrar correos electrónicos de la forma *webmaster@domain-name*, *support@domain-name*, *feedback@domain-name*, que suelen ser comunes en determinadas páginas web, sin aportar información sobre los individuos y dando lugar a agrupaciones incorrectas. Por su parte, el método propuesto por Liu et al. [2011] obtiene mejores resultados con respecto al resto de sistemas. Esto puede deberse a que este sistema basado en HAC fue entrenado con la colección de test de WePS-2, mientras que el resto de sistemas

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Long y Shi [2010]	TP	SI	0.61	0.60	0.55	⇊
UPND	TP	NO	0.63	0.53	0.52	=
Smirnova et al. [2010]	P	SI	0.69	0.46	0.50	⇈
Ferrés y Rodríguez [2010]	P	SI	0.40	0.66	0.44	⇈
Nagy [2012]	TP	SI	0.38	0.61	0.40	⇈
Dornescu et al. [2010]	P	SI	0.31	0.80	0.40	⇈
Lana-Serrano et al. [2010]	P	NO	0.29	0.84	0.39	⇈
AP	B	NO	0.75	0.31	0.39	⇈
ONE IN ONE	B	NO	1.00	0.23	0.35	⇈
ALL IN ONE	B	NO	0.22	1.00	0.32	⇈

Tabla 4.7: Resultados obtenidos por el algoritmo UPND, los sistemas del estado del arte y los *baselines* sobre la colección WePS-3.

emplearon la colección de entrenamiento de WePS-1. La colección de datos de WePS-2 se ajusta mejor a la colección de test de WePS-1, puesto que ambas son más cercanas con respecto al grado de ambigüedad de los nombres de personas que contienen, lo cuál permite que el umbral obtenido sea más ajustado.

- En el caso de WePS-2, UPND mejora significativamente los resultados de la mayoría de sistemas participantes de la campaña y obtiene resultados similares con respecto a dos de ellos, pese a que requieren datos de entrenamiento. Solamente, el mejor participante de esta campaña [Chen et al., 2009] mejora significativamente los resultados obtenidos por UPND. El método propuesto por Xu et al. [2015] es el único sistema que no requiere datos de entrenamiento que obtiene resultados ligeramente superiores a UPND. No obstante, no se ha podido comprobar si existen mejoras significativas con respecto a este método dado que no se tuvo acceso a sus resultados para cada nombre de persona. El método propuesto por Xu et al. [2015] utiliza siete clases de rasgos diferentes, entre los que se incluye información extraída de Wikipedia. La extracción de estos rasgos supone necesariamente un mayor coste computacional, dado que debe realizarse en tiempo real después de que el usuario haya introducido la consulta. Por otro lado, los sistemas presentados por Jiang et al. [2009] y Yoshida et al. [2010] obtienen los mejores resultados. Ambos métodos utilizan información de los datos de entrenamiento y aplican una estrategia basada en agrupar las páginas web en dos fases: una primera fase consistente en obtener *clusters* muy precisos, y una segunda fase consistente en refinar dichos *clusters*.

- En el caso de WePS-3, los valores de las métricas de todos los sistemas son más

bajos con respecto a las otras dos colecciones, dado que solo se evaluó la calidad de los *clusters* de uno o dos individuos por nombre de persona. Por tanto, los errores en dichos *clusters* penalizan más que evaluar el agrupamiento completo como sucede con WePS-1 y WePS-2. UPND solamente obtiene peores resultados con respecto al sistema propuesto por Long y Shi [2010], que emplea HAC y requiere datos de entrenamiento. Dicho sistema explota el uso de datos de Wikipedia, que los propios autores recolectaron manualmente antes de ejecutar el algoritmo.

4.3.4. Discusión

En esta sección se analiza la configuración del algoritmo UPND y la evaluación con las colecciones WePS.

En primer lugar, se ha comprobado que UPND obtiene resultados similares a HAC empleando la misma configuración (3-gramas en mayúsculas, pesado TF-IDF y similitud coseno), a pesar de que se ha provisto a este último del mejor umbral de similitud promedio. Esto significa que los umbrales adaptativos, además de evitar el requerimiento de datos de entrenamiento, son adecuados a la hora de comparar resultados de búsqueda entre sí. Además, también se ha comprobado que el algoritmo UPND es menos sensible que HAC con respecto al grado de ambigüedad de los nombres de personas.

Con respecto a la configuración del algoritmo, se ha estudiado el uso de distintas funciones de pesado de términos y medidas de similitud. Por un lado, se ha comprobado que UPND obtiene resultados similares empleando distintas funciones de pesado de términos, debido a que el uso de 3-gramas en mayúsculas genera vocabularios muy pequeños, de manera que las agrupaciones de páginas web depende de que compartan un pequeño número de rasgos de este tipo. Por ello, resulta más conveniente emplear funciones de pesado locales por ser menos costosas computacionalmente. En particular, se ha optado por la función binaria por ser la más sencilla de todas. Por otro lado, la similitud coseno es más adecuada que Jaccard pesado por dos motivos: (i) la función coseno asegura una mayor independencia con respecto al tamaño de los documentos gracias a la normalización de los vectores; y (ii) la función Jaccard pesado es similar a la función de umbral adaptativo, de modo que ambas generan valores similares teniendo como consecuencia que haya un menor número de agrupaciones entre resultados de búsqueda. Por tanto, se ha concluido que la función coseno es más adecuada a la hora de comparar los resultados de búsqueda.

Se han comparado los resultados de UPND con respecto a los obtenidos por varios *baselines* y otros sistemas del estado del arte. En particular, UPND obtiene mejoras significativas sobre los *baselines* y los sistemas del estado del arte que no requieren datos de entrenamiento, salvo el presentado por Xu et al. [2015] que obtiene resultados simi-

lares en la colección WePS-2. Por otra parte, UPND también mejora significativamente los resultados de la mayoría de los sistemas que requieren datos de entrenamiento. No obstante, los mejores sistemas de desambiguación utilizan datos de entrenamiento y presentan un rendimiento superior a UPND.

La principal desventaja de UPND consiste en que no es capaz de representar un alto porcentaje de páginas web debido a la representación basada en 3-gramas en mayúsculas. UPND devuelve los resultados de búsqueda que no puede representar en *clusters* unitarios, debido a que no comparten ningún rasgo con ningúna otra página web, de modo que el umbral adaptativo vale 1. Los *clusters* unitarios tienen un impacto positivo en los resultados de precisión pero un efecto perjudicial en los valores de cobertura, lo cual explica los resultados obtenidos por UPND. No obstante, esto implica que UPND tiene margen de mejora en caso de que se enriquezca la representación de las páginas web con otro tipo de rasgos. En particular, los mejores sistemas del estado del arte que mejoran significativamente los resultados de UPND se caracterizan por emplear distintos tipos de rasgos para representar los resultados de búsqueda.

4.4. Conclusiones

Este capítulo se ha dividido en tres bloques. En primer lugar, se han enunciado hipótesis sobre qué clase de rasgos son útiles para identificar y distinguir entre varios individuos que comparten el mismo nombre en el dominio Web, y se han llevado a cabo experimentos preliminares para corroborarlas. En segundo lugar, hemos presentado un nuevo algoritmo de *clustering* aglomerativo llamado UPND para agrupar resultados de búsqueda según el individuo al que mencionan. UPND no requiere conocer el número de *clusters* ni tampoco necesita datos de entrenamiento gracias a que hace uso del concepto de umbral adaptativo, también propuesto en este trabajo. Finalmente, en tercer lugar, se han analizado los resultados del algoritmo UPND con respecto a varios factores, y se han comparado sus resultados con los obtenidos por los sistemas del estado del arte y varios *baselines*. Por tanto, las conclusiones de este capítulo podemos dividirlas de acuerdo a los bloques mencionados anteriormente.

Representación de los documentos

- Las representaciones mediante n-gramas y k-skip-n-gramas tienden a obtener mejores resultados de precisión a medida que dichos rasgos se componen de un mayor número de palabras, tal y como se afirma en la hipótesis (RD1).
- Las representaciones mediante k-skip-n-gramas no mejoran los resultados de aquellas basadas en n-gramas. Además, generan un vocabulario más grande que im-

plica un mayor coste computacional en tiempo y espacio. Por tanto, estos rasgos no son adecuados para representar el contenido de los resultados de búsqueda.

- El porcentaje de resultados de búsqueda con representación es más bajo a medida que los *n*-gramas y *k*-skip-*n*-gramas se componen de más palabras. Esto tiene un impacto negativo en los valores de cobertura. Experimentalmente, se ha comprobado que existe una bajada drástica en los resultados cuando se emplean rasgos compuestos por 4 palabras.

- Las representaciones mediante rasgos compuestos por palabras en mayúsculas obtienen resultados similares con respecto a las representaciones homólogas mediante cualquier tipo de palabra. No obstante, las primeras son más eficientes computacionalmente puesto que generan un vocabulario más pequeño. Esto confirma que las palabras en mayúsculas capturan información adecuada para representar los resultados de búsqueda como se afirma en la hipótesis (RD2).

- La extracción de rasgos en mayúsculas captura la mayoría de las palabras contenidas en las NEs junto con otros rasgos que no son extraídos por sistemas NER. Además, la extracción de las NEs depende del rendimiento del sistema NER empleado, mientras que esto no sucede cuando se extraen palabras en mayúsculas.

- La representación mediante 3-gramas en mayúsculas es menos sensible al grado de ambigüedad de los nombres de persona que los 1-gramas y las NEs, y obtiene *clusters* más precisos.

Algoritmo UPND

- Se ha corroborado que los resultados obtenidos por el algoritmo HAC son sensibles con respecto al valor de umbral empleado para cortar el dendrograma que genera.

- Se ha verificado que con el algoritmo HAC, una estrategia basada en obtener un valor de umbral diferente para cada nombre de persona es capaz de obtener mejoras significativas con respecto a la estrategia empleada por los mejores sistemas de desambiguación, basada en aprender un único umbral mediante datos de entrenamiento que posteriormente se aplica a todos los nombres de persona.

- Se ha presentado el concepto de umbral adaptativo consistente en una función matemática que obtiene un valor de umbral distinto teniendo en cuenta las características de los documentos que se comparan. El uso de este concepto evita la necesidad de aprender un valor de umbral mediante datos de entrenamiento.

- Se ha descrito el algoritmo de *clustering* aglomerativo UPND. Este método obtiene la partición del conjunto de resultados de búsqueda generada a partir de una relación de equivalencia basada en la condición de agrupamiento $sim^n(W_i, W_j) > \gamma^n(W_i, W_j)$, donde *sim* es una medida de similitud y γ es un umbral adaptativo. Gracias al empleo del umbral adaptativo, UPND no requiere conocer a priori el número de *clusters* ni necesita datos de entrenamiento para aprender el valor de algún parámetro.

Resultados

- UPND es capaz de obtener resultados similares a HAC cuando a este se le proporciona el mejor umbral promedio de toda la colección de datos. Esto corrobora que las propiedades de los umbrales adaptativos son adecuadas. Además, UPND es menos sensible que HAC con respecto al grado de ambigüedad de los nombres de personas.
- Los resultados obtenidos por UPND son independientes con respecto a la función de pesado de términos empleada debido a que la mera compartición de pocos 3-gramas en mayúsculas entre páginas web cumple la condición de agrupamiento.
- UPND obtiene mejores resultados cuando aplica la función coseno con respecto a Jaccard pesado, debido a la normalización de los pesos por parte de la primera y a que Jaccard pesado tiende a tener resultados similares al umbral adaptativo, puesto que ambas fórmulas se basan en la intersección de rasgos entre los documentos que se comparan.
- UPND obtiene mejoras significativas con respecto a los *baselines* y la mayoría de los sistemas del estado del arte: tanto los que no requieren datos de entrenamiento, como los que si lo requieren.
- Los sistemas que mejoran los resultados de UPND emplean HAC y suelen representar los documentos mediante una amplia gama de rasgos de distinto tipo. Esto indica que UPND puede tener margen de mejora si se complementa la representación basada en 3-gramas en mayúsculas con otros tipos de rasgos.

En definitiva, el algoritmo UPND cuenta con las ventajas de que no requiere aprender información previa mediante datos de entrenamiento y es capaz de estimar el número de *clusters*. Además, obtiene resultados competitivos con respecto a los sistemas del estado del arte, dado que mejora a la mayoría de los sistemas con estas características e incluso es capaz de obtener resultados similares a los obtenidos por varios de los mejores sistemas del estado del arte. No obstante, el principal inconveniente de este

algoritmo es que la selección de rasgos empleada no permite representar todos los resultados de búsqueda. En concreto, aquellas páginas web que contienen poca información escrita en mayúsculas o incluso todo su contenido está escrito en minúsculas. Por esta razón, UPND obtiene valores altos de precisión con respecto a otros sistemas del estado del arte basados en una representación mediante palabras que sí consiguen representar muchos más documentos. Esto implica que UPND presenta un margen de mejora, pero para ello deben emplearse rasgos adicionales que permitan representar tantos resultados de búsqueda como sea posible. En este sentido, algunos de los mejores sistemas del estado del arte [Jiang et al., 2009; Yoshida et al., 2010] toman una estrategia basada en calcular inicialmente un conjunto de *clusters* muy precisos y, posteriormente, aplicar una serie de fases de refinamiento de dichos *clusters* para poder agrupar el resto de documentos. En el Capítulo 5, se presentará una extensión del algoritmo UPND que soluciona su principal inconveniente en base a esta idea.

5

Segunda aproximación a la desambiguación de nombres de persona en la Web

"En nuestra profesión, la precisión y la perfección no son lujos prescindibles, sino simplemente una necesidad."

— Niklaus Wirth —

Este capítulo presenta la segunda propuesta de desambiguación de nombres de personas en la Web de la presente tesis doctoral. El método propuesto se compone de tres fases y sigue una estrategia de agrupamiento consistente en obtener clusters iniciales con un alto grado de precisión y, posteriormente, mezclarlos entre sí para obtener los clusters finales. Además, la propuesta emplea diversos tipos de rasgos que permiten resolver la limitación de UPND con respecto a la representación de los resultados de búsqueda. En primer lugar, se presenta el esquema de agrupamiento de la segunda propuesta de desambiguación de nombres de personas en la Web. Posteriormente, se explican las fases en las que se divide la propuesta y se analizan los resultados obtenidos por cada una de ellas. A continuación, se comparan los resultados de la propuesta con respecto al algoritmo UPND y los sistemas del estado del arte. Finalmente, se enumeran las principales conclusiones extraídas de este capítulo.

5.1. Algoritmo propuesto: *Adaptive Threshold Clustering* (ATC)

Como vimos en el Capítulo 4, el algoritmo UPND obtiene resultados competitivos sin necesidad de aprender parámetros mediante datos de entrenamiento y, además, es capaz de estimar automáticamente el número de *clusters*. No obstante, los rasgos empleados por UPND no permiten representar todas las páginas web, de manera que las que se quedan sin representar serán devueltas dentro de clusters unitarios. En esta sección se describe la nueva propuesta de esta tesis doctoral, el algoritmo *Adaptive Threshold Clustering* (ATC). Por un lado, este algoritmo se basa en el uso de umbrales adaptativos, de manera que, al igual que UPND, es capaz de estimar el número de *clusters* automáticamente y no requiere aprender ningún parámetro mediante datos de entrenamiento. Por otro lado, ATC emplea diferentes tipos de rasgos que garantizan la representación

de todas las páginas web que contienen información textual, de modo que resuelve la principal limitación del algoritmo UPND.

El algoritmo ATC se compone de tres fases que pueden dividirse en dos grupos: las dos primeras son fases de generación de *clusters* iniciales y la última es una fase de fusión de *clusters*. La estrategia de *clustering* de ATC es similar a la de algunos de los mejores sistemas del estado del arte [Jiang et al., 2009; Yoshida et al., 2010] que trabajan en varias fases, de forma que la primera se centra en obtener *clusters* muy cohesivos. En el caso de ATC, las fases de generación de *clusters* iniciales tienen como objetivo obtener una agrupación con un alto valor de precisión. Para ello, se utilizan como rasgos *links* y 3-gramas en mayúsculas, para los que asumimos que las páginas web que los comparten se refieren al mismo individuo con mucha probabilidad. Posteriormente, la fase de fusión de *clusters* tiene como objetivo agrupar los *clusters* iniciales entre sí. Para ello, durante esta fase los resultados de búsqueda se representan mediante rasgos que coaparecen más frecuentemente en diferentes páginas web, particularmente 1-gramas (BoW).

El algoritmo ATC aplica secuencialmente las siguientes fases de agrupación de resultados de búsqueda:

- **Fases de generación de *clusters* iniciales:** estas fases se caracterizan porque se comparan resultados de búsqueda entre sí.
  - **Fase 1: agrupación de páginas web mediante *links***. Se agrupan las páginas web a partir de los *links* que contienen. Por esta razón, en esta fase cada resultado de búsqueda se representa mediante su URL y sus *links*.
  - **Fase 2: algoritmo UPND**. Se agrupan los resultados de búsqueda mediante el algoritmo UPND aplicando la configuración explicada en el Capítulo 4. Por tanto, en esta fase los resultados de búsqueda se representan mediante 3-gramas en mayúsculas y pesado binario.
- **Fase 3: fusión de *clusters*:** esta fase se caracteriza porque se comparan *clusters* de resultados de búsqueda entre sí. En particular, se intentan mezclar los *clusters* devueltos por las fases anteriores, sean unitarios o no. En esta fase, los *clusters* se representan mediante sus *centroides*, los cuales capturan el contenido de los resultados de búsqueda de los que se compone cada *cluster*. La fusión de *clusters* se justifica por las siguientes razones:
  - Los *clusters* generados por UPND habitualmente se componen de resultados de búsqueda que hablan de un mismo individuo porque tienen altos valores de precisión (ver sección 4.3).

- Muchos resultados de búsqueda no son representados mediante 3-gramas en mayúsculas, de modo que no pueden ser comparados con otros resultados de búsqueda mediante UPND y son devueltos como *clusters* unitarios. Por tanto, es posible que varios de los *clusters* generados por UPND hablen del mismo individuo.

En esta fase los resultados de búsqueda se representan mediante 1-gramas por los siguientes motivos:

- En el apartado 4.1.2 se comprobó que los 1-gramas son rasgos adecuados para representar los resultados de búsqueda.
- Esta representación permite representar el máximo número de resultados de búsqueda. En particular, empleando BoW se pueden representar aquellos resultados de búsqueda escritos completamente en minúsculas que no pudieron ser representados en la fase 2.

Por tanto, ATC representa cada resultado de búsqueda mediante su URL, sus *links*, 3-gramas en mayúsculas y 1-gramas. Las URLs son proporcionadas en todas las colecciones consideradas en esta tesis doctoral. El resto de rasgos se extraen durante la fase de preprocesamiento empleando el *parser* HTML *TiKa Apache* sin necesidad de emplear otros recursos adicionales tales como *POS taggers* o sistemas de *NER*.

Las secciones 5.2 y 5.3 explican en detalle cada una de las fases de ATC.

5.2. Fases de generación de *clusters* iniciales

En esta sección se describen detalladamente las dos fases de generación de *clusters* iniciales de ATC. La primera fase se basa en la agrupación de resultados de búsqueda mediante *links*, mientras que la segunda fase emplea el algoritmo UPND. El objetivo de ambas fases consiste en obtener una agrupación caracterizada por tener un alto grado de precisión, de modo que la última fase de ATC tiene como objetivo mejorar el valor de cobertura.

5.2.1. Fase 1: agrupación de páginas web por *links*

El agrupamiento entre resultados de búsqueda realizado durante la primera fase de ATC se obtiene a partir de la estructura de hipervínculos de los resultados de búsqueda. Varios autores [Bekkerman y McCallum, 2005; Iria et al., 2007; Kozareva et al., 2007; Yoshida et al., 2010; Xu et al., 2015] asumen que los resultados de búsqueda relacionados

entre sí mediante *links* hablan de un mismo individuo. En este apartado verificaremos la validez de esta afirmación. Para ello, compararemos el rendimiento de las siguientes políticas de agrupamiento basadas en *links* en las colecciones WePS:

- **Enlace directo:** consiste en agrupar dos *resultados de búsqueda* si están *enlazados* entre sí, i.e. la URL de uno de ellos aparece en el conjunto de *links* del otro.
- **Enlace indirecto:** consiste en agrupar dos resultados de búsqueda si están enlazados o ambos tienen un *link* en común.

Dado un ranking de resultados de búsqueda $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$ denotaremos mediante $URL(W_i)$ y $links(W_i)$ a la URL y el conjunto de los *links* del resultado de búsqueda $W_i \in \mathcal{W}$, respectivamente. Las URLs de los resultados de búsqueda están contenidas en los *snippets* generados por el buscador y son proporcionadas en todas las colecciones consideradas en esta tesis. Por otro lado, los *links* pueden extraerse mediante cualquier *parser* HTML durante la etapa de preprocesamiento.

Las anteriores políticas de agrupamiento mediante *links* de dos resultados de búsqueda $W_i, W_j \in \mathcal{W}$ se definen mediante los siguientes predicados booleanos:

$$enlaceDirecto(W_i, W_j) \equiv URL(W_i) \in links(W_j) \vee URL(W_j) \in links(W_i) \tag{5.1}$$

$$enlaceIndirecto(W_i, W_j) \equiv enlaceDirecto(W_i, W_j) \vee links(W_i) \cap links(W_j) \neq \emptyset \tag{5.2}$$

El pseudocódigo de la primera fase de ATC se encuentra en el Algoritmo 5.3 y simplemente agrupa en el mismo *cluster* aquellos resultados de búsqueda que estén enlazados y no hayan sido agrupados en alguna iteración anterior. El algoritmo es determinista puesto que siempre se obtiene el mismo resultado de *clustering* con independencia del orden en el que se comparen los resultados de búsqueda, debido a que las políticas de enlace directo e indirecto son transitivas. Además, como ambas relaciones también son simétricas, se garantiza que el algoritmo devuelve la misma agrupación con independencia del orden de las páginas web en el ranking de resultados de búsqueda. Por otro lado, dado que se compara cada par de resultados de búsqueda, la complejidad del algoritmo se encuentra en $\mathcal{O}(N^2)$ siendo N el número de resultados de búsqueda.

La Tabla 5.1 muestra los resultados obtenidos por ambas políticas de agrupamiento mediante *links* en las colecciones de test de WePS con respecto a los del *baseline* ONE IN ONE que no realiza ningún agrupamiento. De este modo, podemos verificar si el agrupamiento mediante *links* consigue agrupar resultados de búsqueda y si dichos agrupamientos son correctos a traves de los valores de cobertura y precisión respectivamente.

Algoritmo 5.3 $LINKS(\mathcal{W}, enlace)$.

Entrada: Ranking de resultados de búsqueda $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$, política de agrupamiento *enlace*, la cual puede ser *enlaceDirecto* o *enlaceIndirecto*.

Salida: Conjunto de *clusters* $\mathcal{C}$.

para $i = 1$ **hasta** N **hacer**
  $C_i = \{W_i\}$
fin para
$\mathcal{C} = \bigcup_{i=1}^{N} C_i$
para $i = 1$ **hasta** N **hacer**
  para $j = i + 1$ **hasta** N **hacer**
    si $\neg mismoCluster(\mathcal{C}, W_i, W_j) \wedge enlace(W_i, W_j)$
      $C_i = C_i \cup C_j$
      $\mathcal{C} = \mathcal{C} \setminus \{C_j\}$
    fin si
  fin para
fin para
devolver $\mathcal{C}$

Además, la tabla muestra el estudio de significancia estadística. Cada experimento tiene una marca en la columna $F_{0,5}$ de la forma (k) donde $k \in \mathbb{N}$, de manera que un experimento marcado con (k) obtiene mejoras significativas sobre otro marcado con (k') si $k < k'$, y ambos obtienen resultados similares si $k = k'$.

Política	**ONE IN ONE**			**enlaceDirecto**			**enlaceIndirecto**		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	1.00	0.43	0.57 (3)	0.99	0.46	0.60 (2)	0.94	0.54	0.67 (1)
WePS-2	1.00	0.24	0.34 (3)	1.00	0.25	0.36 (2)	0.96	0.36	0.48 (1)
WePS-3	1.00	0.23	0.35 (2)	0.98	0.25	0.36 (2)	0.80	0.35	0.45 (1)

Tabla 5.1: Resultados obtenidos por distintas políticas de agrupamiento mediante *links* y el *baseline* ONE IN ONE en las colecciones WePS de test.

En primer lugar, la tabla muestra que los valores de precisión de la política de enlace directo son solo ligeramente más bajos que los del *baseline* ONE IN ONE. Esto significa que los resultados de búsqueda enlazados con enlace directo obtenidos tras consultar un nombre de persona generalmente se refieren al mismo individuo. En particular, el descenso de precisión es algo mayor en la colección WePS-3 porque únicamente se evalúan uno o dos *clusters* a diferencia del resto de colecciones, de modo que cualquier error tiene un impacto más negativo en los resultados.

La política de enlace indirecto obtiene mejoras significativas con respecto a la de enlace directo y el *baseline* ONE IN ONE en todas las colecciones WePS, mientras que la política de enlace directo no mejora significativamente los resultados del *baseline* en la colección WePS-3. El inconveniente de la política de enlace indirecto es que genera un mayor número de agrupaciones incorrectas dado que obtiene un valor de precisión más bajo. No obstante, la política de enlace indirecto obtiene una mejora del valor de cobertura superior a la reducción del valor de precisión, lo cuál explica la mejora significativa en el valor de medida-F. Los errores obtenidos por la política de enlace indirecto vienen provocados por diferentes tipos de *links*, siendo los más habituales los siguientes:

- **Buscadores:** varias páginas web que hablan de individuos distintos enlazan a buscadores como *www.google.com* o *www.yahoo.com*. Por otro lado, algunos errores son debidos a los buscadores verticales especializados en búsqueda de personas, dado que devuelven un listado de individuos diferentes que comparten el mismo nombre.

- **Redes sociales y blogs:** las páginas web de una misma red social se agrupan entre sí por dos posibles motivos: (i) todas ellas están enlazadas a su página principal (*homepage*); y (ii) algunas de ellas consisten en una lista de usuarios de la red social llamados de la misma manera. Por otro lado, esta situación sucede de forma similar con los *blogs* personales de distintos individuos alojados en portales web como *Blogger*[1] o *Wordpress*[2], dado que todos ellos contienen un enlace a su *homepage*.

- **Periódicos:** se agrupan entre sí artículos de un mismo periódico *online* aunque hablen de individuos distintos con el mismo nombre, debido a que enlazan a su *homepage*.

- **Enciclopedias *online* y bases de datos:** se agrupan en el mismo *cluster* las entradas de enciclopedias *online* como *Wikipedia* o *Answers*[3] debido a que enlazan a sus *homepages*. Esta situación sucede de forma similar con bases de datos dedicadas a una determinada temática, como sucede con la página web *IMDb*[4] que alberga información sobre el mundo del cine y la televisión.

- **Recursos web:** en este grupo se encuentran *links* a las *homepages* de servicios empleados por páginas web para monitorizar su actividad y realizar estadísticas de sus visitas, sitios oficiales de descarga de tecnologías necesarias para visualizar correctamente las páginas web (ej. *Adobe*) o sitios web que ofrecen servicios de publicidad en la Red.

[1] `https://www.blogger.com/`
[2] `https://wordpress.org/`
[3] `www.answers.com/`
[4] `http://www.imdb.com/`

Los experimentos llevados a cabo posteriormente emplearán la política de enlace indirecto, puesto que es la que asegura mejoras significativas en los resultados de la medida-F cuando las páginas web se representan mediante URLs y sus hipervínculos.

5.2.2. Fase 2: algoritmo UPND

ATC aplica en la segunda fase el algoritmo de *clustering* UPND representando los resultados de búsqueda mediante sus 3-gramas en mayúsculas. En particular, se asume la configuración discutida en el capítulo anterior: pesado binario, similitud coseno y la función de umbral adaptativo presentada en el apartado 4.2.1. El algoritmo aplicado se corresponde con el Algoritmo 4.2, exceptuando las líneas 1-4 donde inicialmente se genera un *cluster* unitario para cada resultado de búsqueda dado como entrada, puesto que parte de los *clusters* generados en la fase 1, y la línea 15 donde se devuelve el conjunto de *clusters*, puesto que ATC aplica posteriormente la fase de fusión de *clusters*. UPND es un algoritmo determinista, dado que siempre devuelve el mismo resultado con independencia del orden en el que se comparan las páginas web. Esto es debido a que computa el cierre transitivo de una relación. Por otro lado, dado que UPND compara cada par de resultados de búsqueda, su coste temporal se encuentra en $\mathcal{O}(N^2)$ siendo N el número de resultados de búsqueda.

La Tabla 5.2 muestra los resultados obtenidos por el algoritmo UPND, la fase 1 y las fases 1 y 2 de ATC aplicadas secuencialmente. De este modo, por un lado podemos analizar la aportación de la agrupación inicial mediante *links* (al comparar UPND y FASES 1+2) y, por otro lado, podemos comprobar la variación de los resultados tras la aplicación de cada fase (al comparar FASE 1 y FASES 1+2). La tabla también muestra el estudio de significancia estadística empleando la simbología presentada en la Tabla 5.1.

Política	**UPND**			**FASE 1**			**FASES 1+2**		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	0.85	0.70	0.76 (1)	0.94	0.54	0.67 (2)	0.81	0.74	0.76 (1)
WePS-2	0.92	0.70	0.79 (2)	0.96	0.36	0.48 (3)	0.88	0.77	0.81 (1)
WePS-3	0.63	0.53	0.52 (1)	0.80	0.35	0.45 (2)	0.62	0.55	0.53 (1)

Tabla 5.2: Resultados obtenidos por UPND, la fase 1 de ATC y la combinación de las fases 1 y 2 de ATC en las colecciones WePS de test.

La tabla refleja lo siguiente:

- Por un lado, se observa que la fase 2 (experimento FASE 1+2) mejora significativamente los resultados obtenidos en la fase 1 (experimento FASE 1) en todas las

colecciones WePS. Esto significa que se cumple el objetivo de que en cada fase se vaya mejorando el agrupamiento de los resultados de búsqueda.

- Por otro lado, FASES 1+2 obtiene peores resultados de precisión con respecto a UPND en todas las colecciones. Esto se debe a que arrastra los errores de la fase 1 explicados anteriormente. No obstante, FASES 1+2 obtiene mejores valores de cobertura con respecto a UPND, por lo que el agrupamiento por *links* tiene la ventaja de que permite agrupar un mayor número de resultados de búsqueda. En particular, FASES 1+2 y UPND obtienen resultados similares en WePS-1 y WePS-3, pero FASES 1+2 mejora significativamente los resultados de UPND en WePS-2. Esto significa que el agrupamiento inicial mediante *links* es beneficioso porque permite agrupar correctamente resultados de búsqueda entre sí que no se pueden fusionar mediante la representación con 3-gramas en mayúsculas.

5.3. Fase 3: fusión de *clusters*

La fase de fusión de *clusters* de ATC tiene como objetivo agrupar los *clusters* iniciales generados en las dos primeras fases del algoritmo para mejorar los resultados de cobertura. La agrupación de los *clusters* iniciales se justifica porque es posible que varios de ellos contengan páginas web que mencionan a un mismo individuo, debido a que se puede dar el caso de que las páginas web no compartan los rasgos empleados durante las dos fases anteriores. En particular, muchos *clusters* se caracterizan por contener una única página web debido a que no se pudieron representar mediante 3-gramas en mayúsculas. Por otra parte, durante esta última fase los resultados de búsqueda se representan mediante 1-gramas (BoW), puesto que son el tipo de *n*-gramas que permiten representar el máximo número de resultados, como muestra la Tabla 4.1, y porque aumentará la probabilidad de que las páginas web compartan más rasgos.

Los *clusters* son representados mediante sus *centroides*, los cuales resumen el contenido de los documentos contenidos en cada uno de ellos. A continuación, en primer lugar se explica la representación de los *clusters* mediante tres tipos de centroides diferentes. Posteriormente, se detalla el algoritmo de mezcla de *clusters* llevado a cabo durante la fase 3 y se analizan los resultados de esta fase teniendo en cuenta los tipos de centroides explicados y diferentes funciones de pesado de términos. Además, se analiza la aportación de la fase 3 con respecto a los resultados obtenidos en las fases anteriores.

5.3.1. Representación de los *clusters*

La fase de fusión de *clusters* se diferencia de las dos primeras fases porque se comparan *clusters* entre sí en lugar de resultados de búsquedas. Los *clusters* pueden contener

una o varias páginas web, pero las funciones de similitud y umbral adaptativo consideradas en este trabajo solamente se definen para un único documento representado mediante una *bolsa de términos*. Por ello se han utilizado centroides, que permiten representar los clusters de forma similar a las páginas web mediante bolsas de términos. La definición de este concepto se presenta a continuación:

Definición 5.1. Sea $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$ un corpus donde $V_\mathcal{D} = |Voc(\mathcal{D})|$, de manera que cada documento se representa mediante VSM y $\vec{d_i} = (d_{i,1}, d_{i,2}, \ldots, d_{i,V_\mathcal{D}})$ es el vector de rasgos correspondiente al documento $D_i \in \mathcal{D}$. Sea $C \in \mathcal{P}(\mathcal{D})$ un *cluster* de documentos. El **centroide** del *cluster* C es un documento $\mu(C)$ donde:

- Su vocabulario es $Voc(\mu(C)) = \bigcup_{D_i \in C} Voc(D_i)$
- Su vector de rasgos es $\vec{\mu}(C) = (\mu_1^C, \mu_2^C, \ldots, \mu_{V_\mathcal{D}}^C)$ donde:

$$\forall j \in \{1, 2, \ldots V_\mathcal{D}\} : \mu_j^C = \frac{1}{|C|} \cdot \sum_{D_i \in C} d_{i,j} \tag{5.3}$$

En lo sucesivo, denominaremos como *centroide teórico* a aquellos centroides obtenidos mediante la Definición 5.1. El centroide teórico puede verse como un único documento construido mediante la concatenación de los documentos contenidos en el *cluster* correspondiente [Aggarwal y Zhai, 2012]. Por esta razón, el vocabulario del centroide teórico de un *cluster* contiene a todos los términos que aparecen en los documentos del *cluster*. Por otra parte, el vector de rasgos de un centroide teórico se obtiene como el *baricentro* del conjunto de vectores de rasgos de los documentos del *cluster* correspondiente, puesto que el peso asociado a cada rasgo del centroide se computa mediante la media aritmética de los pesos de dicho rasgo en los documentos del *cluster*.

De acuerdo con Manning et al. [2008], la representación de los *clusters* mediante centroides teóricos presenta dos inconvenientes:

- Los centroides teóricos tienden a representar los *clusters* compuestos por más documentos con un mayor número de rasgos, lo cual dificulta la comparación entre *clusters* de distinto tamaño mediante medidas de similitud.
- El empleo de centroides teóricos tiene un impacto negativo en términos de coste computacional.

Por estos motivos, es habitual emplear métodos de truncamiento de centroides consistentes en seleccionar un subconjunto de rasgos considerados relevantes mediante algún criterio. Los métodos de truncamiento permiten obtener representaciones más concisas y eficientes de los *clusters* [Aggarwal y Zhai, 2012]. A continuación, se explican dos métodos truncamiento de centroides que serán comparados posteriormente:

Truncamiento por peso

Manning et al. [2008] explican que el método de truncamiento más habitual consiste en seleccionar los rasgos con mayor peso del centroide teórico. En particular, frecuentemente se toman los r rasgos con mayor peso del centroide teórico. Para evitar la introducción de este parámetro, se propone el siguiente método de selección de rasgos:

- Para cada *cluster* $C \in \mathcal{C}$ hacer lo siguiente:
  - Calcular la mediana de los pesos de los rasgos del centroide teórico $\mu(C)$, i.e.

$$Me(\mu(C)) = Me\{\mu_j^C | j \in \{1, 2, \ldots, V_{\mathcal{D}}\}\} \tag{5.4}$$

  - Eliminar del centroide los rasgos con menor peso que la mediana: un rasgo $f_j \in Voc(C)$ se filtra del centroide si se cumple que $\mu_j^C < Me(\mu(C))$.

El empleo de la mediana se justifica porque se trata de una medida de tendencia central que evita los casos extremos, al contrario que otros estadísticos como la media aritmética. Por ejemplo, la mediana evita un filtrado masivo de rasgos con respecto a la media aritmética en el caso de que haya muy pocos rasgos con un peso muy alto.

En lo sucesivo, emplearemos la notación $\omega(C)$ para denotar el centroide del *cluster* C obtenido mediante el método anterior de truncamiento por peso.

Truncamiento DF-ICF

Este criterio de selección de rasgos relevantes del centroide de un *cluster* se basa en las siguientes propiedades:

- El rasgo aparece en la mayoría de los documentos del *cluster*. Los rasgos que cumplen esta propiedad consisten en información común entre los documentos del *cluster*. Dado que los *clusters* obtenidos en las fases 1 y 2 tienen altos valores de precisión, estos rasgos pueden corresponderse con información útil para identificar a cada individuo.
- El rasgo aparece en pocos *clusters*. Los rasgos que cumplen esta propiedad consisten en información distintiva de cada *cluster* que puede servir de utilidad para distinguir a un individuo con respecto el resto.

La detección de los rasgos que cumplen la primera propiedad puede llevarse a cabo a partir de su *frecuencia de documento* (*Document Frequency*, DF) en un *cluster*. Dado un rasgo $f \in Voc(\mathcal{D})$, su frecuencia de documento en un *cluster* C se computa de la siguiente manera:

$$DF(f, C) = |\{D_i \in C | f \in Voc(D_i)\}| \tag{5.5}$$

Por otro lado, la detección de los rasgos que cumplen la segunda propiedad puede llevarse a cabo mediante su *frecuencia inversa de* cluster (*Inverse Cluster Frequency*, ICF) en un conjunto de *clusters*. Dado un conjunto de *clusters* $\mathcal{C} \subseteq \mathcal{P}(\mathcal{D})$, el valor ICF de un rasgo $f \in Voc(\mathcal{D})$ se computa de la siguiente manera:

$$ICF(f, \mathcal{C}) = \log(\frac{|\mathcal{C}|}{|\{C \in \mathcal{C} | \exists D_i \in C : f \in D_i\}|}) \tag{5.6}$$

Nótese que ICF se trata de un valor de *especifidad* de un rasgo en el sentido de Jones [1972], de manera similar al factor IDF [Salton y Buckley, 1988].

Finalmente, la detección de los rasgos que cumplen ambas propiedades puede realizarse mediante el producto de los factores anteriores:

$$DFICF(f, C, \mathcal{C}) = DF(f, C) \cdot ICF(f, \mathcal{C}) \tag{5.7}$$

El proceso de filtrado de términos en los centroides de cada *cluster* consiste en eliminar aquellos rasgos con valores DF-ICF más bajos. En particular, el filtrado propuesto se realiza de la siguiente manera:

- Calcular los valores DF-ICF de los rasgos de cada *cluster*.
- Para cada *cluster* $C \in \mathcal{C}$ hacer lo siguiente:
  - Calcular la mediana de los valores DF-ICF de los rasgos de cada *cluster*, i.e.

$$Me(C, \mathcal{C}) = Me\{DFICF(f, C, \mathcal{C}) | f \in Voc(C)\} \tag{5.8}$$

  - Eliminar del centroide los rasgos con valores DF-ICF más bajos: un rasgo $f \in Voc(C)$ se filtra del centroide si cumple que $DFICF(f, C, \mathcal{C}) < Me(C, \mathcal{C})$.

Los valores DF-ICF se emplean solamente para seleccionar los rasgos más relevantes de acuerdo al criterio anterior, de manera que el truncamiento DF-ICF es independiente de la función de pesado de términos empleada, a diferencia del truncamiento por peso. Por otro lado, el truncamiento DF-ICF emplea la mediana al igual que el truncamiento por peso por las siguientes razones: (i) evita seleccionar un número prefijado de rasgos; y (ii) evita casos extremos a diferencia de otros estadísticos como la media aritmética.

En lo sucesivo, denominaremos a los centroides obtenidos mediante el método de truncamiento anterior como *centroides DF-ICF* y emplearemos la notación $\delta(C)$ para denotar el centroide DF-ICF del *cluster* C.

5.3.2. Pseudocódigo

La estrategia de *clustering* empleada durante la fase 3 consiste en mezclar los *clusters* más similares entre sí tales que cumplan la condición de agrupamiento del umbral adaptativo.

El Algoritmo 5.4 muestra el pseudocódigo para obtener el *cluster* más similar de otro dado por parámetro. Se ha empleado la notación $CT(C_i)$ para identificar al centroide del *cluster* C_i. En particular, CT puede ser los centroides μ, ω o δ descritos anteriormente. Dado un *cluster* C_j, se compara su similitud con respecto a cada *cluster* diferente a él (líneas 3-4) representando ambos mediante sus centroides de tipo CT. Esta comparación se realiza mediante una cierta medida de similitud *sim* y un umbral adaptativo γ. Nótese que los centroides pueden ser vistos como documentos, de modo que pueden aplicarse sobre ellos tanto las medidas de similitud como el umbral adaptativo considerados en esta tesis. En caso de que se cumpla la condición de agrupamiento, se comprueba que el *cluster* actual sea más similar que los anteriores (línea 5), y en ese caso se guarda (línea 7). Finalmente, se devuelve el *cluster* C_{sim} más similar al *cluster* C_j dado como parámetro (línea 11). En caso de que $C_{sim} = \emptyset$, significa C_j no cumple la condición de agrupamiento con ninguno de los *clusters*.

Algoritmo 5.4 mejorCluster$(C, \mathcal{C}, sim, \gamma, n, CT)$.

Entrada: *Cluster* C_j, conjunto de *clusters* $\mathcal{C} = \{C_1, C_2, \ldots, C_p\}$ tal que $C_j \in \mathcal{C}$, medida de similitud *sim*, umbral adaptativo γ, $n \in \mathbb{N}$, donde n es la longitud de los rasgos empleados y CT es el tipo de centroides utilizados (por ejemplo, μ, ω o δ).

Salida: *Cluster* C_{sim} más similar a C tal que cumple la condición de agrupamiento del umbral adaptativo γ.

1: $C_{sim} = \emptyset$
2: $maxSim = -\infty$
3: **para** $C_k \in \mathcal{C}$, $k \neq j$ **hacer**
4: **si** $sim^n(CT(C_j), CT(C_k)) > \gamma^n(CT(C_j), CT(C_k))$
5: **si** $sim^n(CT(C_j), CT(C_k)) > maxSim$
6: $maxSim = sim^n(CT(C_j), CT(C_k))$
7: $C_{sim} = C_k$
8: **fin si**
9: **fin si**
10: **fin para**
11: **devolver** C_{sim}

El Algoritmo 5.5 muestra el pseudocódigo de la fase de mezcla de *clusters* de ATC. El algoritmo parte del conjunto de *clusters* $\mathcal{C}$ generado tras las fases 1 y 2. Se ha empleado la notación $CT(\mathcal{C})$ para identificar al conjunto de centroides de tipo CT de los *clusters*

incluidos en $\mathcal{C}$: por ejemplo, μ, ω o δ. En primer lugar, se calculan los centroides de tipo CT de los *clusters* mediante la función *computarCentroides* (línea 1). A continuación, se recorre cada *cluster* y se computa su *cluster* más similar mediante la función *mejorCluster* (línea 3) que aplica el Algoritmo 5.4. En caso de que el *cluster* devuelto por esa función no sea vacío (línea 4), se mezclan ambos *clusters* (líneas 5-6) y se recalculan los centroides (línea 7). Finalmente, el algoritmo devuelve el conjunto de *clusters* generado tras aplicar las tres fases de ATC (línea 10).

Algoritmo 5.5 fusionarClusters(C, sim, γ, n, CT).

Entrada: Conjunto de *clusters* iniciales $\mathcal{C} = \{C_1, C_2, \ldots, C_p\}$, medida de similitud sim, umbral adaptativo γ, $n \in \mathbb{N}$, donde n es la longitud de los rasgos empleados y CT es el tipo de centroides utilizados (por ejemplo, μ, ω o δ).

1: $CT(\mathcal{C}) =$ computarCentroides(CT,$\mathcal{C}$)
2: **para** $C_j \in \mathcal{C}$ **hacer**
3: $\quad C_{sim} =$ mejorCluster$(C_j, \mathcal{C}, sim, \gamma, n, CT)$
4: $\quad$ **si** $C_{sim} \neq \emptyset$
5: $\quad\quad C_j = C_j \cup C_{sim}$
6: $\quad\quad \mathcal{C} = \mathcal{C} \setminus \{C_{sim}\}$
7: $\quad\quad CT(\mathcal{C}) =$ computarCentroides(CT,$\mathcal{C}$)
8: $\quad$ **fin si**
9: **fin para**
10: **devolver** $\mathcal{C}$

A continuación, analizamos un par de propiedades del algoritmo anterior:

Determinismo

El orden en el que se compara cada par de centroides puede afectar en el rendimiento del algoritmo, puesto que cada vez que hay una mezcla de *clusters*, los centroides se recalculan. Como vimos anteriormente, esta situación no suponía ningún problema en el caso de las fases 1 y 2. Para asegurar el determinismo en la fase 3 de ATC se debe establecer un orden de comparación entre centroides. Se ha seguido el criterio de comparar en primer lugar aquellos *clusters* que contienen los primeros resultados de búsqueda del ranking devuelto por el buscador: siguiendo la notación del Capítulo 4, el resultado de búsqueda que aparece en la i-ésima posición de ranking se denota mediante W_i. Por tanto, dado un *cluster* $C_j = \{W_{j_1}, W_{j_2}, \ldots, W_{j_m}\}$, el índice j del *cluster* es $j = \text{mín}\{j_i, j_2, \ldots, j_m\}$.

Coste computacional

El coste temporal del algoritmo se encuentra en $\mathcal{O}(p^2)$, donde p es el número de *clusters* iniciales, puesto que se recorren todos los *clusters* y el Algoritmo 5.4 realiza $p-1$ comparaciones por cada *cluster*. Dado que $p \leq N$, siendo N el número de resultados de

búsqueda, entonces el coste de la fase 3 en el caso peor se encuentra en $\mathcal{O}(N^2)$. Nótese que esta es la complejidad computacional de todas las fases del algoritmo ATC, de modo que la complejidad del algoritmo completo también está en $O(N^2)$. En particular, ATC tiene la misma complejidad computacional que HAC bajo determinadas condiciones: por ejemplo, uso de las políticas de enlace simple o completo.

5.3.3. Estudio de la configuración de la fase de fusión de *clusters*

En este apartado se estudia la configuración del algoritmo 5.5 empleado en la fase de fusión de *clusters* de ATC. En particular, se evaluará el impacto del tipo de centroide utilizado y de la función de pesado usada para pesar los 1-gramas. La justificación del estudio de este último factor viene dada porque, como se explicó en el apartado 4.3.2, el impacto de la función de pesado sobre 3-gramas en mayúsculas es irrelevante debido a que estos rasgos generan un vocabulario muy pequeño, de modo que las agrupaciones se realizan en base a que se compartan unos pocos rasgos con independencia del peso que tengan asignado. En cambio, la representación mediante 1-gramas genera un vocabulario que puede ser de un orden de magnitud superior con respecto al obtenido por 3-gramas en mayúsculas, como muestra la Tabla 4.1 en el caso de la colección de entrenamiento de WePS-1.

A continuación se resume la configuración utilizada para el resto de factores en los experimentos que se llevaran a cabo:

- Las fases 1 y 2 emplean la configuración detallada en los apartados anteriores.
- Los resultados de búsqueda se representan mediante 1-gramas en la fase 3.
- Se emplea la función de umbral adaptativo mostrada en la fórmula 4.7.
- Se utiliza la medida de similitud coseno.

La Tabla 5.3 muestra los resultados obtenidos por ATC empleando distintos tipos de centroides y diferentes funciones de pesado de términos en la fase 3 en las colecciones de test de WePS. Se han efectuado los siguientes estudios de significancia estadística, que deben leerse para cada colección de manera independiente:

- **Mismos centroides / Distinto peso:** este estudio permite analizar el rendimiento de las funciones de pesado de términos dado un tipo de centroide para representar los *clusters*. La tabla muestra los resultados del test de significancia estadística empleando la notación utilizada en las Tablas 5.1 y 5.2.

- **Distintos centroides / Mismo peso:** este estudio permite analizar el rendimiento del tipo de centroide empleado con independencia de la función de pesado de términos utilizada. La tabla resalta en negrita los valores de medida-F de los centroides que obtienen los mejores resultados, i.e., mejoran significativamente los resultados de los centroides cuyos valores de medida-F no estan resaltados en negrita y obtienen resultados similares con respecto a los centroides cuyos valores de medida-F están resaltados en negrita.

	CENTROIDES TEÓRICOS											
	Bin			**TF**			**TF-IDF**			**z-score**		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	0.27	0.93	0.37 (3)	0.31	0.92	0.41 (2)	0.46	0.88	0.55 (1)	0.50	0.86	0.57 (1)
WePS-2	0.47	0.95	0.57 (3)	0.51	0.95	0.61 (2)	0.59	0.93	0.68 (1)	0.64	0.90	0.70 (1)
WePS-3	0.25	0.95	0.35 (3)	0.25	0.94	0.36 (2)	0.27	0.91	0.38 (1)	0.29	0.90	0.39 (1)
	CENTROIDES CON TRUNCAMIENTO POR PESO											
	Bin			**TF**			**TF-IDF**			**z-score**		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	0.48	0.87	0.57 (3)	0.59	0.86	0.66 (2)	0.72	0.81	0.75 (1)	0.75	0.77	**0.75 (1)**
WePS-2	0.59	0.93	0.68 (3)	0.70	0.90	**0.76 (2)**	0.83	0.84	0.82 (1)	0.85	0.77	**0.80 (1)**
WePS-3	0.27	0.91	0.38 (4)	0.30	0.86	0.41 (3)	0.43	0.74	0.49 (2)	0.52	0.64	**0.52 (1)**
	CENTROIDES DF-ICF											
	Bin			**TF**			**TF-IDF**			**z-score**		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	0.59	0.85	**0.66 (3)**	0.67	0.83	**0.71 (2)**	0.76	0.82	**0.78 (1)**	0.76	0.78	**0.76 (1)**
WePS-2	0.71	0.89	**0.77 (3)**	0.70	0.90	**0.76 (2)**	0.83	0.86	**0.84 (1)**	0.85	0.77	**0.81 (2)**
WePS-3	0.34	0.84	**0.50 (3)**	0.39	0.80	**0.52 (2)**	0.57	0.63	**0.55 (1)**	0.59	0.59	**0.53 (2)**

Tabla 5.3: Resultados obtenidos por ATC en las colecciones de test de WePS empleando distintos tipos de centroides y distintas funciones de pesado de términos en la fase 3.

En cuanto al tipo de centroide empleado, los centroides teóricos obtienen peores resultados con respecto a los centroides obtenidos mediante métodos de truncamiento debido, por lo general, a un drástico descenso en los valores de precisión. Esto se debe a que los centroides teóricos benefician la relevancia de palabras comunes que aparecen en muchos resultados de búsqueda, puesto que el peso de cada rasgo del centroide se obtiene mediante la media aritmética de los pesos del rasgo en cada uno de los resultados de búsqueda incluidos en el *cluster* correspondiente. En cambio, los centroides obtenidos mediante truncamiento evitan en mayor medida el impacto negativo de los rasgos ruidosos. Por otro lado, la aplicación del truncamiento DF-ICF obtiene mejoras significativas con respecto al truncamiento por peso en todas las colecciones. Esto se explica porque el criterio del truncamiento por peso no filtra los rasgos ruidosos que normalmente tienen mayor peso por aparecer en un mayor número de documentos.

Sin embargo, los centroides DF-ICF evitan en mayor medida este tipo de rasgos gracias al factor ICF. Además, las diferencias entre los resultados obtenidos por los centroides DF-ICF con diferentes funciones de pesado de términos son menores con respecto a los otros dos tipos de centroides. Esto implica que la selección de rasgos llevada a cabo por el truncamiento DF-ICF es adecuada. En lo sucesivo, asumiremos el uso de los centroides DF-ICF en la fase 3 de ATC.

En cuanto a las funciones de pesado de términos, la tabla muestra que las funciones locales (pesado binario y TF) no mejoran los resultados obtenidos por las funciones globales (TF-IDF y z-score). Esto se explica porque las funciones locales benefician a los rasgos ruidosos referidos anteriormente. El impacto de las funciones locales en esta colección consiste en que hay un drástico descenso en los valores de precisión y una mejora en los valores de cobertura, lo cuál significa que se agrupan *clusters* que mencionan a individuos diferentes. En cambio, las funciones globales atenúan su impacto debido a que asignan un menor peso a los rasgos que aparecen en muchos documentos. En cuanto a las funciones de pesado de términos globales, TF-IDF obtiene mejoras significativas con respecto a z-score cuando se emplean centroides DF-ICF en todas las colecciones. Además, TF-IDF requiere menos coste computacional puesto que solamente requiere conocer el número de documentos en los que aparece cada rasgo, mientras que z-score computa la media aritmética y la desviación típica de las frecuencias de cada rasgo en todas las páginas web. Por tanto, es más conveniente el uso de TF-IDF por razones de rendimiento y eficiencia computacional. En lo sucesivo se asumirá el empleo de esta función de pesado de términos en la fase 3 de ATC.

En general, los errores en las agrupaciones de la fase 3 se deben a información ruidosa contenida en los resultados de búsqueda aislados que no ha sido filtrada por los centroides DF-ICF. Esta información ruidosa puede clasificarse en dos grupos:

- **Vocabulario de Internet:** estos rasgos normalmente se corresponden con palabras que aparecen en formularios de registro, anuncios, o términos legales sobre la política de privacidad de las páginas web.

- **Vocabulario común:** se trata de palabras empleadas frecuentemente que no son identificadas como palabras vacías como, por ejemplo, los días de la semana o los meses del año. Además, dentro de este grupo de *tokens* también se incluyen números que no ayudan a distinguir entre diferentes individuos. Por ejemplo, muchas páginas web indican el año actual que es común en el momento en el que se realiza una consulta en un buscador.

Podemos extraer las siguientes conclusiones a partir de los resultados de la tabla:

- Los centroides DF-ICF son más adecuados para representar el contenido de los *clusters* debido a las siguientes razones:
  - Realizan un proceso de filtrado de rasgos ruidosos que permite obtener mejoras significativas con respecto a los otros dos tipos de centroides considerados.
  - Presentan una mayor independencia con respecto a la función de pesado de términos empleada.
- Las funciones de pesado globales (TF-IDF y z-score) son más adecuadas a la hora de asignar la relevancia de los 1-gramas. Esto se debe a que asignan un menor peso a los rasgos ruidosos que no han sido filtrados por los centroides DF-ICF.
- TF-IDF y z-score presentan resultados similares cuando se emplean centroides DF-ICF. No obstante, el cálculo de TF-IDF es menos costoso y, además, obtiene mejores significativas con respecto a z-score en todas las colecciones. Por tanto, TF-IDF es la función más adecuada para pesar los 1-gramas en la fase 3 de ATC.

La Tabla 5.4 resume la configuración empleada por ATC en cada una de las fases. En particular, en las fases 2 y 3 se emplea la similitud coseno y la función de umbral adaptativo presentada en el Capítulo 4. El símbolo - indica que en la fase correspondiente no se requiere especificar el factor.

Algoritmo	Rasgos	Política de enlace	Función de pesado	Centroide
FASE 1	URL y *links*	Enlace Indirecto	-	-
FASE 2	3-gramas en mayúsculas	-	*Bin*	-
FASE 3	1-gramas	-	TF-IDF	DF-ICF

Tabla 5.4: Configuración de ATC.

5.3.4. Resultados

Una vez seleccionada la configuración final del algoritmo ATC, se puede comparar la evaluación de los resultados obtenidos para cada una de sus fases. La Tabla 5.5 muestra los resultados obtenidos en las tres fases del algoritmo ATC empleando la configuración detallada en la Tabla 5.4 con las colecciones de test de WePS, junto al estudio de significancia estadística entre cada una de las fases empleando la notación de marcas (k) con $k \in \mathbb{N}$ explicada con anterioridad.

La mejora de los resultados en la fase 2 con respecto a la fase 1 se analizó en el apartado 5.2.2. Por otro lado, la tabla muestra que a medida que se avanza en las fases,

FASE	FASE 1			FASE 2			FASE 3		
Colección	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
WePS-1	0.94	0.54	0.67 (3)	0.81	0.74	0.76 (2)	0.76	0.82	0.78 (1)
WePS-2	0.96	0.36	0.48 (3)	0.88	0.77	0.81 (2)	0.83	0.86	0.84 (1)
WePS-3	0.80	0.35	0.45 (3)	0.62	0.55	0.53 (2)	0.57	0.63	0.55 (1)

Tabla 5.5: Resultados obtenidos por cada una de las fases de ATC en las colecciones de test de WePS.

los resultados mejoran significativamente con respecto a la fase anterior. En particular, la aportación de la fase 3 consiste en una mejora de los valores de cobertura sin un descenso drástico de los valores de precisión con respecto a los obtenidos en la fase 2, lo que indica que la mayoría de las agrupaciones realizadas durante la fase 3 son correctas. Se confirma entonces que el empleo de los centroides DF-ICF y el pesado TF-IDF es adecuado en esta fase.

5.4. Comparativa con otros sistemas

En esta sección se compararán los resultados del algoritmo ATC con respecto a los obtenidos por los *baselines*, los sistemas del estado del arte y el algoritmo UPND en las colecciones WePS.

Las Tablas 5.6, 5.7 y 5.8 muestran los resultados promedios de las métricas *B-Cubed* obtenidos por ATC (resaltados en color verde), otros sistemas del estado del arte, los *baselines* y UPND (resaltados en rojo) para las colecciones de test de la campañas de evaluación WePS-1, WePS-2 y WePS-3, respectivamente. Los resultados aparecen ordenados con respecto al valor promedio $F_{0,5}$, empleado como métrica oficial de las campañas WePS. Las tablas indican el tipo de sistema de la siguiente manera: P identifica a los sistemas participantes en las campañas WePS, TP identifica los trabajos presentados con posterioridad a las campañas WePS y B identifica a los *baselines*. Además, las tablas indican si cada sistema necesita datos de entrenamiento y muestra el estudio de significancia estadística siguiendo la siguiente simbología:

- $\uparrow\uparrow$: indica que ATC obtiene mejoras significativas con respecto al sistema.
- $\downarrow\downarrow$: indica que el sistema obtiene mejoras significativas con respecto a ATC.
- $=$: indica que el sistema obtiene resultados similares a los de ATC.
- **?**: indica que no se ha podido realizar la comparación con el test de significancia estadística puesto que no se ha tenido acceso a los resultados obtenidos por el

sistema para cada nombre de persona, a pesar de tener acceso a los resultados promedios de las métricas de evaluación.

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Liu et al. [2011]	TP	SI	0.79	0.85	0.81	⇊
ATC	TP	NO	0.76	0.82	0.78	=
Jiang et al. [2009]	TP	SI	0.80	0.79	0.78	*?*
UPND	TP	NO	0.85	0.70	0.76	⇈
Chen y Martin [2007a]	P	SI	0.61	0.83	0.70	⇈
Elmacioglu et al. [2007]	P	SI	0.68	0.73	0.70	⇈
Popescu y Magnini [2007]	P	SI	0.68	0.71	0.69	⇈
Saggion [2007]	P	SI	0.54	0.74	0.62	⇈
Balog et al. [2007]	P	SI	0.79	0.50	0.61	⇈
ONE IN ONE	B	NO	1.00	0.43	0.57	⇈
Ellman y Emery [2007]	P	SI	0.59	0.63	0.57	⇈
AP	B	NO	0.80	0.53	0.55	⇈
Kalmar y Blume [2007]	P	SI	0.43	0.84	0.53	⇈
Lefever et al. [2007]	P	SI	0.42	0.80	0.51	⇈
Kozareva et al. [2007]	P	NO	0.54	0.53	0.49	⇈
Rao et al. [2007]	P	NO	0.36	0.73	0.43	⇈
Iria et al. [2007]	P	NO	0.28	0.88	0.39	⇈
Sugiyama y Okumura [2007]	P	SI	0.29	0.82	0.38	⇈
Heyl y Neumann [2007]	P	SI	0.30	0.74	0.38	⇈
del Valle-Agudo et al. [2007]	P	SI	0.26	0.91	0.36	⇈
ALL IN ONE	B	NO	0.18	0.98	0.28	⇈

Tabla 5.6: Resultados obtenidos por los algoritmos UPND y ATC, los sistemas del estado del arte y los *baselines* sobre la colección de test de WePS-1.

Los resultados de las Tablas 5.6, 5.7 y 5.8 nos indican lo siguiente:

- Los resultados de ATC obtienen mejoras significativas con respecto a los *baselines* ONE IN ONE, ALL IN ONE y con respecto al algoritmo *Affinity Propagation* (AP), que no necesita datos de entrenamiento.

- ATC obtiene los mejores resultados entre todos los sistemas que no requieren datos de entrenamiento en las tres colecciones WePS. En el caso de WePS-2, no se ha podido aplicar el test de Wilcoxon entre ATC y el sistema propuesto por Xu et al. [2015], que es el mejor sistema del estado del arte de este tipo, debido a que

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Yoshida et al. [2010]	TP	SI	0.89	0.82	0.85	*?*
ATC	TP	NO	0.83	0.86	0.84	=
Jiang et al. [2009]	TP	SI	0.85	0.83	0.83	*?*
Chen et al. [2009]	P	SI	0.87	0.79	0.82	↑↑
Balog et al. [2009]	P	SI	0.85	0.80	0.81	↑↑
Ikeda et al. [2009]	P	SI	0.93	0.73	0.81	↑↑
Xu et al. [2015]	TP	NO	0.88	0.78	0.81	*?*
UPND	TP	NO	0.92	0.70	0.79	↑↑
Romano et al. [2009]	P	SI	0.82	0.66	0.72	↑↑
Kalmar y Freitag [2009]	P	SI	0.85	0.62	0.70	↑↑
Gong y Oard [2009]	P	SI	0.94	0.60	0.70	↑↑
Song et al. [2009]	P	SI	0.54	0.93	0.63	↑↑
Han y Zhao [2009]	P	SI	0.65	0.75	0.63	↑↑
Lefever et al. [2009]	P	SI	0.73	0.58	0.57	↑↑
González et al. [2009]	P	SI	0.60	0.66	0.56	↑↑
ALL IN ONE	B	NO	0.43	1.00	0.53	↑↑
AP	B	NO	0.82	0.39	0.44	↑↑
Lan et al. [2009]	P	NO	0.50	0.55	0.41	↑↑
Martínez-Romo y Araujo [2009]	P	NO	0.66	0.39	0.40	↑↑
Venkateshan [2009]	P	NO	0.61	0.38	0.39	↑↑
ONE IN ONE	B	NO	1.00	0.24	0.34	↑↑
Pinto et al. [2009]	P	NO	0.89	0.25	0.33	↑↑

Tabla 5.7: Resultados obtenidos por los algoritmos UPND y ATC, los sistemas del estado del arte y los *baselines* sobre la colección WePS-2.

no se ha tenido acceso a los resultados obtenidos para cada nombre de persona. No obstante, ATC mejora significativamente los resultados de otros sistemas que obtienen resultados similares a este método, como los propuestos por Ikeda et al. [2009], Balog et al. [2009] y Chen et al. [2009]. Además, ATC mejora significativamente los resultados de UPND en las tres colecciones y al mismo tiempo, resuelve el principal inconveniente de este algoritmo con respecto al porcentaje de páginas web representadas.

- En el caso de la colección de test de WePS-1 (ver Tabla 5.6), ATC obtiene mejoras significativas con respecto a todos los sistemas participantes y UPND. No obstante, el sistema propuesto por Liu et al. [2011] mejora significativamente los resultados de ATC. Como se comentó en el Capítulo 4, este sistema es el único que ha sido en-

Sistema	Tipo	Entr.	BP	BR	$F_{0,5}$	Wilcoxon
Long y Shi [2010]	TP	SI	0.61	0.60	0.55	=
ATC	TP	NO	0.57	0.63	0.55	=
UPND	TP	NO	0.63	0.53	0.52	⇈
Smirnova et al. [2010]	P	SI	0.69	0.46	0.50	⇈
Ferrés y Rodríguez [2010]	P	SI	0.40	0.66	0.44	⇈
Nagy [2012]	TP	SI	0.38	0.61	0.40	⇈
Dornescu et al. [2010]	P	SI	0.31	0.80	0.40	⇈
Lana-Serrano et al. [2010]	P	NO	0.29	0.84	0.39	⇈
AP	B	NO	0.75	0.31	0.39	⇈
ONE IN ONE	B	NO	1.00	0.23	0.35	⇈
ALL IN ONE	B	NO	0.22	1.00	0.32	⇈

Tabla 5.8: Resultados obtenidos por los algoritmos UPND y ATC, los sistemas del estado del arte y los *baselines* sobre la colección WePS-3.

trenado con la colección WePS-2, a diferencia del resto de sistemas que requieren entrenamiento, que han empleado la colección de entrenamiento de WePS-1. Este hecho puede explicar que este sistema obtenga mejores resultados, puesto que la colección WePS-2 es más similar a la colección de test de WePS-1, tanto en el grado de ambigüedad de los nombres de personas como en el número de resultados de búsqueda por nombre de persona. Por otro lado, no se ha podido comparar mediante el test de Wilcoxon los resultados de ATC y el sistema propuesto por Jiang et al. [2009] debido a que no se ha tenido acceso a sus resultados para cada nombre de persona. No obstante, ATC y este sistema obtienen el mismo valor de medida-F, pero ATC no necesita utilizar información de los datos de entrenamiento.

- En cuanto a la colección WePS-2 (ver Tabla 5.7), ATC también obtiene mejoras significativas con respecto a todos los sistemas participantes y UPND. No se ha podido aplicar el test de significancia estadística con respecto a los dos mejores sistemas del estado del arte propuestos por Jiang et al. [2009] y Yoshida et al. [2010] debido a que no proporcionan los resultados para cada nombre de persona. Nótese que ambos sistemas emplean una estrategia de *clustering* similar a ATC, consistente en generar *clusters* iniciales mediante rasgos poco ruidosos y, posteriormente, refinan dichos *clusters* mediante una representación basada en BoW. En particular, Jiang et al. [2009] generan los *clusters* iniciales empleando información biográfica, NEs, *links*, números de teléfonos y *e-mails*. Por su parte, Yoshida et al. [2010] dividen los rasgos entre *fuertes* y *débiles*, de modo que los *clusters* iniciales se obtienen mediante el primer tipo de rasgos, en donde se encuentran los *links*, las

NEs, y *keywords* compuestas por varias palabras y pesados según su cercanía con el nombre de persona consultado. Dado que ATC emplea una estrategia similar a estos dos sistemas, existe un paralelismo entre los rasgos empleados en cada fase: por un lado, los *clusters* iniciales se obtienen mediante *links* y rasgos compuestos por varias palabras, como las NEs o palabras compuestas en el caso de estos sistemas, o 3-gramas en mayúsculas en el caso de ATC. En cambio, se emplean 1-gramas para generar los *clusters* resultantes. Sin embargo, estos dos sistemas requieren datos de entrenamiento para aprender umbrales de similitud, a diferencia de ATC.

- Finalmente, con la colección WePS-3 (ver Tabla 5.8), ATC mejora los resultados de UPND y todos los sistemas excepto los obtenidos por el mejor participante de esta campaña [Long y Shi, 2010]. Este sistema basado en HAC requiere datos de entrenamiento y emplea conceptos extraídos manualmente de Wikipedia. Recuérdese que los valores de las métricas de todos los sistemas son más bajos con respecto a las otras dos colecciones, puesto que solo se evalúa la calidad de los *clusters* de uno o dos individuos por nombre de persona.

5.5. Conclusiones

Las principales conclusiones extraídas en este capítulo son las siguientes:

- Se ha presentado el algoritmo de *clustering* ATC para la desambiguación de nombres de personas en la Web. La estrategia de agrupamiento de resultados de búsqueda empleada por ATC es la siguiente:

  - Generar *clusters* iniciales caracterizados por tener un alto grado de precisión a partir de rasgos poco comunes. En particular, ATC genera estos *clusters* representando los resultados de búsqueda mediante su URL y sus *links* (fase 1) y 3-gramas en mayúsculas (fase 2).
  - Fusionar los *clusters* iniciales mediante el empleo de rasgos que suelen coaparecer más frecuentemente entre diferentes documentos. En particular, ATC emplea 1-gramas (fase 3), los cuales garantizan la representación del contenido de tantos resultados de búsqueda como sea posible.

  ATC se asemeja a UPND porque estima automáticamente el número de *clusters* y se basa en el uso de umbrales adaptativos para comparar resultados de búsqueda, de modo que no requiere aprender ningún parámetro a partir de datos de entrenamiento. Ambos algoritmos se diferencian porque ATC garantiza que se

representan tantas páginas web como sea posible debido a que emplea diferentes tipos de rasgos en cada fase, mientras que UPND solamente emplea 3-gramas en mayúsculas y no logra representar un alto porcentaje de resultados de búsqueda.

- Las políticas de agrupamiento por *links* tienen un impacto positivo en los resultados y generalmente agrupan correctamente resultados de búsqueda que hablan del mismo individuo. El agrupamiento mediante enlace directo logra resultados muy altos de precisión pero no siempre logra mejoras significativas con respecto a no aplicarla. En cambio, la política de enlace indirecto obtiene mejoras significativas aunque realiza un mayor número de agrupaciones erróneas debidas a *links* a páginas web populares o de tecnologías y servicios empleados por las páginas web.

- Las funciones de pesado globales son más adecuadas para asignar el peso de los 1-gramas que las funciones de pesado locales. En cambio, la función de pesado empleada no tiene impacto en el caso de los 3-gramas en mayúsculas. Esto se debe a que los 1-gramas coaparecen más frecuentemente que los 3-gramas en mayúsculas y contienen una mayor proporción de información ruidosa. Las funciones de pesado globales asignan un menor peso a los rasgos ruidosos, de modo que atenúan su impacto cuando se comparan los documentos. Las funciones TF-IDF y z-score obtienen resultados similares, pero el cálculo de la primera es menos costoso computacionalmente, por lo que resulta más adecuada.

- Los centroides DF-ICF empleados en este trabajo evitan agrupaciones de *clusters* incorrectas debido a que realizan un proceso de truncamiento y filtrado de rasgos ruidosos. Además, los resultados obtenidos al aplicar este tipo de centroides son más independientes con respecto a la función de pesado de términos empleada.

- ATC obtiene mejoras en los resultados en todas las colecciones WePS con respecto a los *baselines*, los sistemas del estado del arte que no requieren datos de entrenamiento y el algoritmo UPND. Además, ATC obtiene resultados similares con respecto a la mayoría de los mejores sistemas del estado del arte que emplean parámetros prefijados o aprendidos mediante datos de entrenamiento. Por tanto, podemos concluir que los umbrales adaptativos son una técnica adecuada para comparar los resultados de búsqueda en el problema de desambiguación de nombres de personas en la Web.

- El algoritmo ATC emplea una estrategia de *clustering* similar a la de algunos de los mejores sistemas del estado del arte [Jiang et al., 2009; Yoshida et al., 2010]. Por ello, podemos concluir que esta metodología es adecuada para el problema de desambiguación de nombres de personas en la Web.

Pese a que ATC obtiene resultados competitivos, el algoritmo realiza agrupaciones incorrectas que podemos dividir en dos tipos:

- Agrupaciones incorrectas debidas a enlaces a determinados tipos de páginas web (fase 1).
- Agrupaciones incorrectas debidas a información ruidosa (fases 2 y 3).

En cuanto al primer tipo de error, Bekkerman y McCallum [2005] proponen evitar *links* a URLs de dominios ampliamente visitados en Internet como `www.amazon.com`. No obstante, se ha observado que los errores de este tipo también se deben a *links* a páginas web de naturaleza muy diversa (periódicos, enciclopedias, *blogs*, servicios de monitorización en la Red, ...), de modo que es complicado establecer un criterio que englobe a todos estos tipos de páginas web.

Los errores del segundo tipo son debidos a dos grupos de palabras: vocabulario empleado en Internet, y vocabulario común no identificado como palabras vacías. En cuanto al primer tipo de palabras, Nagy [2012] trata de evitar su impacto negativo mediante el uso de una lista de palabras vacías elaborada manualmente que contiene términos comunes en la Web como *webmaster* o *support*. En cuanto al segundo grupo de palabras, Jiang et al. [2009] emplean una lista de palabras vacías compuesta por 150.000 palabras, mucho más extensa que las utilizadas habitualmente. Una posibilidad para evitar ambos grupos de palabras consiste en identificarlas mediante técnicas de filtrado de términos durante la fase de preprocesamiento. Por otro lado, algunos autores [Yoshida et al., 2010; Chen et al., 2012] otorgan un mayor peso a las palabras situadas cerca del nombre de persona consultado. Este último tipo de técnicas pueden ser especialmente útiles para atenuar el efecto del vocabulario empleado en Internet, puesto que este tipo de palabras suelen aparecer de forma separada al contenido albergado por la página web.

A pesar de que ATC obtiene resultados competitivos en las colecciones WePS, hay que señalar que en estas colecciones no se tienen en cuenta dos factores que pueden tener lugar en un escenario de búsqueda real en la actualidad:

- Las colecciones WePS albergan muy pocas páginas web consistentes en perfiles de redes sociales, dado que la popularidad de este tipo de plataformas web era mucho menor cuando fueron recopiladas. El Capítulo 6 estudia el tratamiento de las redes sociales en el problema utilizando dos colecciones recopiladas más recientemente.
- Las colecciones WePS asumen un escenario monolingüe pese a que la Web cada vez alberga una mayor cantidad de contenido en diversos idiomas [Pimienta et al.,

2009]. El Capítulo 7 estudia el tratamiento del multilingüísmo en el problema utilizando una colección caracterizada por contener páginas web escritas en diferentes idiomas.

6
Tratamiento de las redes sociales

"La Web es más una creación social que tecnológica."

— Tim Berners-Lee —

Los perfiles de redes sociales aparecen frecuentemente como resultados de búsqueda cuando se consulta un nombre de persona a un buscador debido al éxito de estos servicios web en los últimos años. De acuerdo al estado del arte, la presencia de este tipo de páginas web puede impactar negativamente en los sistemas de desambiguación de nombres de persona y deben proponerse políticas que las traten de manera diferenciada con respecto al resto de resultados de búsqueda. No obstante, la mayoría de los sistemas del estado del arte se han evaluado con colecciones que contienen muy pocas páginas web de este tipo y no han tenido en cuenta su impacto. Por otro lado, las escasas políticas propuestas para tratar las redes sociales en el problema no toman en consideración situaciones habituales como que un mismo individuo tenga perfiles en varias redes sociales. En este capítulo, se presentan tres nuevas heurísticas para tratar este tipo de páginas web en el problema, de manera que resuelven las limitaciones de las propuestas del estado del arte. Además, se evalúa la efectividad de las heurísticas propuestas utilizando las colecciones de datos disponibles que contienen un mayor porcentaje de páginas sociales. Finalmente, se resumen las principales conclusiones extraídas en este capítulo.

6.1. Introducción

El reciente éxito de las redes sociales ha supuesto que sea común encontrar este tipo de páginas web dentro de los ranking de resultados devueltos por los motores de búsqueda cuando se les realiza una consulta consistente en un nombre de persona. Por esta razón, debe tenerse en cuenta el papel de este tipo de páginas web a la hora de afrontar un escenario de búsqueda en la Web como el tratado en la presente tesis. Estos servicios web constituyen un ejemplo de la heterogeneidad de la Web. Por un lado, *Facebook*[1] o *Twitter*[2] son redes sociales en las que los usuarios habitualmente publican in-

[1] `https://www.facebook.com/`

[2] `https://twitter.com/`

formación de índole personal, mientras que *LinkedIn*[3] o *XING*[4] están enfocadas al ámbito profesional. Por otra parte, algunas redes sociales están especializadas en determinadas temáticas. Por ejemplo, *ResearchGate*[5] es una red social pensada para investigadores y académicos, o *Last.fm*[6] se centra en los gustos musicales de los usuarios.

El impacto de las páginas sociales sobre la desambiguación de nombres de personas en la web no ha sido tratado por la mayoría de los trabajos del estado del arte, puesto que las colecciones de datos de referencia contienen un bajo porcentaje de este tipo de páginas web debido a que su popularidad era mucho menor en el momento en el que fueron recopiladas. Además, en algunos casos, como en la campaña de evaluación WePS-2 [Artiles et al., 2009b], este tipo de páginas web fueron excluídas de la evaluación por parte de los organizadores. Por este motivo, tanto los mejores sistemas participantes de las campañas WePS (ej. [Chen y Martin, 2007a; Balog et al., 2009; Long y Shi, 2010]), como los métodos presentados posteriormente (ej. [Liu et al., 2011; Xu et al., 2015]) evaluados con estas colecciones, no han tenido en cuenta el impacto de las páginas sociales en el problema.

Más recientemente, Berendsen et al. [2012] recopilaron la colección ECIR 2012, la cual contiene un mayor porcentaje de páginas sociales con respecto a las colecciones WePS. En particular, ECIR 2012 incluye páginas sociales de las plataformas *Facebook*, *LinkedIn*, *Twitter*, *Hyves*[7] y *Myspace*[8]. Los autores emplearon esta colección para estudiar el papel de las páginas sociales concluyendo que su presencia en los rankings de búsqueda puede tener un impacto negativo en los resultados de los sistemas de desambiguación del estado del arte y, por tanto, deben ser tratadas de manera diferenciada con respecto al resto de resultados de búsqueda.

Berendsen [2015] propone una estrategia dual, mostrada en el Algoritmo 6.6, para tratar la desambiguación de nombres de persona teniendo en cuenta las redes sociales. La estrategia consiste en dividir el conjunto de resultados de búsqueda entre *páginas no sociales* y *páginas sociales* (línea 1 del algoritmo) de acuerdo a si se corresponden con alguna de las redes sociales consideradas en la colección ECIR 2012. Posteriormente, cada uno de estos conjuntos de resultados de búsqueda se desambiguan de forma separada mediante distintos algoritmos de *clustering* (líneas 2 y 3 del algoritmo). Finalmente, se aplica un método de mezcla entre los *clusters* de ambos tipos de resultados de búsqueda (línea 4 del algoritmo) y se devuelve el conjunto de *clusters* resultantes (línea 5 del algoritmo).

[3] `https://www.linkedin.com/`
[4] `https://www.xing.com/`
[5] `https://www.researchgate.net/`
[6] `https://www.last.fm/`
[7] Antigua red social neerlandesa similar a *Facebook*.
[8] `https://myspace.com/`

Algoritmo 6.6 Estrategia dual para desambiguar nombres de personas propuesta por Berendsen [2015].

Entrada: Conjunto de páginas web $\mathcal{W}$.

Salida: Conjunto de *clusters* $\mathcal{C}$.

1: Dividir $\mathcal{W}$ en dos conjuntos $\mathcal{W}_s$ y $\mathcal{W}_{ns}$ tales que $\mathcal{W} = \mathcal{W}_s \cup \mathcal{W}_{ns}$. $\mathcal{W}_s$ contiene las páginas sociales de $\mathcal{W}$, mientras que $\mathcal{W}_{ns}$ contiene el resto de páginas web.
2: Obtener un conjunto de *clusters* $\mathcal{C}_{ns}$ mediante la aplicación de un algoritmo de desambiguación $\mathcal{A}_{ns}$ aplicado sobre $\mathcal{W}_{ns}$.
3: Obtener un conjunto de *clusters* $\mathcal{C}_s$ mediante la aplicación de un algoritmo de desambiguación $\mathcal{A}_s$ aplicado sobre $\mathcal{W}_s$.
4: Obtener $\mathcal{C}$ mediante un algoritmo de mezcla $\mathcal{A}_{mix}$ sobre $\mathcal{C}_{ns}$ y $\mathcal{C}_s$.
5: **devolver** $\mathcal{C}$

Por un lado, Berendsen [2015] agrupa las páginas no sociales ($\mathcal{A}_{ns}$) mediante el algoritmo HAC con política de enlace simple representando las páginas web como BoW (1-gramas) y aplicando un umbral de similitud obtenido por entrenamiento, concretamente $\gamma = 0.225$. Esto se debe a que Berendsen [2015] asume que la aplicación de este método obtiene resultados competitivos como concluyeron varios autores [Artiles et al., 2009b; Balog et al., 2009] tras las campañas de evaluación WePS.

Por otro lado, Berendsen [2015] estudió diversos métodos heurísticos para agrupar las páginas sociales ($\mathcal{A}_s$), llegando a la conclusión de que se obtienen mejores resultados cuando se establece la política de que cada red social conforma un *cluster* unitario. En lo sucesivo, nos referiremos a este tratamiento de páginas sociales como *ONE IN ONE Social* (OIOS)[9]. El resto de políticas de agrupación de páginas sociales estudiadas por Berendsen [2015] se basan en la compartición de *links* entre las páginas sociales, agrupar las páginas en base a los *clicks* realizados por los usuarios del motor de búsqueda o el *clustering* de resultados mediante una herramienta de reconocimiento de caras, proporcionada por la antigua aplicación *online Google Picasa*, aplicada sobre las fotos contenidas en las páginas web sociales.

Finalmente, Berendsen [2015] propone dos métodos de mezcla ($\mathcal{A}_{mix}$) de *clusters* sociales y no sociales:

- El primer método genera los *clusters* resultantes mediante la unión de ambos conjuntos de *clusters*, i.e. $\mathcal{C} = \mathcal{C}_{ns} \cup \mathcal{C}_s$, de manera que las páginas sociales se devuelven en *clusters* unitarios.

[9] Berendsen [2015] denomina a esta política de tratamiento de redes sociales como ONE IN ONE porque puede verse como la aplicación de este *baseline* solamente sobre las páginas sociales. No obstante, en esta tesis emplearemos la denominación OIOS para distinguirla del *baseline* ONE IN ONE que se aplica sobre todas las páginas web.

- El segundo método consiste en un algoritmo de mezcla iterativo que calcula los pares formados por un *cluster* social y un *cluster* no social, de modo que sean lo más similares entre sí y además: (i) su similitud exceda un cierto umbral $\tau \in [0, 1]$; y (ii) se penalizan los *clusters* no sociales que contienen alguna página social añadida en iteraciones anteriores, a partir del número de páginas sociales que contengan y un factor de penalización $p \in \mathbb{R}$ dado por parámetro. En particular, Berendsen [2015] prefija los siguientes valores: $tau = 0.5$ y $p = 1$.

Por un lado, Berendsen [2015] concluye que la estrategia dual que propone obtiene mejores resultados que el empleo de HAC sobre todas las páginas web, de modo que se verifica la hipótesis de que deben tratarse de manera especial las páginas sociales y su presencia puede tener un impacto negativo en los resultados de los sistemas que usan HAC para agrupar los resultados de búsqueda. Por otro lado, también concluye que los dos métodos que propone para fusionar los *clusters* sociales y no sociales obtienen resultados similares. Dado que el primer método devuelve las páginas sociales en *clusters* unitarios, esto significa que la aplicación de la política OIOS es adecuada para tratar este tipo de páginas web.

La heurística OIOS asume que cada página social se refiere a un individuo diferente, de modo que no permite agruparlas entre sí. Sin embargo, esto no es así necesariamente. De acuerdo con un estudio presentado por el *Pew Research Center*[10], un 42 % de usuarios de Internet usaban dos o más redes sociales en el año 2014. La Figura 6.1 muestra como ejemplo los perfiles del Presidente del Gobierno de España, *Mariano Rajoy*, en las redes sociales *Facebook*, *LinkedIn*, *Instagram*[11] y *Twitter*. Siguiendo la política OIOS, estas páginas web no podrían agruparse entre sí aunque se refieran al mismo individuo.

Por otra parte, es posible que se mencione a un mismo individuo en varias páginas web de la misma red social por diferentes motivos. Por ejemplo, otros usuarios pueden mencionarle (ej. *retweets* en *Twitter*), puede participar en algún grupo específico de una red social (ej. grupos temáticos de *Facebook*) o puede tener varios perfiles en la misma red social. Esta última situación es especialmente habitual en el caso de las celebridades, puesto que en una misma red social es común encontrar diferentes perfiles dedicados a apoyarles (ej. clubs de *fans*), criticarles o parodiarles. La Figura 6.2 ilustra esta situación mostrando varios perfiles de *Twitter* de este tipo sobre *Mariano Rajoy*.

La política OIOS no permite agrupar entre sí las páginas web en los casos anteriores. Dado que se trata de la única propuesta del estado del arte para el tratamiento de las

[10] `http://www.pewinternet.org/~/media//Files/Reports/2013/Social%20Networking%202013_PDF.pdf`. Fecha de acceso: 20/03/2017

[11] `https://www.instagram.com/`

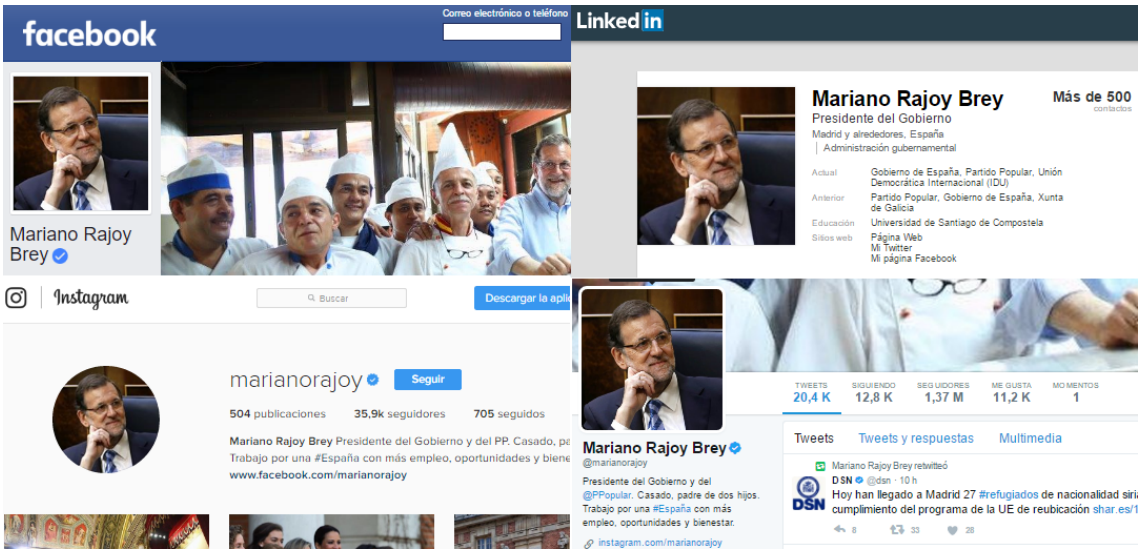

Figura 6.1: Varias redes sociales del Presidente del Gobierno de España, *Mariano Rajoy*.

Figura 6.2: Diferentes cuentas de *Twitter* de apoyo o parodia al Presidente del Gobierno de España, *Mariano Rajoy*.

redes sociales en el problema, es necesario estudiar nuevas propuestas de tratamiento de las páginas sociales que eviten las limitaciones de la política OIOS.

6.2. Tipología de las páginas sociales para la desambiguación

Las páginas sociales devueltas por los motores de búsqueda cuando se consulta un nombre de persona pueden dividirse en tres tipos:

- **Perfil de usuario:** se trata del perfil de un usuario de la red social.
- **Grupos:** se trata de páginas de una red social en las que un grupo de usuarios hablan sobre una determinada temática. Por ejemplo, *Facebook* contiene grupos de admiradores de artistas y deportistas, mientras que *LinkedIn* tiene grupos de usuarios que estudiaron en una determinada universidad.
- **Listado de usuarios:** se trata de una lista de usuarios de una cierta red social que se llaman igual.

El último tipo de páginas web es especialmente problemático, puesto que enlaza a los perfiles de distintos individuos con el nombre consultado al motor de búsqueda. Por este motivo, los sistemas de desambiguación que agrupan resultados de búsqueda mediante *links* (ej. Jiang et al. [2009]; Yoshida et al. [2010]; Xu et al. [2015]) pueden agrupar en el mismo *cluster* todos estos perfiles. La Figura 6.3 muestra como ejemplo un fragmento de un listado de usuarios de *LinkedIn* correspondiente a la página web *079* del nombre de persona *John Smith* contenida en la colección MC4WePS.

No obstante, esta situación puede suceder en los propios perfiles de usuarios. La Figura 6.4 muestra el perfil de *LinkedIn* de un individuo llamado *John Smith* en donde se enlaza a los perfiles de otros usuarios con el mismo nombre, correspondiente a la página web *081* de este nombre de persona en la colección MC4WePS.

Además, esta situación también sucede en las páginas web correspondientes a buscadores verticales de personas como *Pipl*, *Spokeo* o *Intelius*. Estos buscadores de personas normalmente devuelven un listado de perfiles de diferentes redes sociales correspondientes a individuos llamados como el nombre consultado, de manera que deben tenerse en cuenta a la hora de estudiar el papel de las redes sociales en el problema. La Figura 6.5 muestra como ejemplo los resultados devueltos por el buscador *Pipl* al consultar el nombre de persona *John Smith*. Como se puede ver en la figura, *Pipl* muestra al usuario los perfiles de diferentes redes sociales como *Facebook* o *LinkedIn*.

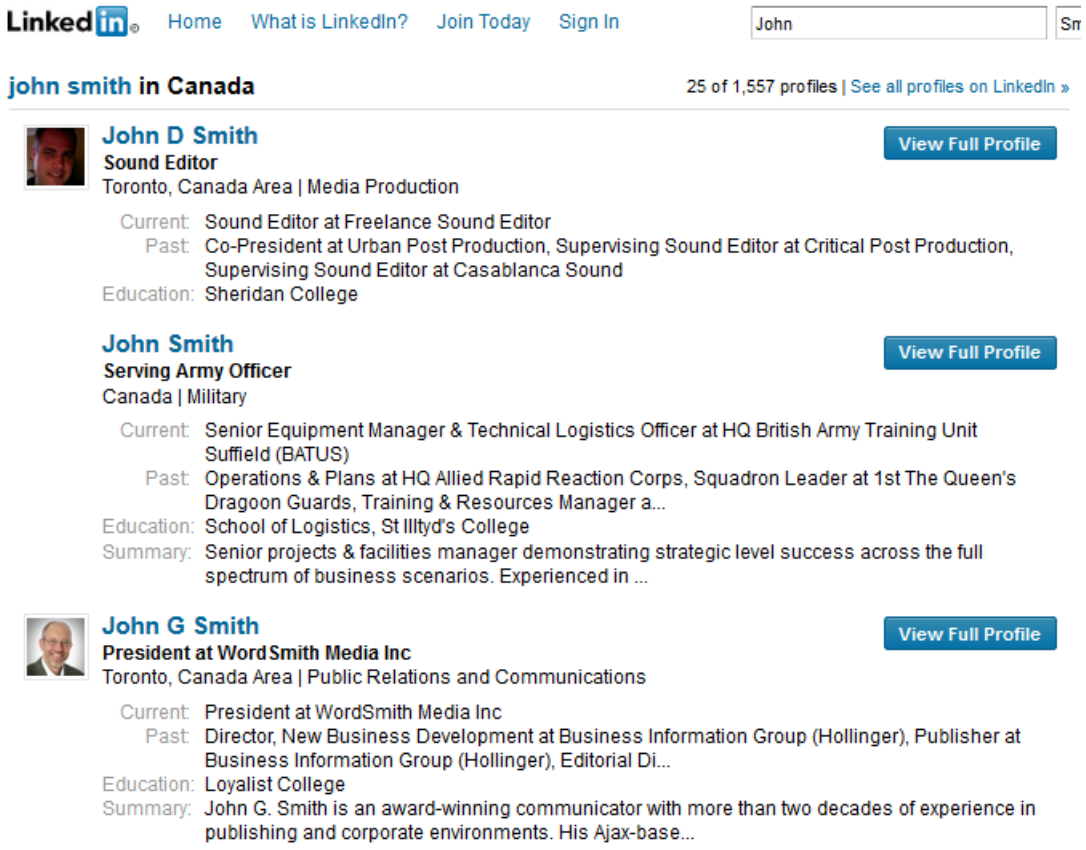

Figura 6.3: Fragmento de un listado de usuarios de *LinkedIn* cuyo nombre es *John Smith*.

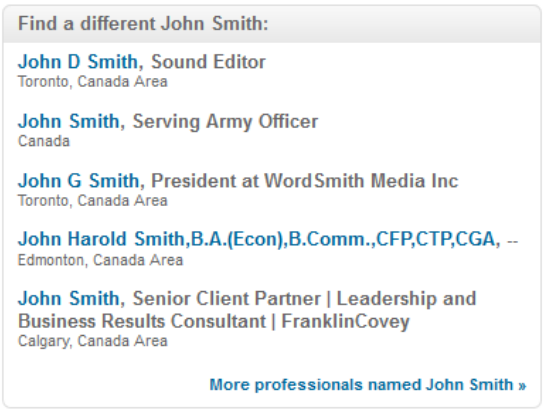

Figura 6.4: Enlace a perfiles de distintos usuarios desde el perfil de *LinkedIn* de un individuo llamado *John Smith*.

La sección 6.3 presenta tres nuevas heurísticas para tratar las redes sociales y los buscadores de personas en el problema de la desambiguación de nombres de persona, que resuelven las limitaciones de la política OIOS. Posteriormente, la sección 6.4 presenta los resultados obtenidos por ATC aplicando estas heurísticas sobre las colecciones ECIR2012 y MC4WePS y los compara con los dos métodos propuestos por Berendsen [2015]. Por último, la sección 6.5 enumera las principales conclusiones extraídas durante

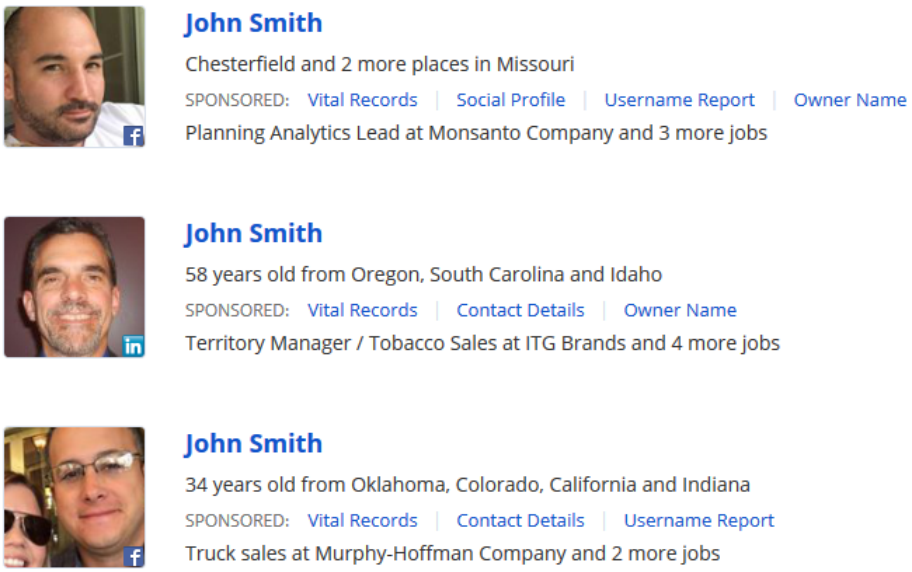

Figura 6.5: Resultados devueltos por *Pipl* al consultar el nombre de persona *John Smith*.

este capítulo.

6.3. Heurísticas de tratamiento de redes sociales y buscadores de personas

En este apartado se describen tres nuevas heurísticas que proponemos para el tratamiento de las redes sociales y los buscadores de personas en el problema de la desambiguación de nombres de personas en la Web. Todas estas propuestas requieren determinar si un resultado de búsqueda se corresponde a una página social o a un buscador vertical de personas o no y, en caso afirmativo, a cuál en concreto. Por ello, en primer lugar se describirá de qué manera se identifican las páginas sociales y los buscadores de personas con respecto al resto de páginas web. Posteriormente, se presentarán las heurísticas propuestas y las hipótesis en las que se fundamentan, que se justificarán mediante distintos experimentos.

6.3.1. Identificación de páginas sociales y buscadores de personas

La identificación de páginas sociales se basa en la comparación entre los dominios de los resultados de búsqueda con respecto a los dominios de las redes sociales. Dado un ranking de resultados de búsqueda $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$, podemos obtener la URL de cada resultado de búsqueda a partir de los *snippets* generados por el buscador. Como detallamos en la sección 3.3, esta información se proporciona en todas las colecciones de datos consideradas en esta tesis. A partir de las URLs, podemos extraer el

dominio del resultado de búsqueda mediante expresiones regulares. En lo sucesivo, denotaremos mediante $dom(W_i)$ al dominio del resultado de búsqueda $W_i \in \mathcal{W}$. Por otro lado, denotaremos por $\mathcal{RS}$ al conjunto de dominios de redes sociales. Este conjunto se ha recolectado a través de las redes sociales incluidas en la edición de Wikipedia en inglés[12]. Podemos determinar el conjunto de redes sociales presentes en un ranking de resultados $\mathcal{W}$ de la siguiente manera: $\mathcal{RS}(\mathcal{W}) = \{rs \in \mathcal{RS} \mid \exists W_i \in \mathcal{W} : dom(W_i) = rs\}$.

Se decide si un resultado de búsqueda $W_i \in \mathcal{W}$ es una página social mediante el siguiente predicado booleano:

$$esSocial(W_i) \equiv dom(W_i) \in \mathcal{RS} \tag{6.1}$$

Mediante este predicado booleano, podemos dividir las páginas web incluidas en el ranking de resultados $\mathcal{W}$ en páginas sociales, $\mathcal{W}_s = \{W_i \in \mathcal{W} \mid esSocial(W_i)\}$ y páginas no sociales $\mathcal{W}_{ns} = \{W_i \in \mathcal{W} \mid \neg esSocial(W_i)\}$.

La heurística OIOS empleada por Berendsen [2015] consiste en que las páginas sociales sean contenidas en *clusters* unitarios. Formalmente, esto significa que solamente se permiten comparaciones entre páginas no sociales:

$$OIOS(\mathcal{W}, W_i, W_j) \equiv W_i, W_j \in \mathcal{W}_{ns} \tag{6.2}$$

La identificación de la red social a la que pertenece una página social se realiza mediante una función $redSocial : \mathcal{W} \rightarrow \mathcal{RS}(\mathcal{W}) \cup \{NO\_SOCIAL\}$, donde $NO\_SOCIAL \notin \mathcal{RS}$ es un símbolo especial que identifica a las páginas no sociales, definida como sigue:

$$redSocial(W_i) = \begin{cases} dom(W_i) & si \quad W_i \in \mathcal{W}_s \\ \\ NO\_SOCIAL & si \quad W_i \in \mathcal{W}_{ns} \end{cases} \tag{6.3}$$

Por otro lado, denotaremos como $\mathcal{BP}$ al conjunto de dominios de buscadores verticales de personas. Al igual que con las redes sociales, este conjunto se ha recolectado a traves de los buscadores de este tipo incluidos en la edición de Wikipedia en inglés[13]. Análogamente, identificaremos un resultado de búsqueda $W_i \in \mathcal{W}$ como una página web correspondiente a este tipo de buscadores mediante el siguiente predicado booleano:

$$esBP(W_i) \equiv dom(W_i) \in \mathcal{BP} \tag{6.4}$$

[12] `https://en.wikipedia.org/wiki/Category:Social_networking_services`

[13] `https://en.wikipedia.org/wiki/Category:Online_person_databases`

6.3.2. Heurística (HRS1): *ONE IN ONE per social network*

Berendsen [2015] muestra que la política OIOS es un criterio efectivo a la hora de tratar a las redes sociales con respecto a otro tipo de métodos de agrupamiento, lo cuál significa que en muchas ocasiones estas páginas web se refieren a individuos diferentes. No obstante, esta política tiene la limitación de que no tiene en cuenta que un individuo pueda tener perfiles en varias redes sociales. La heurística (HRS1) asume la siguiente hipótesis:

(RS1): Las páginas sociales de la misma red social que aparecen en un mismo ranking de búsqueda suelen referirse a individuos diferentes.

La hipótesis (RS1) establece que las páginas sociales de la misma red social no se comparen entre sí. Por ejemplo, esta hipótesis no permite comparar dos páginas de *Facebook*, pero sí permite comparar una página de *Facebook* y otra página de *LinkedIn*. Esta suposición se basa en la observación de que es más probable que un individuo tenga perfiles en distintas redes sociales en vez de tener varios perfiles en la misma red social. Para corroborar esta afirmación, se ha calculado la proporción de páginas sociales de la misma red social que hablan de la misma persona en las colecciones ECIR 2012 y MC4WePS de la siguiente manera:

- Por un lado, se ha calculado el conjunto C de pares de páginas sociales de la misma red social para cada nombre de persona incluido en la colección.
- Por otro lado, se ha calculado el subconjunto $S \subseteq C$ de dichos pares de páginas web que hablan del mismo individuo, i.e. existe un *cluster* en el *gold standard* correspondiente que contiene a ambas páginas web.
- Finalmente, la proporción buscada se calcula como $100 \cdot \frac{|S|}{|C|}$ %.

En el caso de ECIR 2012, solamente un 2 % de las páginas de la misma red social hablan del mismo individuo, mientras que en el caso de MC4WePS el porcentaje es del 8,45 %. La colección MC4WePS se recopiló más recientemente que ECIR 2012, de modo que el hecho de que contenga más casos de individuos con varios perfiles en una misma red social puede indicar una tendencia del empleo de estas plataformas por parte de sus usuarios en los últimos años. Dado que ambos porcentajes son muy pequeños, se verifica que normalmente las páginas sociales de la misma red social no hablan del mismo individuo, tal y como asume la hipótesis (RS1).

La aplicación de esta heurística consiste simplemente en no comparar aquellos pares de resultados de búsqueda que pertenezcan a la misma red social. La detección de estos pares de páginas web puede llevarse a cabo mediante el siguiente predicado booleano:

$$mismaRedSocial(\mathcal{W}, W_i, W_j) \equiv \exists rs \in \mathcal{RS}(\mathcal{W}) : W_i, W_j \in \mathcal{W}(rs) \tag{6.5}$$

Por tanto, la heurística (HRS1) puede formalizarse de la siguiente manera:

$$HRS1(\mathcal{W}, W_i, W_j) \equiv \neg mismaRedSocial(\mathcal{W}, W_i, W_j) \tag{6.6}$$

Este criterio permite la comparación de páginas sociales de diferente red social, corrigiendo la principal limitación de la política OIOS, aunque no permite la comparación de páginas web de la misma red social. No obstante, dado que la fase 2 de ATC (algoritmo UPND) agrupa páginas web por transitividad, varias páginas web de la misma red social pueden agruparse en un mismo *cluster*: por ejemplo, si $W_{s1}, W_{s2} \in \mathcal{W}$ son páginas web de la misma red social, ambas serán agrupadas en el mismo *cluster* si existe otra página web $W_i \in \mathcal{W}$ de distinto dominio, tal que cumpla el criterio de agrupamiento en ambas, i.e. $sim(W_i, W_{s1}) > \gamma(W_i, W_{s1})$ y $sim(W_i, W_{s2}) > \gamma(W_i, W_{s2})$.

6.3.3. Heurística (HRS2): eliminación de rasgos comunes

Berendsen [2015] concluyó que la presencia de las páginas sociales puede tener un impacto negativo en los resultados de los principales sistemas de desambiguación de nombres de personas. La siguiente hipótesis trata de explicar la razón:

(RS2): Las páginas sociales de la misma red social comparten vocabulario que no aporta información útil para poder distinguir a diferentes individuos que comparten el mismo nombre y son causantes de agrupaciones incorrectas. En particular, estos rasgos consisten en palabras empleadas habitualmente por estas plataformas.

Para corroborar esta hipótesis, se ha calculado la similitud promedio entre distintos tipos de páginas web de cada nombre de persona de las colecciones ECIR 2012 y MC4WePS:

- $Avg(sim(\text{TODAS}))$: similitud promedio entre todos los pares de páginas web de un mismo nombre de persona de una colección.
- $Avg(sim(\text{NS-NS}))$: similitud promedio entre todos los pares de páginas no sociales de un mismo nombre de persona de una colección.
- $Avg(sim(\text{NS-S}))$: similitud promedio entre todos los pares formados por una página no social y otra página social de un mismo nombre de persona de una colección.

- $Avg(sim(\text{S-S}))$: similitud promedio entre todos los pares de páginas sociales de un mismo nombre de persona de una colección.
- $Avg(sim(\text{MISMA RED}))$: similitud promedio entre todos los pares de páginas sociales de la misma red social de un mismo nombre de persona de una colección.
- $Avg(sim(\text{DISTINTA RED}))$: similitud promedio entre todos los pares de páginas sociales de diferentes redes sociales de un mismo nombre de persona de una colección.

La Tabla 6.1 muestra las similitudes promedio anteriores para las colecciones ECIR 2012 y MC4WePS. Dado que la hipótesis (RS2) se refiere a palabras, estos cálculos se han realizado asumiendo una representación de los resultados de búsqueda mediante BoW (1-gramas) ponderadas mediante la función TF-IDF, y empleando el coseno para computar la similitud entre cada par de páginas web.

Similitud Promedio	**ECIR 2012**	**MC4WePS**
Avg(**sim**(**TODAS**))	0.05	0.045
Avg(**sim**(**NS-NS**))	0.041	0.048
Avg(**sim**(**NS-S**))	0.017	0.022
Avg(**sim**(**S-S**))	0.181	0.10
Avg(**sim**(**MISMA RED**))	0.61	0.287
Avg(**sim**(**DISTINTA RED**))	0.039	0.033

Tabla 6.1: Similitudes promedio entre distintos tipos de pares de resultados de búsqueda de un mismo nombre de persona en las colecciones ECIR 2012 y MC4WePS.

La tabla muestra que en ambas colecciones la similitud promedio entre páginas sociales es mayor que la similitud promedio entre páginas no sociales. En particular, las páginas sociales de la misma red social tienen la similitud promedio más elevada. Por tanto, cuando se comparan dos páginas web de la misma red social es más probable que sean agrupadas entre sí. No obstante, anteriormente verificamos que dos páginas web de la misma red social hablan del mismo individuo en una baja proporción de los casos, de manera que la mayoría de estas agrupaciones son incorrectas, como asume la hipótesis (RS2), puesto que se refieren a individuos distintos. Por otro lado, los altos valores de similitud son debidos a la coaparición de vocabulario común empleado por cada red social como asume (RS2). Por ejemplo, todos los perfiles de *LinkedIn* suelen contener el mismo vocabulario referido a la información laboral de sus usuarios. Además, la tabla muestra que la similitud promedio entre páginas no sociales es mayor a la de las páginas web sociales de distinta red social. Esto puede explicarse porque las redes sociales más conocidas tratan temáticas distintas. Por ejemplo, *LinkedIn* se centra

en la información laboral de los usuarios, mientras que en *Facebook* la gente publica información sobre sus intereses, opiniones o aficiones. Finalmente, MC4WePS presenta valores de similitud más bajos entre las páginas sociales debido a que muchas de ellas están escritas en diferentes idiomas y no se han empleado recursos de tratamiento del multilingüismo sobre los resultados de búsqueda.

En resumen, los resultados de la Tabla 6.1 muestran que los sistemas de desambiguación de nombres de persona que representan los resultados de búsqueda mediante BoW y emplean una medida de similitud para agrupar las páginas web pueden verse afectados por la presencia de las páginas sociales. En particular, algunos de los mejores sistemas de desambiguación de nombres de persona cumplen estas características [Balog et al., 2009; Long y Shi, 2010; Yoshida et al., 2010; Liu et al., 2011]. Estos resultados corroboran lo concluido por Berendsen [2015] al respecto. La aportación de este estudio consiste en concluir que esto se debe principalmente a las comparaciones entre páginas sociales pertenecientes a la misma red social, debido a que son mucho más similares entre sí con respecto al resto de comparaciones entre páginas web y en la mayoría de los casos se refieren a individuos distintos, como verificamos anteriormente.

La heurística (HRS2) asume la hipótesis (RS2) y se basa en identificar y eliminar el vocabulario común de cada red social presente en el ranking de resultados de búsqueda mediante el siguiente procedimiento de filtrado de términos:

- Dado un ranking de resultados de búsqueda $\mathcal{W}$ devuelto por un buscador tras consultar un nombre de persona, se obtiene el conjunto de páginas sociales del ranking $\mathcal{W}_s$ y se representa mediante BoW cada página social.
- Para cada red social $rs_j \in \mathcal{RS}(\mathcal{W})$, se obtiene el conjunto $\mathcal{W}(rs_j)$.
- Para cada grupo $\mathcal{W}(rs_j) \subseteq \mathcal{W}_s$ tal que $|\mathcal{W}(rs_j)| > 1$, realizar lo siguiente:
  - Calcular la *frecuencia de documento* de cada palabra que aparece en alguna página web del grupo, i.e.:

$$DF(f, \mathcal{W}(rs_j)) = |\{W_i \in \mathcal{W}(rs_j) | f \in Voc(W_i)\}| \tag{6.7}$$

  - Dado un valor $\alpha \in [0,1] \subset \mathbb{R}$, se decide que un rasgo f es vocabulario común de una red social $rs_j \in \mathcal{RS}$ si $DF(f, \mathcal{W}(rs_j)) \geq \alpha \cdot |\mathcal{W}(rs_j)|$.
  - Eliminar el vocabulario común de la red social rs_j en las páginas web del grupo $\mathcal{W}(rs_j)$. Los rasgos pueden componerse de una o varias palabras, de modo que se eliminan aquellos que contengan alguna de las palabras comunes de la red social identificadas en el paso anterior.

Este proceso de filtrado de términos se lleva a cabo de manera previa a la aplicación del algoritmo de *clustering*, de modo que puede verse como una fase adicional de preprocesamiento. El valor α determina la proporción de páginas web del grupo en la que aparece un rasgo identificado como común de la red social. Además, el valor de α también determina el tamaño del vocabulario filtrado. Si el valor de α es pequeño, entonces se filtrará un mayor número de palabras puesto que se requiere que aparezcan en un menor número de páginas web del grupo. En cambio, un valor grande de α significa que los rasgos identificados aparecen en una mayor proporción de páginas web de la red social, de manera que es más probable que formen parte del vocabulario habitualmente empleado por la misma. No obstante, si se toman valores muy altos de α, el conjunto de rasgos filtrados sería excesivamente pequeño, de modo que el comportamiento es similar a no aplicar ninguna heurística. Para evitar estas situaciones, se ha establecido el valor $\alpha = 0.75$, lo cuál significa que si un rasgo aparece en un 75 % de las páginas web del grupo entonces es eliminado.

La heurística (HRS2) tiene la ventaja de que no impone restricciones a la hora de comparar páginas sociales entre sí, de manera que permite la comparación de páginas web de distinta red social, resolviendo la principal limitación de la política OIOS. Además, (HRS2) permite la comparación entre páginas web de la misma red social a diferencia de (HRS1). No obstante, la heurística (HRS2) tiene la desventaja de que añade una fase adicional de preprocesamiento, de modo que es más costosa que la heurística (HRS1).

6.3.4. Heurística (BP): tratamiento de buscadores de personas

Los buscadores de personas son páginas web que ofrecen un servicio especializado en la búsqueda de individuos en Internet. De acuerdo con Artiles et al. [2010], a partir del año 2005 surgieron varias *start-ups* centradas en este tipo de motores de búsqueda, lo cual refleja la relevancia del problema tratado en esta tesis doctoral. Estos buscadores se caracterizan por emplear información extraída de bases de datos públicas (ej. censos) y la web profunda a la hora de encontrar información de diferentes individuos con el mismo nombre. No obstante, en los últimos años estos buscadores han enfocado la búsqueda de personas a traves de las redes sociales, de modo que devuelven un listado de perfiles de este tipo de páginas web correspondientes con distintos individuos llamados igual. Por este motivo, las páginas web de los buscadores de personas pueden ser consideradas como páginas sociales similares a aquellas consistentes en listados de usuarios con el mismo nombre. No obstante, ambos tipos de páginas web se diferencian entre sí porque los listados de usuarios de redes sociales solamente enlazan a perfiles de su propia de red social, mientras que los buscadores de personas presentan un conjunto de perfiles de usuarios de varias redes sociales (ver Figura 6.5).

La aparición de enlaces a perfiles de redes sociales de distintos individuos en una única página web puede tener como consecuencia que se agrupen los perfiles de dichos individuos cuando se emplean *links* para agrupar las páginas web. En el caso de las páginas sociales consistentes en listados de usuarios, esta situación puede prevenirse mediante el empleo de las heurísticas OIOS y (HRS1), dado que se tienen páginas enlazadas de la misma red social y ambas heurísticas evitan comparar estas páginas web entre sí. En cambio, los buscadores de personas no son páginas sociales y pueden enlazar a diferentes redes sociales, de modo que deben tratarse de manera diferenciada.

La heurística propuesta para el tratamiento de los buscadores de personas solamente se aplica cuando se emplean *links* para agrupar páginas web, puesto que se caracterizan por contener un conjunto de hipervínculos a diferentes perfiles de páginas sociales. En concreto, esto significa que esta heurística solo se aplica en la fase 1 del algoritmo ATC. La heurística consiste en evitar la comparación entre los siguientes tipos de páginas web:

- Por un lado, se evitan las comparaciones entre páginas web de buscadores de personas entre sí.
- Por otro lado, se evitan las comparaciones entre páginas web de buscadores de personas con cualquier página social.

Esto se justifica porque las páginas web de los buscadores de personas contienen *links* a perfiles sociales de diferentes individuos, de modo que su comparación con otras páginas web de buscadores de personas o páginas sociales puede dar lugar al agrupamiento de los resultados de búsqueda que mencionan a los individuos que aparecen en el listado de usuarios generado por el buscador de personas.

Formalmente, esta heurística puede expresarse de la siguiente manera:

$$BP(\mathcal{W}, W_i, W_j) \equiv \neg(esBP(W_i) \wedge esBP(W_j)) \wedge \neg(esBP(W_i) \wedge W_j \in \mathcal{W}_s) \\ \wedge \neg(W_i \in \mathcal{W}_s \wedge esBP(W_j)) \tag{6.8}$$

En el apartado 6.4.1 se estudia el impacto de esta heurística en la fase 1 del algoritmo ATC, dado que se trata de la fase en la que se agrupan las páginas web mediante *links*.

6.4. Resultados

En esta sección se presentan los resultados obtenidos por las heurísticas de tratamiento de redes sociales OIOS, (HRS1), (HRS2) y (BP). En primer lugar, se estudia

el impacto de la presencia de las redes sociales cuando se agrupan los resultados de búsqueda a partir de *links*. Posteriormente, se comparan los resultados obtenidos por el algoritmo ATC y el propuesto por Berendsen [2015] empleando las diferentes heurísticas. Estos tres métodos emplean rasgos textuales para comparar los resultados de búsqueda. En todos los experimentos llevados a cabo en esta sección se evalúan los resultados teniendo en cuenta todas las páginas web del ranking de resultados y solamente las páginas sociales. Estos últimos resultados nos indican si los experimentos agrupan adecuadamente este tipo de páginas web.

Para realizar este estudio se han empleado las colecciones ECIR 2012 [Berendsen et al., 2012] y MC4WePS [Montalvo et al., 2016], puesto que contienen un mayor porcentaje de páginas sociales con respecto a las colecciones WePS (ver apartado 3.3.6). La Tabla 6.2 muestra el número de nombres de personas y de documentos contenidos en las colecciones, los porcentajes de páginas sociales y buscadores verticales de personas y, finalmente, los porcentajes de nombres muy ambiguos y poco ambiguos. Se ha considerado que un nombre es muy ambiguo si en las páginas web se hace referencia a más de 10 individuos diferentes, siguiendo el criterio empleado por Montalvo et al. [2016]. En caso contrario, se ha considerado que el nombre es poco ambiguo.

Dato	ECIR 2012	MC4WePS
#Nombres	33	100
#Docs	3487	10432
%Social	34.73 %	8.36 %
%Buscadores	6.74 %	3.63 %
%MuyAmbiguos	81.82 %	51.00 %
%PocoAmbiguos	18.18 %	49.00 %

Tabla 6.2: Datos de las colecciones ECIR 2012 y MC4WePS.

Los nombres de persona contenidos en la colección ECIR2012 son de origen neerlandés y las páginas web están escritas en dicho idioma. Los resultados de búsqueda 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 sociales. Por esta razón, ECIR 2012 contiene un porcentaje muy elevado de este tipo de páginas web con respecto a MC4WePS. Por otro lado, esta colección también contiene un porcentaje mayor de páginas web correspondientes a buscadores de personas con respecto al de MC4WePS. Esto se debe a que ECIR 2012 fue recopilada mediante un buscador de personas, mientras que MC4WePS fue recopilada empleando un buscador convencional, concretamente Google. Finalmente, la colección ECIR 2012 está compuesta principalmente por nombres muy ambiguos,

lo cual puede explicarse porque los apellidos de la mayoría de ellos se encuentran entre los más comunes en los Países Bajos [Weerkamp et al., 2011].

En el caso de MC4WePS, se incluyen ranking de resultados reales devueltos por un único buscador. Esta colección se caracteriza por contener páginas web escritas en diferentes idiomas, a diferencia del resto de colecciones. Los anotadores identificaron páginas web escritas en 30 idiomas diferentes, prevaleciendo el inglés y el castellano (96.08 % entre ambos idiomas). No obstante, no se han preprocesado los resultados de búsqueda empleando recursos específicos para tratar el multilingüísmo para los experimentos llevados a cabo en esta sección. Por otra parte, en esta colección hay un mayor equilibrio entre el número de nombres muy ambiguos y poco ambiguos, a diferencia de lo que ocurre en la colección ECIR 2012.

6.4.1. Agrupamiento mediante *links*

En este apartado se estudia el impacto de la presencia de redes sociales cuando se agrupan resultados de búsqueda mediante *links* utilizando el Algoritmo 5.3 denominado LINKS (ver apartado 5.2.1). La agrupación de páginas web mediante hipervínculos normalmente genera agrupaciones correctas y es empleada por algunos de los mejores sistemas del estado del arte [Jiang et al., 2009; Yoshida et al., 2010; Xu et al., 2015] evaluados en las colecciones WePS. Dada la efectividad de este criterio de agrupación, es conveniente evaluar su rendimiento cuando hay presencia de páginas sociales.

La Tabla 6.3 muestra los resultados obtenidos por distintas heurísticas de tratamiento de redes sociales cuando se emplea la política de agrupamiento por *links* en las colecciones ECIR 2012 y MC4WePS, junto con el estudio de significancia estadística. Las columnas denominadas *TODOS* muestran los resultados obtenidos al considerar todos los resultados de búsqueda, mientras que las columnas llamadas *SOCIAL* muestran los resultados obtenidos al considerar exclusivamente las páginas sociales.

Los experimentos llevados a cabo pueden agruparse en los siguientes grupos:

- **LINKS:** el experimento *LINKS* consiste en aplicar el algoritmo LINKS empleando la política de enlace indirecto y sin aplicar ninguna heurística de tratamiento de redes sociales.
- **OIOS:** estos experimentos aplican la política de agrupamiento OIOS.
- **Heurística (HRS1):** estos experimentos aplican la heurística (HRS1).

No se ha incluido la heurística (HRS2) puesto que solamente tiene sentido emplearla cuando se utilizan rasgos de tipo textual para agrupar los resultados de búsqueda.

Por otro lado, la tabla muestra los resultados obtenidos al aplicar la heurística (BP) sobre las páginas de buscadores de personas junto con las heurísticas sobre páginas sociales mencionadas anteriormente. De esta manera, se puede analizar el impacto de estos resultados de búsqueda cuando se agrupan las páginas web mediante *links* y si la heurística (BP) propuesta para su tratamiento es adecuada.

Por último, la tabla muestra el estudio de significancia estadística entre los experimentos. Al lado de los resultados de medida-F se muestra una marca de la forma (k), donde $k \in \mathbb{N}$, de manera que un experimento marcado con (k) mejora significativamente los resultados de otro experimento marcado con (k') si $k < k'$, y ambos obtienen resultados similares si $k = k'$.

Colección	**ECIR2012**						**MC4WePS**					
Webs	**TODOS**			**SOCIAL**			**TODOS**			**SOCIAL**		
Experimento	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$
LINKS	0.60	0.74	0.64 (2)	0.24	0.91	0.32 (3)	0.86	0.46	0.54 (1)	0.76	0.80	0.72 (3)
LINKS+OIOS	0.92	0.67	0.76 (1)	1.00	0.75	0.84 (1)	0.90	0.43	0.54 (1)	1.00	0.67	0.77 (2)
LINKS+OIOS+(BP)	0.97	0.64	0.75 (1)	1.00	0.75	0.84 (1)	0.92	0.42	0.54 (1)	1.00	0.67	0.77 (2)
LINKS+(HRS1)	0.82	0.70	0.74 (1)	0.81	0.80	0.79 (2)	0.89	0.45	0.55 (1)	0.96	0.72	0.80 (1)
LINKS+(HRS1)+(BP)	0.95	0.65	0.75 (1)	0.97	0.77	0.85 (1)	0.91	0.44	0.55 (1)	0.98	0.72	0.80 (1)

Tabla 6.3: Resultados obtenidos por el algoritmo LINKS empleando la política de enlace indirecto con diferentes heurísticas de tratamiento de redes sociales.

Los resultados muestran lo siguiente:

- La aplicación de las heurísticas de tratamiento de redes sociales mejoran los resultados obtenidos con respecto a no aplicar ninguna heurística social (experimento *LINKS*) salvo para los resultados para todas las páginas web en MC4WePS. La mejora obtenida por las heurísticas de redes sociales se debe a un incremento notable de los valores de precisión con respecto al descenso de los valores de cobertura. La tabla muestra que el aumento de precisión se debe fundamentalmente al agrupamiento de las páginas sociales. En el caso de MC4WePS, la variación de los resultados es menor, debido a que contiene un menor porcentaje de páginas sociales con respecto a ECIR 2012, de manera que la aplicación de las heurísticas tiene un menor impacto. Esto también explica por qué el algoritmo LINKS sin aplicar ninguna heurística sobre redes sociales obtiene peores resultados en la colección ECIR 2012 que en MC4WePS.

  Las agrupaciones incorrectas cuando no se emplean heurísticas se producen por los siguientes motivos:

  - Las páginas sociales de una misma red social se agrupan entre sí porque enlazan a su página principal aunque se correspondan con individuos diferentes.

    - Se agrupan perfiles sociales de distintos individuos que aparecen como resultados de buscadores de personas.

- El impacto de los buscadores de personas es negativo en el agrupamiento de las páginas sociales. Esto se debe a que este tipo de páginas web enlazan a perfiles de distintas redes sociales correspondientes a diferentes individuos llamados igual. La política OIOS evita el impacto negativo de este tipo de páginas web porque no permiten agrupar las páginas sociales con otros resultados de búsqueda. En cambio, esto no sucede con la heurística (HRS1) dado que sí permite agrupar páginas sociales de distintas plataformas y con páginas no sociales. En particular, la comparación de los experimentos *LINKS+(HRS1)* y *LINKS+(HRS1)+(BP)* en la colección ECIR 2012 muestra que si no se aplica la heurística (BP) se produce un peor agrupamiento de las redes sociales. En cambio, si se aplica la heurística (BP) entonces se mejoran los resultados de precisión sin que exista un descenso drástico del valor de cobertura, lo que significa que se evitan agrupamientos incorrectos. Esto implica que tratar de manera especial los buscadores de personas es adecuado y la heurística propuesta (BP) es efectiva a la hora de evitar agrupamientos incorrectos causados por este tipo de páginas web. Finalmente, la heurística (BP) presenta un pequeño impacto en los resultados de MC4WePS debido a que contiene un menor porcentaje de páginas web correspondientes a buscadores de personas con respecto a ECIR 2012. En concreto, la aplicación de (BP) en MC4WePS da lugar a un ligero aumento de los valores de precisión acompañado de un pequeño descenso en los valores de cobertura. Esto indica que (BP) evita un pequeño número de agrupamientos incorrectos, pero al mismo tiempo no permite realizar unas pocas agrupaciones entre páginas web que mencionan al mismo individuo.

- La heurística OIOS evita las agrupaciones incorrectas de redes sociales puesto que no compara este tipo de páginas web entre sí. No obstante, no permite agrupar los perfiles sociales de un mismo individuo, a diferencia de la heurística (HRS1). Este comportamiento no supone diferencias significativas en los resultados en la colección ECIR 2012 dado que en esta colección la mayoría de las páginas sociales se corresponden con individuos diferentes, ya que se compone principalmente de nombres de persona muy ambiguos. No obstante, MC4WePS contiene un mayor número de páginas sociales que se corresponden con el mismo individuo, lo cual explica que OIOS agrupe las páginas sociales peor que la heurística propuesta (HRS1) en esta colección.

Por un lado, se puede concluir que el agrupamiento mediante *links* obtiene resultados bajos de precisión cuando hay presencia de páginas sociales. La aplicación de heurísticas de tratamiento de redes sociales permite evitar un gran número de agrupaciones

incorrectas de este tipo de páginas web. En particular, la heurística OIOS tiene la ventaja de que evita agrupamientos incorrectos de este tipo de páginas web pero no permite agrupar los perfiles sociales de un mismo individuo. Por su parte, el uso de la heurística (HRS1) tiene el inconveniente de que puede dar lugar a algunas agrupaciones de redes sociales incorrectas, pero permite agrupar los perfiles sociales de un mismo individuo. Este comportamiento es especialmente beneficioso en la colección MC4WePS, compuesta por ranking de resultados reales, en donde hay una mayor proporción de páginas sociales que se refieren al mismo individuo, y por ello obtiene mejoras significativas con respecto a la política OIOS.

Por otro lado, también se puede concluir que la presencia de páginas web de buscadores verticales de personas tiene un impacto negativo en el agrupamiento de páginas sociales. Esto se debe a que este tipo de páginas suelen consistir en listados de perfiles de distintas redes sociales de individuos diferentes, de modo que la agrupación mediante *links* puede agrupar en el mismo *cluster* las páginas de todos estos individuos. Las agrupaciones incorrectas debidas a los buscadores verticales pueden evitarse de dos maneras: (i) tener en cuenta este tipo de páginas web, de modo que no se comparen con páginas sociales; o (ii) aplicar la heurística OIOS que no permite comparar páginas sociales con cualquier otro resultado de búsqueda. En particular, se propone la primera opción empleando la heurística (BP) de tratamiento de buscadores verticales descrita en el apartado 6.3.4.

En cuanto a la elección de una heurística sobre otra, podemos concluir que la heurística propuesta (HRS1) es en general más adecuada que OIOS a la hora de agrupar páginas sociales, puesto que resuelve la principal limitación de esta última, consistente en no poder agrupar páginas sociales entre sí. En particular, (HRS1) obtiene un mejor agrupamiento de páginas sociales con respecto a OIOS en la colección MC4WePS, debido a que esta colección contiene páginas sociales diferentes que se refieren al mismo individuo. No obstante, ambas obtienen resultados similares de medida-F con todas las páginas web, puesto que aunque (HRS1) mejora los valores de cobertura por permitir agrupar páginas sociales de distinta plataforma entre sí, tiene la desventaja de que comete algunas agrupaciones incorrectas debidas a páginas no sociales que contienen enlaces a perfiles sociales de distintas personas llamadas igual, lo cuál implica un ligero descenso de precisión. Por tanto, en lo sucesivo asumiremos la aplicación de la heurística (HRS1) al agrupar páginas web mediante *links*, dado que obtiene un agrupamiento de redes sociales más adecuado y resuelve las limitaciones de la política OIOS.

En conclusión, en lo sucesivo se asumirá que en la fase 1 de ATC se emplearán las heurísticas (HRS1) y (BP) para tratar de manera especial las redes sociales y las páginas de buscadores de personas respectivamente.

6.4.2. Comparativa con el estado del arte

En este apartado se comparan los resultados obtenidos al aplicar diferentes heurísticas de tratamiento de redes sociales en los siguientes *baselines* y algoritmos:

- ***Baselines*:** ONE IN ONE y ALL IN ONE.
- **BEREN:** se trata del algoritmo propuesto por Berendsen [2015] y explicado con anterioridad. Se ha considerado la configuración detallada por el autor: por un lado, las páginas web se representan mediante el modelo VSM empleando BoW (1-gramas) ponderados con la función de pesado de términos TF-IDF. Por otro lado, las páginas no sociales se agrupan mediante HAC con política de enlace simple tomando el umbral $\gamma = 0{,}225$ obtenido por entrenamiento y empleando la similitud coseno. Finalmente, el algoritmo de mezcla de *clusters* sociales y no sociales utiliza el umbral $\tau = 0{,}5$.
- **ATC:** se emplea el algoritmo ATC con la configuración resumida en la Tabla 5.4. Además, en la fase 1 de agrupamiento mediante *links* se utiliza las heurística de redes sociales (HRS1) junto con la heurística de tratamiento de buscadores de personas (BP), tras lo concluido en el apartado anterior.

La Tabla 6.4 presenta los resultados de los algoritmos y los *baselines* ONE IN ONE y ALL IN ONE para las colecciones ECIR 2012 y MC4WePS sobre todos los resultados de búsqueda (columnas *TODOS*) y considerando solamente las páginas sociales (columnas *SOCIAL*). La tabla muestra los resultados de los algoritmos cuando no se tratan de manera especial las redes sociales y cuando se aplican distintas políticas de tratamiento de redes sociales, con el objetivo de comparar la eficiencia de estas heurísticas. En el caso del algoritmo BEREN se muestran los resultados de aplicar HAC sobre todas las páginas web (experimento *BEREN_HAC*) junto a varios experimentos que aplican las siguientes heurísticas: OIOS, algoritmo de mezcla sin penalización ($p = 0$), y algoritmo de mezcla con penalización ($p = 1$), denominados *BEREN+OIOS*, *BEREN+MIX ($p = 0$)* y *BEREN+MIX ($p = 1$)* respectivamente. Nótese que el experimento *BEREN+OIOS* se trata de la aplicación del primer método de mezcla de clusters de páginas sociales y no sociales propuesto por Berendsen [2015], dado que devuelve cada página social en un *cluster* unitario. En el caso del algoritmo ATC, se muestran los resultados obtenidos sin aplicar ninguna heurística social en la fase 3 (experimento *ATC*), y los obtenidos aplicando las heurísticas OIOS, (HRS1) y (HRS2), denominados *ATC+OIOS*, *ATC+(HRS1)* y *ATC+(HRS2)*, respectivamente. Además, la tabla muestra el estudio de significancia estadística de los resultados. Cada experimento tiene una marca (k) donde $k \in \mathbb{N}$ al lado de sus resultados de medida-F, de manera que se comparan los resultados situados

en una misma columna. Un experimento marcado con (k) obtiene mejoras significativas con respecto a otro marcado con (k') si $k < k'$, y ambos tienen resultados similares si $k = k'$.

Colección	ECIR2012						MC4WePS					
Webs	**TODOS**			**SOCIAL**			**TODOS**			**SOCIAL**		
Experimento	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$	*BP*	*BR*	$F_{0,5}$
ONE IN ONE	1.00	0.48	0.62 (5)	1.00	0.75	0.84 (2)	1.00	0.28	0.37 (6)	1.00	0.67	0.77 (2)
ALL IN ONE	0.17	1.00	0.25 (6)	0.13	1.00	0.20 (5)	0.52	1.00	0.59 (4)	0.43	0.99	0.54 (3)
BEREN_HAC	0.56	0.87	0.67 (4)	0.14	1.00	0.21 (5)	0.91	0.43	0.50 (5)	0.99	0.68	0.77 (2)
BEREN+OIOS	0.90	0.78	0.82 (1)	1.00	0.75	0.84 (2)	0.91	0.43	0.50 (5)	0.99	0.68	0.77 (2)
BEREN+MIX ($p = 0$)	0.74	0.82	0.76 (2)	0.55	0.85	0.62 (3)	0.91	0.43	0.50 (5)	0.99	0.68	0.77 (2)
BEREN+MIX ($p = 1$)	0.90	0.80	0.83 (1)	1.00	0.79	0.87 (1)	0.91	0.43	0.50 (5)	0.99	0.68	0.77 (2)
ATC	0.64	0.81	0.71 (3)	0.36	0.88	0.47 (4)	0.74	0.86	0.77 (2)	0.78	0.85	0.77 (2)
ATC+OIOS	0.95	0.74	0.82 (1)	1.00	0.75	0.84 (2)	0.82	0.79	0.79 (1)	1.00	0.67	0.77 (2)
ATC+(HRS1)	0.91	0.77	0.82 (1)	0.95	0.78	0.83 (2)	0.79	0.84	0.80 (1)	0.89	0.80	0.81 (1)
ATC+(HRS2)	0.89	0.78	0.81 (1)	0.97	0.78	0.85 (2)	0.78	0.84	0.79 (1)	0.94	0.73	0.78 (2)

Tabla 6.4: Resultados del algoritmo propuesto por Berendsen [2015] y ATC para las colecciones ECIR2012 y MC4WePS sobre todas las páginas web y únicamente sobre las páginas sociales.

Los resultados obtenidos por los *baselines* muestran que ONE IN ONE obtiene un agrupamiento de páginas sociales mejor que ALL IN ONE. Esto significa que la mayoría de las páginas sociales hablan de individuos diferentes. En particular, ALL IN ONE obtiene resultados muy bajos en la colección ECIR 2012 porque solamente un 1,45 % de los pares de páginas sociales de esta colección mencionan al mismo individuo, de modo que la aplicación de este *baseline* implica un descenso muy drástico en el valor de precisión. En cambio, MC4WePS contiene un mayor porcentaje de páginas sociales correspondientes al mismo individuo, de modo que ALL IN ONE no presenta un descenso tan drástico en la precisión en esta colección, a diferencia de lo que sucede con ECIR 2012.

Por otro lado, los resultados muestran que los métodos que realizan un menor tratamiento diferenciado sobre las páginas sociales (*BEREN_HAC* y *ATC*) obtienen peores resultados con respecto a aquellos que las tratan de forma diferenciada, lo cual corrobora lo concluido por Berendsen [2015]. En particular:

- *BEREN_HAC* consiste en aplicar el algoritmo HAC sobre todas las páginas web, en el cuál se basan la mayoría de los mejores sistemas del estado del arte [Chen y Martin, 2007a; Balog et al., 2009; Long y Shi, 2010; Liu et al., 2011], de modo que la presencia de las páginas sociales puede afectar negativamente sobre ellos. El experimento *BEREN+MIX ($p = 0$)* trata por separado las páginas sociales, pero posteriormente no penaliza los *clusters* no sociales en los que se han incluido páginas sociales en el proceso de mezcla. En cambio, el experimento *BEREN+MIX*

($p = 1$) aplica el mismo método pero penalizando dichos *clusters*, y obtiene mejoras significativas, lo cual indica que es conveniente aplicar políticas conservadoras a la hora de agrupar las páginas sociales.

- El experimento *ATC* aplica el algoritmo ATC sin emplear ninguna heurística social en las fases 2 y 3. Esto tiene como consecuencia un descenso drástico en los valores de precisión con respecto a los experimentos *ATC+OIOS*, *ATC+(HRS1)* y *ATC+(HRS2)*. Los resultados para las páginas sociales de este experimento muestran que dicho descenso de la precisión se debe a este tipo de páginas web. En cambio, los experimentos de ATC que aplican alguna heurística social obtienen mejoras significativas con respecto al experimento ATC.

En cuanto a los algoritmos, los resultados de ATC aplicando cualquiera de las heurísticas de páginas sociales consideradas son similares a los obtenidos por los dos métodos propuestos por Berendsen [2015], *BEREN+OIOS* y *BEREN+MIX($p = 1$)* en la colección ECIR 2012. No obstante, el algoritmo propuesto por Berendsen [2015] obtiene un resultado muy bajo de medida-F en la colección MC4WePS y, en particular, no supera al obtenido por el *baseline* ALL IN ONE. Esto se debe a que el rendimiento de HAC, utilizado por BEREN, es muy sensible con respecto al valor del umbral de similitud empleado y al grado de ambigüedad de los nombres de persona, tal y como se analizó en el capítulo 4. En concreto, el umbral $\gamma = 0{,}225$, empleado por Berendsen [2015], obtiene mejores resultados para nombres muy ambiguos con respecto a los nombres poco ambiguos. Además, todos los experimentos de tipo *BEREN* obtienen el mismo resultado en la colección MC4WePS porque el algoritmo de mezcla de *clusters* propuesto por Berendsen [2015] no tiene ningún efecto, dado que el umbral $\tau = 0{,}5$ es demasiado estricto y no permite unir ningún *cluster* social con otro *cluster* no social. En cambio, ATC mejora los resultados de los *baselines* y obtiene resultados similares en ambas colecciones. Esto significa que ATC se ve menos afectado por el grado de ambigüedad de los nombres de persona gracias al empleo de los umbrales adaptativos.

En cuanto a las heurísticas sobre páginas sociales propuestas (HRS1) y (HRS2), ambas mejoran los resultados con respecto a no emplearlas (experimento *ATC*), especialmente en la colección ECIR 2012 que contiene un mayor porcentaje de páginas sociales. El efecto de ambas heurísticas consiste en mejorar los valores de precisión sin alterar de manera drástica los valores de cobertura, lo cuál significa que son capaces de evitar muchos agrupamientos incorrectos entre páginas sociales. Además, las dos heurísticas resuelven la principal limitación de la heurística OIOS (experimento *ATC+OIOS*) que no permite agrupar páginas sociales entre sí. Tanto (HRS1) como (HRS2) obtienen resultados similares con respecto a la heurística OIOS en la colección ECIR 2012 debido a que un bajo porcentaje de páginas sociales de esta colección se refieren al mismo indi-

viduo. Por otro lado, (HRS1) obtiene un mejor agrupamiento de las páginas sociales en la colección MC4WePS con respecto a las heurísticas OIOS y (HRS2) por los siguientes motivos:

- Como se explicó anteriormente, (HRS1) mejora a OIOS en MC4WePS porque esta colección contiene una mayor proporción de páginas sociales que se refieren al mismo individuo, las cuales pueden agruparse mediante (HRS1) pero no pueden agruparse mediante OIOS.
- (HRS2) tiene un comportamiento similar a OIOS en la colección MC4WePS, de modo que (HRS1) también mejora sus resultados en esta colección. Esto se debe a que (HRS2) divide en grupos las páginas web de acuerdo a la plataforma a la que pertenecen, pero los grupos generados por (HRS2) en MC4WePS son muy pequeños porque esta colección contiene pocas páginas sociales de la misma red social. Esto tiene como consecuencia que se eliminan la mayoría de los rasgos de las páginas web sociales tras el proceso de filtrado explicado en el apartado 6.3.3, de modo que tienen similitud nula entre ellas y, por tanto, no pueden agruparse entre sí como sucede con la heurística OIOS.

Finalmente, se concluye que (HRS1) es una heurística más adecuada que (HRS2) porque obtiene resultados similares a nivel global, pero es menos costosa, dado que no requiere una fase adicional de preprocesamiento de rasgos. Por tanto, en lo sucesivo supondremos que ATC emplea la heurística (HRS1) en todas sus fases y, en particular, también aplica la heurística (BP) de tratamiento de buscadores de personas en la fase 1 que agrupa los resultados de búsqueda a partir de la compartición de *links*.

6.5. Conclusiones

Las principales conclusiones extraídas de este capítulo son las siguientes:

- Se ha corroborado que la presencia de páginas sociales impacta negativamente en el rendimiento de los resultados obtenidos por sistemas de desambiguación de nombres de personas que emplean *links* y rasgos textuales para representar las páginas web. Por un lado, esto se debe a que algunas de estas páginas web enlazan a perfiles sociales de individuos diferentes con el mismo nombre. Por otro lado, esto también se explica porque las páginas sociales de la misma red social son muy similares entre sí debido a que tienen mucho vocabulario en común.
- Se ha extendido el estudio de las redes sociales en el problema a las páginas de buscadores de personas, debido a que suelen consistir en un listado de enlaces a perfiles de redes sociales de difentes individuos con el mismo nombre.

- Se ha comprobado que la asunción de la heurística OIOS habitualmente es acertada, pero tiene la limitación de que no asume que varias páginas sociales se refieran a un mismo individuo.

- Se han propuesto dos nuevas heurísticas de tratamiento de páginas sociales. La primera heurística (HRS1) consiste en impedir las comparaciones entre páginas web de la misma red social, mientras que la segunda heurística (HRS2) identifica rasgos comunes de cada una de las redes sociales y, posteriormente, los elimina de las páginas web correspondientes. Ambas heurísticas evitan las principal limitación de la heurística OIOS.

- Se ha propuesto la heurística (BP) de tratamiento de los buscadores de personas consistente en impedir las comparaciones entre este tipo de páginas web y entre los buscadores de personas y las páginas sociales. De este modo, la heurística evita que se agrupen las páginas web que mencionan a los individuos diferentes que aparecen en los listados generados por los buscadores de personas.

- La heurística (HRS1) es adecuada a la hora de agrupar páginas web mediante *links* puesto que permite agrupar los perfiles sociales de un mismo individuo. Por otro lado, la heurística (BP) evita un peor agrupamiento de las páginas sociales aunque no tiene un alto impacto en los resultados globales.

- Las heurísticas propuestas (HRS1) y (HRS2) obtienen resultados similares y mejoran los obtenidos por la política OIOS en la colección MC4WePS, que contiene resultados de búsqueda reales. En cambio, ambas obtienen resultados similares a los de la política OIOS en ECIR2012, debido a que la naturaleza altamente ambigua de los nombres de personas contenidos en dicha colección tiene un impacto positivo en los resultados de OIOS. En particular, solamente un 1,45 % de pares de páginas sociales de ECIR 2012 se refieren al mismo individuo.

- Se ha concluido que el impacto negativo provocado por las páginas sociales puede atenuarse de manera eficiente con la heurística propuesta (HRS1). Esto se debe a que la heurística (HRS1) tiene un menor coste que la heurística (HRS2), dado que no requiere ninguna fase adicional de preprocesamiento. Por otro lado, la heurística (HRS1) es independiente con respecto a la representación de los resultados de búsqueda, mientras que (HRS2) supone que las páginas web se representan mediante rasgos textuales. Esto significa que (HRS1) puede aplicarse en cualquier caso y, en particular, para todas las fases del algoritmo ATC.

- Se ha comprobado de nuevo que el algoritmo HAC es sensible con respecto al umbral de similitud que emplea para generar los *clusters* resultantes y el grado de ambigüedad de los nombres de personas incluidos en las colecciones. El algoritmo

propuesto por Berendsen [2015] emplea HAC para agrupar las páginas no sociales pero obtiene resultados dispares en las colecciones ECIR 2012 y MC4WePS, diferenciadas por el grado de ambigüedad de los nombres de persona que contienen. En cambio, el algoritmo ATC obtiene resultados similares en ambas colecciones y no requiere aprender ningún parámetro mediante datos de entrenamiento.

7

Propuesta final de desambiguación de nombres de personas en la Web

"Cuanto más pienso en el lenguaje, más me asombra que la gente pueda siquiera entenderse."

— Kurt Gödel —

Este capítulo estudia el impacto del multilingüismo en el problema de la desambiguación de nombres de personas en la Web. Los sistemas del estado del arte no han tenido en cuenta este factor debido a que las colecciones disponibles se componen de documentos escritos en el mismo idioma, pese a que cada vez existe un mayor contenido en diferentes idiomas en la Red. Por ello, se ha utilizado la colección MC4WePS, recopilada recientemente, que se caracteriza por contener páginas web en diferentes idiomas. En primer lugar, en este capítulo se analiza la utilidad de un traductor automático para tratar el multilingüismo en el problema. Posteriormente, se presenta un método que generaliza las propuestas presentadas en los capítulos anteriores al escenario multilingüe y no requiere el uso de recursos de traducción. Además, el método propuesto obtiene mejoras significativas con respecto a los experimentos que emplean recursos de traducción. Por último, se presentan las principales conclusiones que resumen el contenido de este capítulo.

7.1. Introducción

Las colecciones de desambiguación de nombres de personas utilizadas por los sistemas del estado del arte asumen un escenario monolingüe en el que todas las páginas web devueltas por el motor de búsqueda están escritas en el mismo idioma. Por ejemplo, las colecciones WePS fueron recopiladas empleando la opción de búsqueda de Yahoo! que permite indicar que solamente se obtengan páginas web escritas en inglés. Por este motivo, los sistemas del estado del arte no han tenido en cuenta este factor en el problema. Solamente Mann y Yarowsky [2003] proponen un método que tiene en cuenta el multilingüismo basado en la extracción de atributos biográficos de los individuos en textos escritos en distintos idiomas mediante patrones aprendidos por entrenamiento. No obstante, esta metodología necesita un conjunto de entrenamiento diferente para cada tipo de atributo en cada lenguaje, lo cual supone un gran esfuerzo de recolección y

anotación de información. La necesidad de nuevas técnicas que traten el multilingüismo en el problema surge porque Internet alberga cada vez más contenido escrito en diferentes idiomas [Pimienta et al., 2009] y los motores de búsqueda son capaces de devolver páginas web escritas en lenguas diferentes para una misma consulta. No obstante, el tratamiento del multilingüismo en la Web es complicado y supone un reto para la comunidad científica [Montalvo et al., 2015a].

En este capítulo se presentan dos enfoques para tratar el multilingüismo:

- La primera propuesta se basa en el uso de una herramienta de traducción automática para tratar el multilingüismo. Esta estrategia ha sido empleada por varios autores en diferentes problemas de NLP donde el multilingüismo juega un papel esencial. Por ejemplo, Montalvo [2012] explora el uso de la traducción automática en el *clustering* multilingüe de noticias, mientras que Duque [2017] analiza los resultados obtenidos tras emplear una herramienta de este tipo en un problema de desambiguación del sentido de las palabras en un contexto multilingüe.
- La segunda propuesta consiste en una generalización del algoritmo ATC al escenario multilingüe sin necesidad de emplear recursos de traducción. La motivación de esta propuesta surge con el objetivo de evitar algunos inconvenientes del uso de herramientas de traducción automática. Por un lado, estos recursos requieren una etapa adicional de preprocesamiento dedicada a la traducción de los resultados de búsqueda, de modo que aumenta necesariamente el tiempo de proceso del sistema de desambiguación. En particular, esto es especialmente negativo en problemas que deben resolverse en tiempo real, como el tratado en esta tesis, donde los usuarios esperan una respuesta lo antes posible. Por otro lado, las herramientas de traducción automática actuales tienen un bajo rendimiento con textos que contienen errores ortográficos y gramaticales. No obstante, es frecuente encontrar este tipo de errores en diferentes tipos de páginas web en las que los usuarios emplean un lenguaje informal como, por ejemplo, los blogs personales, los foros, las redes sociales y, en general, en páginas web que permiten que los usuarios puedan introducir cualquier tipo de comentario.

La sección 7.2 presenta la primera aproximación basada en el uso de una herramienta de traducción automática. Posteriormente, la sección 7.3 estudia más profundamente el papel del multilingüismo en el problema a partir del análisis de ciertos nombres de personas caracterizados por presentar una mayor proporción de resultados de búsqueda escritos en distintos idiomas. A continuación, la sección 7.4 presenta la segunda propuesta de este capítulo, que trata el multilingüismo sin emplear ningún recurso de traducción. Finalmente, la sección 7.5 resume las principales conclusiones obtenidas a lo largo de este capítulo.

7.2. Propuesta utilizando traducción automática

Como primera aproximación al tratamiento del multilingüismo en el problema de desambiguación de nombres de personas en la Web se propone el uso de recursos de traducción automática. En esta sección, se detalla en primer lugar el proceso de traducción llevado a cabo y la etapa de preprocesamiento de los resultados de búsqueda. Posteriormente, se presentan los resultados obtenidos por el algoritmo ATC empleando traducción automática, comparándolos con respecto a emplear los rasgos originales.

7.2.1. Proceso de traducción

El proceso de traducción de los resultados de búsqueda se compone de los siguientes pasos:

- **Identificación del idioma:** se identifica el idioma en el que está escrito cada resultado de búsqueda mediante un detector de idiomas.
- **Selección del *idioma ancla*:** se establece un criterio para determinar el idioma al que se traducirán los resultados de búsqueda, denominado *idioma ancla*.
- **Traducción automática:** se emplea una herramienta de traducción automática para traducir al idioma ancla aquellos resultados de búsqueda escritos en otros idiomas diferentes.

Para la identificación de los idiomas de los resultados de búsqueda hemos utilizado un detector de idiomas de libre distribución[1]. Esta herramienta detecta el idioma en el que está escrito un texto mediante un clasificador *Naive Bayes* que toma en cuenta las secuencias de caracteres (o *c*-gramas) más habituales de cada idioma, de longitudes 2 y 3. En particular, este detector es capaz de identificar 54 idiomas diferentes, entre los que se encuentran todos los idiomas anotados en la colección MC4WePS exceptuando los siguientes:

- **Euskera:** tres páginas web en toda la colección, identificadas como castellano.
- **Gallego:** una página web en toda la colección, identificada como portugués.
- **Occitano:** una página web en toda la colección, identificada como catalán.

Esta herramienta tiene una tasa de aciertos del 96.17 % en la colección MC4WePS, de acuerdo con los idiomas anotados por los expertos. Los errores más frecuentes se

[1] `https://code.google.com/archive/p/language-detection/`

deben a que el detector confunde idiomas de la misma familia, por ejemplo: catalán con castellano, rumano con italiano o checo con eslovaco.

El criterio de selección del idioma ancla de un ranking de resultados de búsqueda asociado a un nombre de persona consiste en escoger el idioma más frecuente entre los resultados de búsqueda. En caso de que varios idiomas aparezcan con la misma frecuencia, se selecciona aquel en el que está escrito el resultado de búsqueda con menor posición en el ranking. Este criterio asegura que se traduce el menor número de resultados de búsqueda posible, lo cuál es conveniente en términos de eficiencia. Usando este criterio de selección se traducen un 17.21 % de resultados de búsqueda de la colección MC4WePS, siendo el inglés el idioma ancla para 69 nombres de persona, el castellano para 29 nombres de persona y el francés para los otros 2 nombres de persona restantes.

Finalmente, se ha seleccionado la herramienta de traducción automática de la compañía de Internet rusa *Yandex*[2]. Esta herramienta es capaz de traducir textos en 94 idiomas diferentes, entre los que se encuentran todos los idiomas identificados por el detector de idiomas utilizado. El traductor de Yandex[3] se basa en técnicas estadísticas y parte de un modelo de traducción y un modelo de cada uno de los idiomas obtenidos mediante la comparación de textos escritos en diferentes idiomas extraídos de Internet: por ejemplo, tomando las versiones en distintos idiomas de páginas web de empresas. Además, el traductor emplea varios diccionarios, de modo que cada palabra o multi-término tiene asociada una lista de traducciones a otros idiomas ordenada de acuerdo a su sentido más frecuente. La manera de traducir de esta herramienta consiste en dividir el texto en las oraciones en las que se compone y traducir separadamente cada una de ellas, de modo que se selecciona el sentido más probable de cada palabra teniendo en cuenta el contexto. Este recurso no traduce correctamente si se produce una identificación incorrecta del idioma del resultado de búsqueda por parte del detector de idiomas. Por ejemplo, suponiendo que se quiere traducir al inglés una página escrita en castellano pero identificada como catalán, la herramienta solamente podrá traducir aquellos fragmentos del texto donde hay vocabulario común entre el castellano y el catalán, mientras que no traducirá las partes del texto que contienen palabras empleadas exclusivamente por el castellano.

En lo sucesivo, nos referiremos como *rasgos originales* a los rasgos textuales extraídos a partir de los textos originales de los resultados de búsqueda, mientras que denominaremos como *rasgos traducidos* a los rasgos textuales extraídos tras la aplicación de este proceso de traducción.

[2] `https://www.yandex.com/`

[3] `https://yandex.com/company/technologies/translation/`

7.2.2. Preprocesamiento

La etapa de preprocesamiento llevada a cabo en los experimentos de este capítulo ha sido ligeramente modificada con respecto a la detallada en el apartado 3.1.1 con el fin de adaptarla al escenario multilingüe. A continuación, se detallan los pasos efectuados durante el preprocesamiento para cada nombre de persona en aquellos experimentos que emplean el proceso de traducción explicado anteriormente:

- **Extracción del texto plano:** MC4WePS se diferencia del resto de colecciones porque contiene resultados de búsqueda que no son necesariamente páginas web, por ejemplo, documentos en los formatos *pdf* o *doc*. La extracción del texto plano de los documentos se ha llevado a cabo mediante los diferentes *parsers* contenidos en la librería *TiKa Apache*, de acuerdo con la extensión del resultado de búsqueda.
- **Extracción de los hipervínculos:** se ha empleado el *parser* HTML de *TiKa Apache*, puesto que también permite extraer los *links* de las páginas web.
- **Identificación de idiomas:** se identifica el idioma en el que están escritos los resultados de búsqueda mediante el detector de idiomas descrito anteriormente, lo cual permite obtener el idioma ancla del nombre de persona.
- **Proceso de traducción:** se traducen al idioma ancla los textos de aquellos resultados de búsqueda escritos en otros idiomas empleando la herramienta de traducción automática de *Yandex*.
- **División del texto en frases:** se divide el texto traducido en las oraciones que lo componen a partir de los separadores ortográficos: puntos, saltos de línea, etc ...
- ***Stemming*:** se aplica un algoritmo de *stemming* a cada palabra sin incluir símbolos de puntuación, pero manteniendo los acentos.
- **Normalización del texto:** durante esta etapa se sustituyen las letras acentuadas por sus letras equivalentes sin acentuar. Por otro lado, se eliminan los separadores ortográficos como las comas, puntos y comas, guiones, etc.
- **Eliminación de palabras vacías:** se eliminan de los resultados de búsqueda las palabras vacías del idioma ancla, puesto que tras el proceso de traducción todos los resultados de búsqueda están escritos en dicho idioma. Los idiomas ancla identificados son inglés, castellano y francés, y se ha empleado una lista de palabras vacías habitual para cada uno de ellos. Además, se eliminan las apariciones del nombre y apellido del nombre de persona consultado, puesto que las páginas web han sido recuperadas por un motor de búsqueda bajo dicha consulta y se asume que aparecen en todas ellas.

Tras aplicar el preprocesamiento anterior, se efectúa la extracción de los rasgos de cada frase empleados por ATC: 3-gramas en mayúsculas y 1-gramas. Finalmente, se eliminan los rasgos que aparecen en un único resultado de búsqueda de un nombre de persona, puesto que no tienen poder discriminativo.

En el caso de los experimentos para los que no se emplea el proceso de traducción, el preprocesamiento llevado a cabo solamente se diferencia en que se eliminan las palabras vacías del idioma en el que está escrito cada resultado de búsqueda. Para ello, se han empleado listas de palabras vacías de diferentes idiomas extraídas de Internet[4]. El único idioma identificado por el detector para el que no se ha encontrado una lista de palabras vacías es el vietnamina. No obstante, solamente se ha identificado ese idioma en una única página web en toda la colección.

7.2.3. Resultados

La Tabla 7.1 muestra los resultados obtenidos por los *baselines* ONE IN ONE y ALL IN ONE y los algoritmos HAC y ATC cuando se representan los resultados de búsqueda mediante sus rasgos originales y sus rasgos traducidos. En particular, los experimentos llevados a cabo mediante los algoritmos HAC y ATC son los siguientes:

- **HAC:** se emplea la configuración detallada en la sección 3.2. El experimento HAC toma los rasgos extraídos de los textos originales mientras que HAC+TRAD toma los rasgos obtenidos tras aplicar el proceso de traducción. En ambos casos, el mejor umbral de similitud promedio para la colección MC4WePS es $\gamma = 0.13$.
- **ATC:** se emplea la configuración de ATC resumida en la Tabla 5.4 y las heurísticas de redes sociales explicadas en el Capítulo 6 (heurística (HRS1) en todas las fases y heurística (BP) en la fase 1). El experimento ATC toma los rasgos extraídos de los textos originales, mientras que el experimento ATC+TRAD toma los rasgos extraídos tras aplicar el proceso de traducción. Por último, el experimento ATC+CENT_TRAD traduce por separado los 1-gramas contenidos en los centroides DF-ICF empleados en la fase 3 de ATC, con el fin de explorar la idoneidad de traducir solamente estos rasgos, sin necesidad de traducir los documentos por completo. Nótese que los centroides DF-ICF se obtienen tras un proceso de filtrado de rasgos (ver apartado 5.3.1), de modo que no es necesario traducir cada uno de los 1-gramas de todos los textos.

La tabla también muestra el estudio de la significancia estadística. Cada experimento tiene asociada una marca de la forma (k) donde $k \in \mathbb{N}$, de manera que un experimento

[4] `http://www.ranks.nl/stopwords`

marcado con (k) obtiene mejoras significativas con respecto a otro marcado con (k') si $k < k'$, y ambos obtienen resultados similares si $k = k'$.

Experimento	BP	BR	$F_{0,5}$
ONE IN ONE	1.00	0.28	0.37 (4)
ALL IN ONE	0.52	1.00	0.59 (3)
HAC	0.68	0.82	0.70 (2)
HAC+TRAD	0.72	0.76	0.69 (2)
ATC	0.79	0.84	0.80 (1)
ATC+TRAD	0.81	0.79	0.79 (1)
ATC+CENT_TRAD	0.78	0.83	0.79 (1)

Tabla 7.1: Resultados obtenidos por los algoritmos HAC y ATC empleando rasgos originales y traducidos.

Los resultados pueden analizarse desde el punto de vista de los algoritmos empleados y la representación utilizada.

Resultados en función de los algoritmos

El *baseline* ALL IN ONE obtiene mejoras significativas con respecto a ONE IN ONE, lo cuál significa que, en promedio, la mayoría de los individuos de la colección tienen asociado más de un resultado de búsqueda. Al igual que en las colecciones WePS, HAC y ATC mejoran significativamente los resultados de los *baselines*. Además, ATC mejora significativamente los resultados de HAC, pese a que se emplea el valor de umbral que obtiene los mejores resultados para la colección. Esto se explica por dos motivos:

- ATC emplea una representación más rica de los resultados de búsqueda.
- El empleo del mismo umbral para todos los nombres de persona hace que HAC sea sensible con respecto al grado de ambigüedad. En particular HAC obtiene pobres resultados para los nombres de persona muy ambiguos debido a un descenso drástico en los valores de precisión. ATC evita esta situación gracias al empleo de los umbrales adaptativos.

Resultados en función de la traducción

La tabla muestra que las representaciones mediante rasgos originales y rasgos traducidos obtienen resultados similares considerando toda la colección MC4WePS en ambos algoritmos. Esto se explica por dos razones: (i) por un lado, se traducen una minoría de resultados de búsqueda debido al criterio de selección del idioma ancla explicado con anterioridad; y (ii) la colección MC4WePS contiene varios nombres de persona monolingües (ver Tabla 3.1) o con un bajo porcentaje de páginas web escritas en un idioma

distinto al idioma ancla. Por este motivo, existe un alto solapamiento entre las representaciones mediante rasgos originales y rasgos traducidos. El solapamiento entre ambas representaciones se ha calculado de la siguiente manera: sean $Voc^O(PN)$ y $Voc^T(PN)$ el vocabulario de rasgos originales y traducidos de un nombre de persona PN, respectivamente. Se calculan los siguientes valores:

$$\text{Solapamiento}^O(PN) = 100 \cdot \frac{|Voc^O(PN) \cap Voc^T(PN)|}{|Voc^O(PN)|} \,\% \tag{7.1}$$

$$\text{Solapamiento}^T(PN) = 100 \cdot \frac{|Voc^O(PN) \cap Voc^T(PN)|}{|Voc^T(PN)|} \,\% \tag{7.2}$$

El valor $\text{Solapamiento}^O(PN)$ es el porcentaje de rasgos originales que aparecen en la representación mediante rasgos traducidos, mientras que el valor $\text{Solapamiento}^T(PN)$ es el porcentaje de rasgos traducidos que aparecen en la representación mediante rasgos originales.

Posteriormente, se computa el solapamiento de ambas representaciones en la colección MC4WePS mediante la media aritmética de los valores anteriores de todos los nombres de persona, i.e.:

$$\text{Solapamiento}^O(\text{MC4WePS}) = \sum_{PN_i \in \text{MC4WePS}} \frac{\text{Solapamiento}^O(PN_i)}{|\,\text{MC4WePS}\,|} \,\% \tag{7.3}$$

$$\text{Solapamiento}^T(\text{MC4WePS}) = \sum_{PN_i \in \text{MC4WePS}} \frac{\text{Solapamiento}^T(PN_i)}{|\,\text{MC4WePS}\,|} \,\% \tag{7.4}$$

donde $|\,\text{MC4WePS}\,| = 100$ es el número de nombres de personas contenidos en la colección. Nótese que no tiene sentido comparar los vocabularios de distintos nombres de persona debido a que se extraen de búsquedas diferentes.

La Tabla 7.2 muestra las proporciones de solapamiento entre rasgos originales y traducidos en MC4WePS para los 1-gramas y los 3-gramas en mayúsculas (3-*gramas MAY*), debido a que son los rasgos empleados por el algoritmo ATC.

Features	$\text{Solapamiento}^{\mathbf{O}}(\text{MC4WePS})$	$\text{Solapamiento}^{\mathbf{T}}(\text{MC4WePS})$
1-gramas	89.18 %	95.98 %
3-gramas MAY	91.68 %	95.27 %

Tabla 7.2: Porcentaje promedio de solapamiento entre las representaciones de rasgos originales y traducidos en la colección MC4WePS para los tipos de rasgos empleados por HAC y ATC.

El porcentaje de rasgos traducidos que forman parte de la representación mediante rasgos originales es superior al 95 % tanto para 3-gramas en mayúsculas como para 1-gramas. Esto significa que el proceso de traducción solamente añade alrededor de un

5 % de rasgos nuevos con respecto a los originales. Por otro lado, el número de rasgos originales que aparecen en la representación obtenida tras el proceso de traducción es ligeramente superior al 89 %, lo que significa que solamente un 11 % de los rasgos originales han sido modificados tras el proceso de traducción. Estos datos indican que las representaciones de la mayoría de las páginas web se modifican ligeramente tras el proceso de traducción, lo cuál explica por qué obtienen resultados similares.

Por otro lado, HAC es más sensible en términos de los valores de precisión con respecto a ATC según se empleen rasgos originales o traducidos. Esto se explica porque HAC emplea la política de enlace simple. Esta política mezcla los *clusters* que contienen el par de resultados de búsqueda más similares entre sí. Por tanto, el par de *clusters* más similares puede variar más fácilmente aunque se modifique ligeramente la representación de los resultados de búsqueda. En cambio, ATC se ve menos afectado porque compara los *clusters* mediante sus centroides DF-ICF. Estos centroides eliminan términos frecuentes que aparecen en muchos *clusters* (ver apartado 5.3.1), lo cuál tiene un impacto positivo en términos de precisión.

Por último, el experimento ATC+CENT_TRAD obtiene resultados similares con respecto a ATC y ATC+TRAD. ATC+CENT_TRAD tiene el mismo comportamiento que ATC en las fases 1 y 2, y solamente traduce por separado los 1-gramas contenidos en los centroides DF-ICF empleados durante la última fase de mezcla de *clusters*. La similitud promedio entre los *clusters* es un 5 % superior en ATC+CENT_TRAD. Esto se puede deber a que el traductor devuelve siempre la misma traducción cuando se traduce una palabra, en particular, la correspondiente al sentido de la palabra más común de acuerdo con el diccionario. En cambio, cuando se traduce todo el texto, el traductor puede devolver distintas traducciones de una misma palabra según el contexto en el que aparezca. Por este motivo, los centroides obtenidos por ATC+CENT_TRAD comparten un mayor número de rasgos, lo cual podría explicar por qué obtiene una ligera mejora de los resultados de cobertura pero un ligero descenso en los valores de precisión.

Para tratar de analizar el impacto real de la traducción, en la siguiente sección se van a analizar más detalladamente los resultados obtenidos sobre un subconjunto de nombres de personas de la colección MC4WePS. Estos nombres se caracterizan porque presentan una mayor proporción de resultados de búsqueda escritos en un lenguaje distinto al idioma ancla.

7.3. Impacto del grado de multilingüismo de los nombres de persona

En esta sección se van a estudiar los resultados obtenidos por aquellos nombres de personas de MC4WePS cuyos resultados de búsqueda asociados presentan un alto

grado de multilingüismo. Este factor puede medirse a partir del número de resultados de búsqueda que se traducen durante el proceso de traducción, i.e. aquellos escritos en un lenguaje distinto al idioma ancla del nombre de persona. Sea $\mathcal{W} = \{W_1, W_2, \dots W_N\}$ un ranking de resultados de búsqueda, donde $l_{\mathcal{W}}$ es el idioma ancla y l_i denota el idioma del resultado de búsqueda $W_i \in \mathcal{W}$. Se calculará el grado de multilingüismo del conjunto de resultados de búsqueda $\mathcal{W}$ como el porcentaje de resultados de búsqueda escritos en un idioma distinto a l_W, i.e.:

$$GM(\mathcal{W}) = 100 \cdot \frac{|\{W_i \in \mathcal{W} | l_i \neq l_{\mathcal{W}}\}|}{|\mathcal{W}|} \,\% \tag{7.5}$$

En lo sucesivo, se denominará *nombres de persona altamente multilingües* a aquellos cuyo ranking de resultados de búsqueda asociado tiene un grado de multilingüismo de al menos un 25 %. Los nombres de persona de este tipo son especialmente adecuados para estudiar el impacto de la traducción, ya que se modifica la representación de un mayor porcentaje de sus páginas web con respecto al resto de nombres de persona.

Por otro lado, se analizarán aquellos *nombres de persona altamente multilingües* en los que la mayoría de los resultados de búsqueda están monopolizados por un individuo, habitualmente una celebridad. En particular, se considerará que estos nombres de persona son aquellos para los que al menos un 75 % de sus resultados de búsqueda hacen referencia a un mismo individuo. Nótese que estos nombres de persona no son necesariamente poco ambiguos. Por ejemplo, el *nombre de persona altamente multilingüe Mario Gómez* cumple esta propiedad, pero es un nombre muy ambiguo puesto que la mayoría de las páginas web mencionan a un futbolista profesional, pero muchas otras páginas web son perfiles de redes sociales de personas diferentes llamadas igual. Por otro lado, el *nombre de persona altamente multilingüe George Bush* es poco ambiguo pero no cumple esta propiedad, puesto que sus resultados de búsqueda asociados mayoritariamente se refieren a uno de los dos ex-presidentes de los Estados Unidos de América.

En primer lugar, se presentan los resultados obtenidos por ATC para los *nombres de persona altamente multilingües* tomando rasgos originales y traducidos. En segundo lugar, analizaremos el impacto de distintas características de los *nombres altamente multilingües* sobre los resultados, particularmente el grado de ambigüedad y la presencia de las páginas sociales.

7.3.1. Resultados de *nombres de persona altamente multilingües*

La Tabla 7.3 muestra los resultados obtenidos por los *nombres altamente multilingües* contenidos en la colección MC4WePS cuando se emplea el algoritmo ATC y se representan los resultados de búsqueda mediante rasgos originales y rasgos traducidos, i.e.:

los experimentos ATC, ATC+TRAD y ATC+CENT_TRAD explicados con anterioridad. El contenido de la tabla se explica a continuación:

- La columna *Nombre* muestra el nombre de persona y el código ISO de su idioma ancla según el detector de idiomas: EN (inglés), ES (castellano) o FR (francés). Los nombres de persona están ordenados de acuerdo al porcentaje de páginas web traducidas durante el proceso de traducción. Además, aquellos nombres de persona para los que un individuo monopoliza al menos un 75 % de los resultados de búsqueda se marcan con el símbolo *.
- La columna *Gr. Amb.* muestra dos datos:
  - Por un lado el grado de ambigüedad de los nombres de personas según el criterio de los creadores de la colección MC4WePS [Montalvo et al., 2016]. Los posibles valores son los siguientes:
    - *NA*: nombre no ambiguo, i.e. los resultados de búsqueda mencionan a un único individuo de acuerdo con los anotadores.
    - *PA*: nombre poco ambiguo, i.e. los resultados de búsqueda mencionan entre 2 y 10 individuos diferentes de acuerdo con los anotadores.
    - *MA*: nombre muy ambiguo, i.e. los resultados de búsqueda mencionan a más de 10 individuos diferentes de acuerdo con los anotadores.
  - Por otro lado, se indica entre corchetes el número de individuos diferentes mencionados en los resultados de búsqueda de cada nombre de persona.
- La columna *T* % indica el porcentaje de páginas web traducidas del nombre de persona durante el proceso de traducción, i.e.: aquellas escritas en un idioma distinto al idioma ancla según la identificación efectuada por el detector de idiomas.
- La columna *S* % indica el porcentaje de páginas sociales del nombre de persona.
- Las siguientes 9 columnas muestran los resultados de las tres métricas *B-Cubed* obtenidos por los experimentos ATC, ATC+TRAD y ATC+CENT_TRAD respectivamente.
- La última fila, *MEDIA*, muestra los resultados promedio de las métricas *B-Cubed* considerando todos los *nombres altamente multilingües*.

La tabla también muestra el resultado del estudio de la significancia estadística entre los experimentos para cada nombre de persona. En estos casos, se comparan distintos agrupamientos sobre un mismo nombre de persona en lugar de comparar resultados sobre un conjunto de nombres de persona como se ha realizado en los experimentos

llevados a cabo con anterioridad en esta tesis. Por este motivo, se han tomado distintos vectores a la hora de aplicar el test de Wilcoxon para cada nombre de persona. En concreto, se han tomado como muestra los vectores formados por los valores de medida-F de cada resultado de búsqueda con respecto al *gold standard*. La significancia estadística se muestra mediante la notación basada en marcas de la forma (k) donde $k \in \mathbb{N}$, explicada anteriormente, y que se sitúan al lado del valor de medida-F de cada experimento. En el caso de la última fila, se muestra el resultado del test de significancia estadística sobre todos los nombres de persona considerados, de modo que se ha calculado la significancia estadística de la manera habitual, es decir, tomando como muestra los vectores formados por los valores de medida-F de cada nombre de persona en cada experimento.

				ATC			ATC+TRAD			ATC+CENT_TRAD		
Nombre	**Gr. Amb.**	**T %**	**S %**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
Didier Dupont (FR)	MA [34]	55.96	22.94	0.53	0.84	0.65 (2)	0.73	0.71	0.72 (1)	0.51	0.86	0.65 (1)
Francisco Bernis (EN)	PA [4]	52.00	4.00	0.62	0.44	0.51 (2)	0.60	0.56	0.58 (1)	0.60	0.58	0.59 (1)
Amanda Navarro (EN)	MA [50]	50.00	6.86	0.80	0.82	0.81 (2)	0.80	0.85	0.83 (1)	0.79	0.83	0.81 (2)
Rafael Matesanz (EN) *	PA [6]	50.00	5.45	0.99	0.67	0.80 (1)	0.97	0.58	0.73 (2)	0.97	0.69	0.80 (1)
Rita Levi (ES) *	PA [2]	48.08	0.96	1.00	0.89	0.94 (2)	0.98	0.96	0.97 (1)	1.00	0.92	0.96 (1)
Albert Barillé (EN) *	NA [1]	46.46	2.02	1.00	0.58	0.74 (2)	1.00	0.63	0.78 (1)	1.00	0.63	0.78 (1)
Álex Rovira (EN)	MA [20]	45.26	21.05	0.83	0.57	0.68 (2)	0.80	0.59	0.68 (1)	0.83	0.57	0.68 (2)
Pierre Dumont (EN)	MA [39]	42.42	9.09	0.88	0.64	0.74 (1)	0.91	0.64	0.75 (1)	0.80	0.64	0.71 (2)
Mario Gómez (ES) *	MA [18]	42.00	3.00	0.96	0.60	0.74 (2)	0.96	0.71	0.81 (1)	0.94	0.62	0.74 (2)
Franco Modigliani (EN) *	PA [2]	37.61	1.83	1.00	0.87	0.93 (2)	1.00	0.95	0.97 (1)	1.00	0.86	0.92 (2)
David Robles (ES)	MA [58]	37.00	6.00	0.81	0.74	0.78 (1)	0.82	0.73	0.77 (1)	0.81	0.74	0.78 (1)
Álvaro Vargas (EN)	MA [50]	36.00	22.00	0.77	0.86	0.81 (1)	0.74	0.82	0.78 (2)	0.74	0.86	0.78 (2)
George Bush (EN)	PA [4]	31.48	1.85	0.62	0.75	0.67 (2)	0.60	0.80	0.69 (1)	0.61	0.76	0.67 (2)
Jesse García (EN)	MA [26]	31.19	6.42	0.77	0.56	0.65 (1)	0.77	0.57	0.65 (1)	0.77	0.56	0.65 (1)
Chris Andersen (EN) *	PA [6]	31.00	4.00	0.98	0.86	0.92 (2)	1.00	0.90	0.95 (1)	0.96	0.88	0.92 (2)
Katia Guerreiro (EN) *	PA [8]	30.91	9.09	0.97	0.59	0.73 (1)	1.00	0.65	0.78 (1)	0.94	0.62	0.73 (2)
Thomas Klett (EN)	MA [33]	30.61	7.14	0.92	0.71	0.80 (2)	0.96	0.73	0.83 (1)	0.87	0.75	0.81 (2)
Aldo Donelli (EN)	PA [4]	29.09	7.27	0.95	0.67	0.78 (2)	0.94	0.68	0.79 (1)	0.95	0.67	0.78 (2)
Miriam González (ES)	MA [43]	28.18	10.91	0.83	0.71	0.76 (1)	0.75	0.68	0.71 (3)	0.78	0.73	0.73 (2)
Antonio Camacho (EN)	MA [39]	27.52	22.94	0.73	0.75	0.74 (1)	0.71	0.75	0.73 (1)	0.70	0.75	0.71 (2)
Tim Duncan (EN) *	PA [3]	27.18	2.91	0.98	0.85	0.91 (2)	0.97	0.90	0.94 (1)	0.98	0.91	0.94 (1)
MEDIA				0.85	0.71	0.76 (2)	0.86	0.73	0.78 (1)	0.83	0.73	0.76 (2)

Tabla 7.3: Resultados obtenidos por ATC para los *nombres altamente multilingües* empleando las representaciones mediante rasgos originales y rasgos traducidos.

El test de significancia estadística indica que el proceso de traducción tiene un impacto positivo para los *nombres altamente multilingües* cuando se traducen los documentos completamente (experimento ATC+TRAD). Esta mejora consiste, en la mayoría de los casos, en un aumento de los valores de cobertura sin que se vea afectado el valor de precisión, aunque hay casos como el de *Didier Dupont* o *Thomas Klett* en los que la mejora se debe a un aumento de la precisión. Esto significa que gracias a la traducción automática se agrupa correctamente un mayor número de resultados de búsqueda. El empleo de rasgos originales mejora los resultados con respecto a los rasgos traducidos solamente

en tres nombres de persona: *Rafael Matesanz*, *Álvaro Vargas* y *Miriam Gonzalez*. Esto se debe a los siguientes motivos:

- Hay rasgos que se escriben de la misma manera en páginas web escritas en distinto idioma, pero son modificados tras el proceso de traducción, de modo que dejan de ser de utilidad para agrupar los resultados de búsqueda. En particular, estos rasgos consisten en NEs de personas y organizaciones que son especialmente útiles a la hora de distinguir entre distintos individuos. Por ejemplo, la mayoría de los resultados de búsqueda de *Rafael Matesanz* se refieren a un individuo vinculado con la *Universidad Autónoma de Madrid*. El nombre de esta universidad aparece escrito en castellano tanto en páginas web en inglés (idioma ancla) como en páginas web en castellano, de modo que este rasgo podía emplearse para agrupar páginas web escritas en ambos idiomas con la representación de rasgos originales. En cambio, al traducir las páginas web escritas en castellano, esta NE aparece como *Autonomous University of Madrid*, de modo que solamente se agrupan las páginas web de este individuo de acuerdo al idioma en el que están escritas. Además, el traductor es muy sensible con respecto a los acentos. Cuando la palabra *Autónoma* aparece acentuada se obtiene la traducción anterior, pero si esta palabra aparece sin acento (*Autonoma*), entonces el traductor devuelve *University Autonoma of Madrid*. Este tipo de errores tienen como consecuencia un impacto negativo en los valores de cobertura, como sucede con este nombre de persona.

- Como vimos en el Capítulo 5, ATC es sensible al vocabulario empleado habitualmente en Internet, y que normalmente suele corresponderse con palabras que aparecen en formularios de registro, anuncios, licencias o condiciones y términos de uso de las páginas web. Este tipo de rasgos introducen ruido puesto que no son de utilidad a la hora de distinguir distintos individuos. Su impacto es menos negativo cuando no se traducen los resultados de búsqueda dado que están escritos en distintos idiomas, de modo que así no facilitan el agrupamiento de resultados de búsqueda entre sí. En cambio, tras el proceso de traducción, estos rasgos se escriben de la misma manera en todas las páginas web donde aparecen y provocan un mayor número de agrupaciones incorrectas puesto que los centroides DF-ICF no logran filtrar todas estas palabras ruidosas. Esta situación tiene como consecuencia un impacto negativo en los valores de precisión, como sucede, en particular, con los nombres *Álvaro Vargas* y *Miriam González*.

Por su parte, los resultados del experimento ATC+TRAD_CENT no suponen una mejora significativa con respecto al experimento ATC que no aplica el proceso de traducción. Ambos experimentos se diferencias porque ATC+TRAD_CENT solamente traduce los rasgos de los centroides DF-ICF en la última fase de ATC. Los resultados de

ATC+TRAD_CENT están condicionados por los *clusters* iniciales obtenidos en las dos primeras fases del algoritmo, comunes en ambos experimentos. El efecto de traducir los rasgos de los centroides en la última fase por parte de ATC+TRAD_CENT consiste en una mejora de los resultados del valor de cobertura junto con un descenso proporcional del valor de precisión con respecto al experimento ATC que emplea solamente rasgos originales. En cuanto a la comparación del experimento ATC+TRAD_CENT con respecto a ATC+TRAD, ambos se diferencian porque ATC+TRAD emplea rasgos traducidos en todas las fases, mientras que ATC+TRAD_CENT usa los 3-gramas en mayúsculas originales. Por un lado, ATC+TRAD_CENT solamente tiene un comportamiento más adecuado que ATC+TRAD en nombres de personas como *Rafael Matesanz* por la misma razón que el experimento ATC: no se modifican algunos rasgos de este tipo que son útiles a la hora de agrupar correctamente los resultados de búsqueda (ej. *Universidad Autónoma de Madrid*). No obstante, ATC+TRAD_CENT tiene un comportamiento negativo en la última fase del algoritmo con respecto a ATC+TRAD, lo cual se refleja en un mayor descenso en los valores de precisión con respecto a ATC+TRAD, sin que mejore el valor de cobertura. Esto se debe a que ATC+TRAD_CENT traduce cada palabra de los centroides por separado, de forma que el traductor siempre devuelve la misma traducción. En cambio, como ATC+TRAD traduce todo el documento, el traductor no siempre devuelve la misma traducción para cada palabra, debido a que tiene en cuenta el contexto del texto y selecciona el significado o el sinónimo que considera más apropiado. Por este motivo, existe una mayor compartición de 1-gramas traducidos en ATC+TRAD_CENT con respecto a ATC+TRAD y, por tanto, realiza más agrupaciones que ATC+TRAD en la última fase del algoritmo. No obstante, esto significa que la traducción por palabras da lugar a un mayor número de agrupaciones incorrectas con respecto a traducir todo el texto. Dado que ATC+TRAD mejora significativamente los resultados de ATC+TRAD_CENT, en lo sucesivo nos referiremos exclusivamente a ATC+TRAD cuando se mencione el uso de la herramienta de traducción automática.

El uso de rasgos traducidos tiene un impacto especialmente positivo en aquellos nombres monopolizados por un único individuo, marcados con * en la tabla. Salvo en el caso de *Rafael Matesanz* comentado anteriormente, en todos ellos se obtienen mejoras significativas más pronunciadas con respecto a los nombres que no cumplen esta propiedad. Por ejemplo, este es el caso de *Albert Barillé* (cineasta, mejora del 4 %), *Franco Modigliani* (Premio Nobel, 4 % de mejora), *Katia Guerreiro* (cantante de fado, 5 % de mejora), *Mario Gómez* (futbolista, 7 % de mejora) o *Rita Levi* (Premio Nobel, 3 % de mejora). Estos nombres de persona tienen en común que su individuo más predominante tiene asociadas algunas NEs que le caracterizan y se traducen de manera adecuada, lo cual conlleva a que haya un mayor número de agrupaciones de resultados de búsqueda realizadas correctamente. Esto puede explicarse porque los resultados de búsqueda de este tipo de nombres de persona suelen estar escritos correctamente, lo cuál tiene un efec-

to positivo en el rendimiento del traductor automático. Por ejemplo, los resultados de búsqueda relacionados con estos nombres de persona suelen corresponderse con entradas enciclopédicas (ej. Wikipedia) o noticias en periódicos *online*. Además, las páginas web de este tipo ofrecen una amplia cantidad de información biográfica que puede ser bastante útil a la hora de identificar a los individuos. Por último, salvo la excepción de *Mario Gómez*, todos estos nombres de persona son poco ambiguos y normalmente contienen un menor porcentaje de páginas sociales con respecto a los otros nombres de persona.

Por otro lado, el proceso de traducción también tiene un impacto positivo en la mayoría de los nombres de personas que no son compartidos por un individuo popular. No obstante, estas mejoras generalmente son mucho más pequeñas, salvo en el caso de los nombres de persona *Didier Dupont* y *Francisco Bernis*, que son los nombres en los que se traduce un mayor porcentaje de páginas web. Los resultados de búsqueda de este tipo de nombres de persona suelen consistir en páginas sociales, blogs personales, o foros de Internet, escritos habitualmente en un lenguaje informal y con una mayor presencia de errores ortográficos y gramaticales. De acuerdo con Rozovskaya y Roth [2016], los traductores automáticos son bastante sensibles a estos errores y no suelen traducir las palabras afectadas, puesto que no forman parte de los diccionarios que emplean para traducir. Esto implica que hay un mayor solapamiento entre rasgos originales y traducidos, de modo que ambas representaciones son similares y, por tanto, los resultados obtenidos entre ambas son más cercanos entre sí. Por último, estos nombres de persona también se caracterizan por tener un grado más elevado de ambigüedad y contener un mayor porcentaje de páginas sociales.

El siguiente apartado estudia el impacto del grado de ambigüedad y la presencia de páginas web sociales con respecto a las mejoras obtenidas por los rasgos traducidos, con el fin de corroborar las afirmaciones realizadas anteriormente.

7.3.2. Correlación entre la representación y características de los nombres de persona

En este apartado se analiza el papel del grado de ambigüedad y las páginas sociales con respecto a los resultados obtenidos a partir de rasgos traducidos para los *nombres altamente multilingües*. Dado un *nombre altamente multilingüe* PN, denotaremos como $F^O_{0,5}(PN)$ y $F^T_{0,5}(PN)$ a los valores de medida-F obtenidos para el nombre PN empleando rasgos originales y traducidos, respectivamente. Calculamos la *ganancia* obtenida por los rasgos traducidos en el nombre PN como $g^T(PN) = F^T_{0,5}(PN) - F^O_{0,5}(PN)$. Nótese que $g^T(PN) < 0$ indica que los rasgos originales obtienen mejores resultados en términos de medida-F con respecto a los rasgos traducidos. Las Figuras 7.1 y 7.2 muestran cómo se ven afectados los valores $g^T(PN)$ con respecto al grado de ambigüedad de los nombres

de persona y el porcentaje de páginas sociales respectivamente. En ambos casos, se ha calculado el coeficiente correlación de Pearson de las variables consideradas, denotado como $\rho_{X,Y}$.

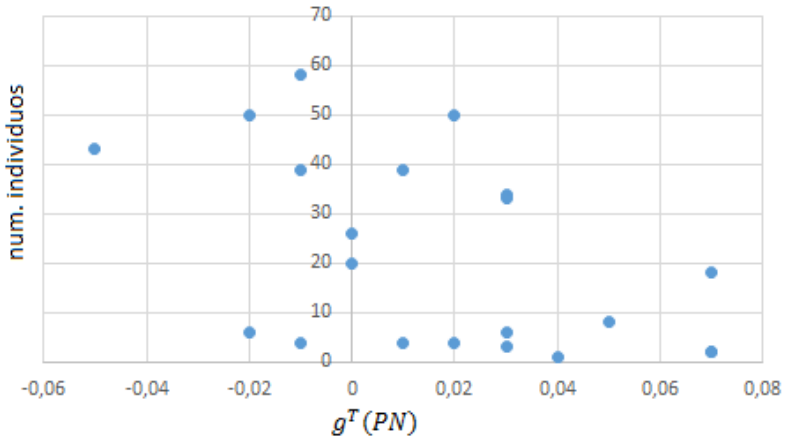

Figura 7.1: Ganancia obtenida por los rasgos traducidos con respecto a los originales de acuerdo al grado de ambigüedad.

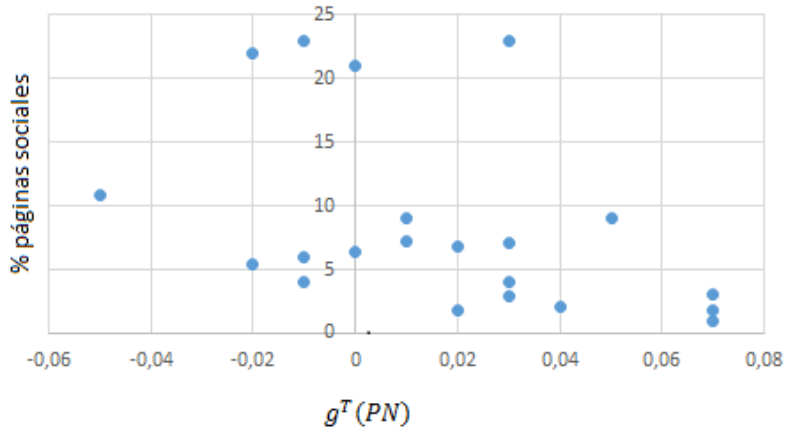

Figura 7.2: Ganancia obtenida por los rasgos traducidos con respecto a los originales de acuerdo al porcentaje de páginas sociales.

Podemos extraer las siguientes conclusiones de las figuras:

- En la Figura 7.1, el eje X representa los valores de $g^T(PN)$, mientras que el eje Y muestra el número de individuos distintos de los *nombres altamente multilingües*. El coeficiente de correlación entre estas dos variables es $\rho_{X,Y} = -0{,}499$, lo cuál indica que existe una tendencia entre débil y moderada de que los rasgos traducidos obtengan mejores resultados con respecto a los originales en nombres de persona poco ambiguos. Esto puede explicarse porque los nombres muy ambiguos no comparten muchos rasgos, tanto originales como traducidos. Por tanto, para los

nombres muy ambiguos hay menos agrupaciones de páginas web, lo cuál implica que los resultados obtenidos con rasgos originales y traducidos son más similares.

- En la Figura 7.2, el eje X representa los valores de $g^T(PN)$, mientras que el eje Y muestra el porcentaje de páginas sociales. Como en el caso anterior, la correlación entre estas variables, $\rho_{X,Y} = -0{,}439$, indica que hay una tendencia entre débil a moderada de que los rasgos traducidos obtengan mejores resultados con respecto a los originales cuando hay un menor porcentaje de páginas sociales. Esto se explica porque la heurística de tratamiento de redes sociales empleada por ATC agrupa este tipo de páginas web de la misma manera con independencia de la representación de los documentos. Por tanto, los resultados obtenidos usando rasgos originales y traducidos serán más similares entre sí a medida que haya un mayor número de páginas sociales entre los resultados de búsqueda.

También se han calculado los anteriores valores de correlación con respecto a toda la colección MC4WePS. En particular, estos valores son $\rho_{X,Y} = -0{,}122$ y $\rho_{X,Y} = -0{,}129$, entre los valores $g^T(PN)$ y el grado de ambigüedad de los nombres de persona y el porcentaje de páginas sociales, respectivamente. Estos valores de correlación son significativamente más bajos que los obtenidos para los *nombres altamente multilingües*. Esto indica que el grado de ambigüedad y el porcentaje de páginas sociales no afectan a los resultados globales considerando todos los nombres de persona, pero tienen un mayor impacto en un escenario de búsqueda con una mayor presencia de resultados multilingües.

En resumen, podemos concluir que el empleo de un traductor automático para manejar el multilingüismo en el problema ofrece mejoras significativas con respecto a representar los resultados de búsqueda mediante sus rasgos originales con los *nombres altamente multilingües*, en los que se traduce al menos un 25 % de sus páginas web asociadas. No obstante, el principal inconveniente del uso de traducción automática consiste en que aumenta necesariamente el tiempo de proceso del sistema de desambiguación debido a que se añade una etapa adicional de preprocesamiento dedicada al proceso de traducción.

7.4. Propuesta sin utilizar traducción automática

Esta sección presenta un nuevo algoritmo para la desambiguación multilingüe de nombres de persona. Este algoritmo consiste en una generalización del algoritmo ATC para el escenario multilingüe, de modo que tiene su mismo comportamiento en el caso de que todos los resultados de búsqueda estén escritos en el mismo idioma. Por otra parte, el nuevo algoritmo no requiere utilizar ningún recurso adicional de traducción,

de modo que evita el aumento de tiempo de proceso debido al empleo de este tipo de herramientas.

En primer lugar, se presenta el algoritmo propuesto. Posteriormente, se presentan los resultados obtenidos por el algoritmo con respecto a ATC empleando traducción automática, poniéndo especial énfasis en los *nombres de persona altamente multilingües*.

7.4.1. *Adaptive Threshold for Multilingual Clustering* (ATMC)

Se propone el algoritmo *Adaptive Threshold for Multilingual Clustering* (ATMC), que se basa en la siguiente hipótesis (ML) sobre el impacto del multilingüismo en la similitud entre documentos:

(ML): Es más probable que dos documentos escritos en el mismo idioma sean más similares entre sí con respecto a dos documentos escritos en idiomas diferentes, debido a que cada idioma emplea su propio vocabulario.

La manera de verificar la hipótesis (ML) ha consistido en calcular las similitudes promedio entre los resultados de búsqueda diferentes de un mismo nombre de persona escritos en el mismo idioma y en idiomas diferentes, empleando la colección MC4WePS. Para ello, en primer lugar se ha calculado la media aritmética de las similitudes de todos los pares de resultados de búsqueda diferentes de un mismo nombre de persona escritos en el mismo idioma y la media aritmética de las similitudes de todos los pares de resultados de búsqueda diferentes de un mismo nombre de persona escritos en distinto idioma.

La Tabla 7.4 muestra los resultados obtenidos sobre 3-gramas en mayúsculas (3-*gramas MAY*) y 1-gramas, debido a que son los rasgos empleados por ATC, en el que se basa ATMC. Además, la tabla muestra la diferencia porcentual entre ambas similitudes promedio. Para ello, se ha empleado la similitud coseno y las funciones de pesado empleadas por el algoritmo ATC para cada tipo de rasgo: función binaria para 3-gramas en mayúsculas y TF-IDF para 1-gramas.

Rasgo	Sim. MISMO idioma	Sim. DISTINTO idioma	Diferencia (%)
1-gramas	0.074	0.055	-25.67 %
3-gramas MAY	0.029	0.011	-62.07 %

Tabla 7.4: Similitudes promedio entre pares de resultados de búsqueda de MC4WePS escritos en el mismo y distinto idioma, para 3-gramas en mayúscula y 1-gramas.

Los resultados de la tabla muestran que la similitud promedio entre resultados de búsqueda escritos en diferentes idiomas es al menos un 25 % menor con respecto a

la similitud promedio entre resultados de búsqueda escritos en el mismo idioma para ambos tipos de rasgos, lo cuál valida la hipótesis (ML).

La idea en la que se basa ATMC consiste en comparar resultados de búsqueda escritos en distintos idiomas otorgando un papel importante a aquellos rasgos compartidos por ambos idiomas. Por un lado, este criterio evita comparar los resultados de búsqueda mediante rasgos que solamente se emplean en uno de los idiomas y son causantes de los bajos valores de similitud cuando se comparan documentos escritos en idiomas diferentes. Por otro lado, este criterio requiere identificar palabras que se escriban de la misma manera en distintos idiomas.

Sea $\mathcal{W} = \{W_1, W_2, \ldots, W_N\}$ un ranking de resultados de búsqueda obtenido tras consultar un nombre de persona. Denotaremos como F_i al conjunto de rasgos del resultado de búsqueda $W_i \in \mathcal{W}$ y l_i a su idioma identificado mediante el detector de idiomas. Por un lado, el vocabulario del ranking $\mathcal{W}$ se calcula de la siguiente manera:

$$F(\mathcal{W}) = \{f \in \bigcup_{i=1}^{N} | \exists W_i, W_j \in \mathcal{W} : i \neq j \wedge f \in F_i \cap F_j\} \tag{7.6}$$

puesto que se eliminan los rasgos que solamente aparecen en un único resultado de búsqueda.

Por otro lado, el conjunto de idiomas identificados en el ranking $\mathcal{W}$ se obtiene de la siguiente manera:

$$L(\mathcal{W}) = \bigcup_{i=1}^{N} l_i \tag{7.7}$$

Dado un rasgo $f \in F(\mathcal{W})$, podemos calcular el conjunto de idiomas de los resultados de búsqueda donde aparece de la siguiente manera:

$$L(f) = \{l_i \in L(\mathcal{W}) | f \in F_i\} \subseteq L(\mathcal{W}) \tag{7.8}$$

Si $|L(f)| = 1$, entonces f es un rasgo que solamente aparece en resultados de búsqueda escritos en un único idioma. En cambio, $|L(f)| > 1$ significa que f es un rasgo común en varios idiomas. Diremos que $f, f' \in F(\mathcal{W})$ son *rasgos comparables* si $L(f) \cap L(f') \neq \emptyset$.

El criterio de comparación de documentos consiste en comparar los resultados de búsqueda mediante sus rasgos comparables. Dados dos resultados de búsqueda $W_i, W_j \in \mathcal{W}$, sus rasgos comparables pueden calcularse de la siguiente manera:

$$F_{i,j} = \{f_i \in F_i | l_j \in L(f_i)\} \subseteq F_i \tag{7.9}$$

$$F_{j,i} = \{f_j \in F_j | l_i \in L(f_j)\} \subseteq F_j \tag{7.10}$$

Nótese que si $l_i = l_j$, entonces trivialmente se tiene que $F_{i,j} = F_i$ y $F_{j,i} = F_j$, lo cual garantiza que este criterio de comparación de documentos es una generalización del escenario monolingüe al multilingüe. Por otro lado, si $l_i \neq l_j$ implica que los resultados de búsqueda no se comparan teniendo en cuenta rasgos que sabemos de antemano que no son comunes gracias a la identificación de idiomas de cada rasgo explicada anteriormente. Por tanto, hay una mayor probabilidad de que los resultados de búsqueda puedan agruparse entre sí con respecto a compararlos teniendo en cuenta todos sus rasgos.

El criterio anterior puede generalizarse fácilmente para la comparación de *clusters* de resultados de búsqueda. El conjunto de idiomas de un *cluster* $C_k = \{W_{k_1}, W_{k_2}, \ldots, W_{k_p}\}$ puede obtenerse como la unión de los idiomas de los resultados de búsqueda que contiene, i.e.:

$$L(C_k) = \bigcup_{i=1}^{p} \{l_{k_p}\} \subseteq L(\mathcal{W}) \tag{7.11}$$

Dados dos *clusters* C_k y C_m representados mediante sus centroides de tipo CT, CT_k y CT_m, sus conjuntos de rasgos comparables son los siguientes:

$$F_{k,m} = \{f_k \in CT_k | L(C_m) \cap L(f_k) \neq \emptyset\} \subseteq CT_k \tag{7.12}$$

$$F_{m,k} = \{f_m \in CT_m | L(C_k) \cap L(f_m) \neq \emptyset\} \subseteq CT_m \tag{7.13}$$

Análogamente, la comparación de *clusters* mediante rasgos comparables generaliza el escenario monolingüe al multilingüe de manera trivial.

El vocabulario capturado por los rasgos comparables puede ser especialmente útil durante el proceso de desambiguación. En particular, algunos de estos rasgos se corresponden con NEs que se escriben igual en distintos idiomas, como sucede en el caso de los nombres de persona (ej. *John*), u organizaciones (ej. *Bank of America*). Nótese que el empleo de la traducción automática puede modificar este tipo de rasgos, de modo que no sean útiles para comparar los resultados de búsqueda. Siguiendo con los ejemplos anteriores, Yandex traduce del inglés al castellano *John* como *Juan* y *Bank of America* como *Banco de América*. Por otro lado, los rasgos comparables no siempre se corresponden con NEs, pero son igualmente útiles durante el proceso de desambiguación. Por ejemplo, el título de los artículos científicos escritos en inglés suelen referenciarse de la misma manera en textos escritos en otros idiomas. No obstante, algunos términos ruidosos también son rasgos comparables porque se escriben igual en la mayoría de los idiomas. Este es el caso de parte del vocabulario empleado habitualmente en Internet (ej. *copyright*), o los números.

Los rasgos comparables $F_{i,j}$ y $F_{j,i}$ consisten en subconjuntos de los rasgos de los resultados de búsqueda o *clusters* obtenidos tras eliminar rasgos que sabemos de antemano que no comparten entre sí. Por tanto, se tiene que $sim(F_i, F_j) \leq sim(F_{i,j}, F_{j,i})$, de modo que se benefician las comparaciones de documentos escritos en distintos idiomas con respecto a las comparaciones de documentos escritos en el mismo idioma. Una manera de evitar este sesgo comparativo consiste en balancear ambas similitudes $sim(F_i, F_j)$ y $sim(F_{i,j}, F_{j,i})$ teniendo en cuenta la proporción de rasgos comparables con respecto a todos los rasgos. Dicha proporción puede calcularse de la siguiente manera:

$$\alpha_{i,j} = \frac{|F_{i,j}| + |F_{j,i}|}{|F_i| + |F_j|} \tag{7.14}$$

Si el valor $\alpha_{i,j} \in [0, 1]$ es cercano a 1, significa que la mayoría de los rasgos de los resultados de búsqueda son comparables, de modo que las similitudes $sim(F_i, F_j)$ y $sim(F_{i,j}, F_{j,i})$ tendrán valores más cercanos. En cambio, si $\alpha_{i,j}$ es cercano a 0, significa que solamente hay un pequeño número de rasgos comparables entre todos los rasgos de los resultados de búsqueda, de modo que la comparación mediante rasgos comparables empleando la similitud $sim(F_{i,j}, F_{j,i})$ tendrá un mayor impacto. Se propone emplear la siguiente similitud y umbral adaptativo balanceados:

$$sim_{ML}(W_i, W_j) = \alpha_{i,j} \cdot sim(F_i, F_j) + (1 - \alpha_{i,j}) \cdot sim(F_{i,j}, F_{j,i}) \tag{7.15}$$

$$\gamma_{ML}(W_i, W_j) = \alpha_{i,j} \cdot \gamma(F_i, F_j) + (1 - \alpha_{i,j}) \cdot \gamma(F_{i,j}, F_{j,i}) \tag{7.16}$$

Esta misma expresión puede emplearse para comparar *clusters* entre sí puesto que se representan con centroides, y estos últimos pueden verse como bolsas de rasgos al igual que los documentos.

Por un lado, sim_{ML} es una similitud porque se trata de una combinación lineal entre similitudes y, de forma análoga, γ_{ML} es un umbral adaptativo porque es una combinación lineal de umbrales adaptativos que trivialmente cumple las propiedades de la Definición 4.1. Nótese que si $\alpha_{i,j} = 1$, entonces todos los rasgos son comparables, como sucede en el escenario monolingüe, de modo que la similitud y el umbral adaptativo se corresponden al calculado en ATC. Por otro lado, si $\alpha_{i,j} = 0$ significa que los documentos no tienen rasgos comparables, lo cuál necesariamente implica que $sim(F_i, F_j) = sim(F_{i,j}, F_{j,i}) = 0$ y, por tanto, los documentos no pueden agruparse entre sí de acuerdo con el criterio de agrupamiento de los umbrales adaptativos.

7.4.2. Resultados

La Tabla 7.5 muestra los resultados obtenidos por ATC, empleando rasgos originales y traducidos, y dos versiones de ATMC (ATMC_1 y ATMC_2) en la colección MC4WePS. Por un lado, ATMC_1 compara los resultados de búsqueda escritos en distinto idioma únicamente teniendo en cuenta sus rasgos comparables, i.e. $sim_{ML}(W_i, W_j) = sim(F_{i,j}, F_{j,i})$ y $\gamma_{ML}(W_i, W_j) = \gamma(F_{i,j}, F_{j,i})$. Por su parte, ATMC_2 compara estos pares de resultados de búsqueda mediante la similitud y umbral adaptativo balanceado obtenidos mediante las fórmulas 7.15 y 7.16 respectivamente. Por tanto, todos los experimentos pueden verse como versiones de ATMC aplicando diferentes políticas de comparación de resultados de búsqueda escritos en distinto idioma. La tabla también muestra el estudio de significancia estadística. El experimento marcado con el símbolo • mejora significativamente los experimentos sin esta marca, mientras que los experimentos que no contienen ninguna marca tienen resultados similares entre sí.

Algoritmo	BP	BR	$F_{0,5}$
ATC	0.79	0.84	0.80
ATC+TRAD	0.81	0.79	0.79
ATMC_1	0.71	0.91	0.79
ATMC_2	0.77	0.88	0.83 •

Tabla 7.5: Resultados obtenidos por ATMC con distintas políticas de comparación de resultados de búsqueda escritos en distintos idiomas con la colección MC4WePS.

ATMC_1 es el experimento que más beneficia las comparaciones entre resultados de búsqueda escritos en distintos idiomas, de modo que realiza un mayor número de agrupaciones entre resultados de búsqueda, lo cual tiene un impacto positivo en los valores de cobertura. No obstante, obtiene un drástico descenso en los valores de precisión, lo cuál significa que beneficiar las comparaciones entre documentos escritos en distinto idioma provoca un mayor número de agrupaciones incorrectas. Por su parte, ATMC_2 obtiene mejoras significativas con respecto al resto de experimentos por los siguientes motivos:

- ATMC_2 presenta un descenso en los valores de precisión con respecto a ATC porque beneficia las comparaciones entre documentos escritos en distinto idioma. No obstante, el descenso de precisión no es tan alto como en ATMC_1 gracias al empleo de la similitud y el umbral adaptativo balanceados. Esto significa que ATMC_2 evita las agrupaciones incorrectas realizadas por ATMC_1.

- ATMC_2 presenta un aumento en los valores de cobertura con respecto a ATC.

En lo sucesivo nos referiremos a ATMC_2 como ATMC, debido a que se trata de la versión del algoritmo propuesto más adecuada y la que finalmente se propone en esta tesis doctoral.

En el apartado 7.3.1 se comprobó que el empleo del traductor automático tiene un impacto positivo en los resultados de los *nombres altamente multilingües*. Por tanto, resulta de interés comparar los resultados de ATC+TRAD con respecto ATMC para este tipo de nombres de persona, con el objetivo de verificar la eficacia de la propuesta y compararla con el empleo de un recurso de traducción para tratar el multilingüismo.

La Tabla 7.6 compara los resultados de ATC sobre rasgos originales y traducidos y ATMC para los *nombres altamente multilingües*. La tabla también muestra el estudio de la significancia estadística empleando la notación mediante marcas de la forma (k) con $k \in \mathbb{N}$ explicada anteriormente.

		ATC			ATC+TRAD			ATMC		
Nombre	**Gr. Amb.**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**	**BP**	**BR**	**$F_{0,5}$**
Didier Dupont (FR)	MA [34]	0.53	0.84	0.65 (2)	0.73	0.71	0.72 (1)	0.43	0.90	0.58 (3)
Francisco Bernis (EN)	PA [4]	0.62	0.44	0.51 (2)	0.60	0.56	0.58 (1)	0.62	0.44	0.51 (2)
Amanda Navarro (EN)	MA [50]	0.80	0.82	0.81 (2)	0.80	0.85	0.83 (1)	0.75	0.82	0.79 (3)
Rafael Matesanz (EN) *	PA [6]	0.99	0.67	0.80 (2)	0.97	0.58	0.73 (3)	0.99	0.70	0.83 (1)
Rita Levi (ES) *	PA [2]	1.00	0.89	0.94 (2)	0.98	0.96	0.97 (1)	1.00	0.94	0.97 (1)
Albert Barillé (EN) *	NA [1]	1.00	0.58	0.74 (3)	1.00	0.63	0.78 (2)	1.00	0.71	0.85 (1)
Álex Rovira (EN)	MA [20]	0.83	0.57	0.68 (2)	0.80	0.59	0.68 (1)	0.78	0.59	0.67 (2)
Pierre Dumont (EN)	MA [39]	0.88	0.64	0.74 (1)	0.91	0.64	0.75 (1)	0.82	0.67	0.75 (1)
Mario Gómez (ES) *	MA [18]	0.96	0.60	0.74 (3)	0.96	0.71	0.81 (2)	0.91	0.80	0.85 (1)
Franco Modigliani (EN) *	PA [2]	1.00	0.87	0.93 (3)	1.00	0.95	0.97 (1)	1.00	0.90	0.95 (2)
David Robles (ES)	MA [58]	0.81	0.74	0.78 (1)	0.82	0.73	0.77 (1)	0.77	0.78	0.77 (1)
Álvaro Vargas (EN)	MA [50]	0.77	0.86	0.81 (1)	0.74	0.82	0.78 (2)	0.77	0.86	0.81 (1)
George Bush (EN)	PA [4]	0.62	0.75	0.67 (2)	0.60	0.80	0.69 (1)	0.62	0.75	0.67 (2)
Jesse García (EN)	MA [26]	0.77	0.56	0.65 (2)	0.77	0.57	0.65 (2)	0.77	0.58	0.67 (1)
Chris Andersen (EN) *	PA [6]	0.98	0.86	0.92 (2)	1.00	0.90	0.95 (1)	0.98	0.88	0.93 (2)
Katia Guerreiro (EN) *	PA [8]	0.97	0.59	0.73 (3)	1.00	0.65	0.78 (2)	0.95	0.73	0.85 (1)
Thomas Klett (EN)	MA [33]	0.92	0.71	0.80 (3)	0.96	0.73	0.83 (1)	0.87	0.76	0.81 (2)
Aldo Donelli (EN)	PA [4]	0.95	0.67	0.78 (3)	0.94	0.68	0.79 (2)	0.79	0.91	0.84 (1)
Miriam González (ES)	MA [43]	0.83	0.71	0.76 (1)	0.75	0.68	0.71 (3)	0.73	0.72	0.72 (1)
Antonio Camacho (EN)	MA [39]	0.73	0.75	0.74 (1)	0.71	0.75	0.73 (1)	0.69	0.80	0.74 (1)
Tim Duncan (EN) *	PA [3]	0.98	0.85	0.91 (2)	0.97	0.90	0.94 (1)	0.98	0.90	0.94 (1)
MEDIA		0.85	0.71	0.76 (2)	0.86	0.73	0.78 (1)	0.82	0.76	0.78 (1)

Tabla 7.6: Resultados de ATC usando rasgos originales y traducidos y ATMC para los *nombres altamente multilingües* de la colección MC4WePS.

La tabla muestra que ATMC mejora significativamente los resultados globales de ATC. Ambos experimentos emplean los rasgos originales y solamente se diferencian en la forma en la que se comparan los resultados de búsqueda. Por tanto, esto significa

que la similitud y el umbral adaptativo de las fórmulas 7.15 y 7.16 son adecuados en un contexto multilingüe. Por otro lado, ATMC obtiene resultados globales similares con respecto a ATC empleando traducción automática, pero ATMC tiene la ventaja de que no emplea ningún recurso de traducción, de modo que es más eficiente. Como se explicó anteriormente, ATMC obtiene valores más bajos de precisión debido a que beneficia las comparaciones entre resultados de búsqueda escritos en distintos idiomas, pero al mismo tiempo aumenta los valores de cobertura. En particular, el descenso en la precisión es un 2 % más pronunciado en los *nombres altamente multingües* con respecto a los resultados globales, lo cual se explica porque en este tipo de nombres se realizan un mayor número de comparaciones entre páginas web escritas en distintos idiomas.

ATMC generaliza el algoritmo ATC al escenario multilingüe, pero no corrige los errores cometidos por este algoritmo debido al vocabulario ruidoso, de modo que este tipo de vocabulario también es el principal responsable de las agrupaciones erróneas cometidas por ATMC. En particular, algunas palabras ruidosas como *copyright* o *email* y los números son rasgos comparables que tienen un impacto especialmente negativo en nombres de persona como *Amanda Navarro* o *Didier Dupont*. En el caso de otros nombres de persona con características similares como *Álvaro Vargas* o *Miriam González*, esta situación no tiene un impacto tan importante en los resultados porque hay un mayor número de rasgos de este tipo escritos en cada uno de sus idiomas (ej. *password* y *contraseña*), de modo que tiene un impacto negativo a nivel local de cada idioma. Esto mismo sucede con ATC cuando emplea rasgos originales, y por ello, tanto ese experimento como ATMC obtienen resultados similares en estos nombres de persona.

Por otra parte, ATMC parece obtener mejoras en nombres de personas compartidos por un individuo popular, como es el caso de *Albert Barillé*, *Mario Gómez* o *Katia Guerreiro* o *Rafael Matesanz*. Esto sucede porque los individuos populares de estos nombres de persona tienen asociado algún rasgo que les identifica, y se escribe de la misma manera en resultados de búsqueda escritos en distintos idiomas, pero son modificados tras el proceso de traducción. Este es el caso del ejemplo de *Rafael Matesanz* y *Universidad Autónoma de Madrid*, o en el caso de la cantante de fado *Katia Guerreiro*, varios de sus rasgos comparables se corresponden a títulos y letras de las canciones que interpreta. No obstante, hay nombres de este tipo para los que ATMC no mejora los resultados obtenidos por el traductor, como sucede con *Chris Andersen* o *Franco Modigliani*. En el caso de estas personas, los rasgos más útiles que les identifican se escriben de manera distinta en cada idioma, de modo que el empleo de la traducción automática tiene un impacto más positivo sobre ellos con respecto a la comparación de rasgos comparables de ATMC.

Finalmente, la Tabla 7.7 resume la configuración del algoritmo ATMC en cada fase. Por simplicidad, la tabla no incluye la medida de similitud y el umbral adaptativos

utilizados en las fases 2 y 3: por un lado, se usa la similitud coseno y se obtiene mediante la fórmula 7.15 al aplicar la comparación mediante rasgos comparables y, por otro lado, el umbral adaptativo empleado se calcula mediante la ecuación 7.16, donde el umbral adaptativo original se define en la fórmula 4.7. El símbolo - indica que en la fase correspondiente no se requiere especificar el factor.

Algoritmo	Rasgos	Política de enlace	Heurística social	Función de pesado	Centroide
FASE 1	URL y *links*	Enlace Indirecto	(HRS1)+(BP)	-	-
FASE 2	3-gramas en mayúsculas	-	(HRS1)	*Bin*	-
FASE 3	1-gramas	-	(HRS1)	TF-IDF	DF-ICF

Tabla 7.7: Configuración de ATMC.

7.5. Conclusiones

El multilingüismo añade una dificultad adicional al problema de desambiguación de nombres de personas en la Web, puesto que se requiere comparar adecuadamente resultados de búsqueda escritos en idiomas diferentes. Se ha explorado el uso de un traductor automático para abordar esta dificultad, empleando el criterio de traducir las páginas web al idioma más frecuente en los resultados de búsqueda de un nombre de persona, denominado *idioma ancla*. No obstante, el proceso de traducción no ofrece ninguna ventaja adicional con aquellos nombres de persona que no son *altamente multilingües*, y además implica una etapa de preprocesamiento adicional que incrementa el tiempo del proceso de desambiguación. Esto se explica porque la selección del idioma ancla implica que solamente se traducen un 17.21 % de los resultados de búsqueda de toda la colección, de modo que el impacto global del traductor es bajo. Por este motivo, se ha analizado el impacto del multilingüismo en los *nombres altamente multilingües*, caracterizados porque contienen al menos un 25 % de resultados de búsqueda escritos en un idioma distinto al idioma ancla. Para estos nombres de persona, el impacto del proceso de traducción es positivo porque permite agrupar un mayor número de páginas web correctamente, de modo que se mejoran los resultados de cobertura con respecto a emplear los rasgos extraídos de los textos originales. Los resultados obtenidos empleando el traductor dependen de la calidad de las traducciones de los rasgos adecuados a la hora de identificar individuos. En particular, el uso de esta herramienta tiene un impacto negativo cuando modifican rasgos escritos igual en distintos idiomas, como nombres de personas u organizaciones. Por otro lado, este tipo de recursos son especialmente sensibles cuando hay errores ortográficos y gramaticales.

Por otra parte, se ha estudiado el impacto en los resultados obtenidos mediante traducción automática de dos variables de los *nombres altamente multilingües*: el grado de

ambigüedad y el porcentaje de páginas sociales. Se ha detectado una tendencia débil a moderada de obtener mejores resultados para nombres poco ambiguos que contienen un bajo porcentaje de páginas sociales. Esto se explica por los siguientes motivos: (i) los individuos de nombres muy ambiguos comparten menos vocabulario entre sí, de modo que pueden realizar menos agrupaciones entre resultados de búsqueda y, por tanto, los resultados obtenidos empleando el proceso de traducción serán similares con respecto a los resultados obtenidos empleando los rasgos originales; y (ii) la heurística de páginas sociales de ATC marca la agrupación de este tipo de páginas web con independencia de la representación utilizada, de modo que cuantas más páginas sociales, más similares serán las agrupaciones obtenidas empleando rasgos originales y traducidos.

Se ha observado que el uso de herramientas de traducción es especialmente positivo para aquellos nombres de persona compartidos por algún individuo popular. Por un lado, esto puede explicarse porque este tipo de nombres de persona son habitualmente poco ambiguos y contienen un menor porcentaje de páginas sociales, de modo que tienen más tendencia a obtener mejores resultados como explicamos anteriormente. Por otro lado, hemos argumentado que esto también puede deberse a la naturaleza de los resultados de búsqueda de este tipo de nombres de persona, donde es más común encontrar páginas web consistentes en entradas enciclopédicas o noticias. Este tipo de resultados de búsqueda suelen estar escritos correctamente, lo cual repercute positivamente en el rendimiento del traductor automático. En cambio, los nombres que no son compartidos por un individuo popular suelen ser más ambiguos y contener un mayor número de páginas sociales. Además, estas páginas web suelen estar escritas de manera informal y normalmente contienen faltas de ortografía que dificultan la calidad de las traducciones. Por estos motivos, el empleo de la traducción automática tiene un impacto ligeramente menor en los resultados de búsqueda.

Se ha presentado una propuesta para tratar el multilingüismo en la tarea, consistente en el algoritmo *Adaptive Threshold for Multilingual Clustering* (ATMC). Este método generaliza el algoritmo ATC al escenario multilingüe y no requiere utilizar ningún tipo de recurso de traducción, de modo que emplea un preprocesamiento más ligero en términos de coste computacional. Esto último es especialmente importante en escenarios de búsqueda en la Web, puesto que los usuarios esperan una respuesta inmediata a sus consultas. El método propuesto se basa en identificar *rasgos comparables*, caracterizados por escribirse igual en distintos idiomas. ATMC otorga un papel importante a este tipo de rasgos a la hora de comparar los resultados de búsqueda escritos en distintos idiomas. No obstante, se ha verificado que comparar los resultados de búsqueda empleando solamente este tipo de rasgos tiende a realizar un mayor número de agrupaciones incorrectas. Esto se debe a que dentro de estos rasgos se incluye vocabulario ruidoso como números o palabras empleadas habitualmente en Internet. Se ha propues-

to una manera de evitar este problema tomando también en consideración todos los rasgos. En particular, se equilibran las similitudes y umbrales adaptativos obtenidos teniendo en cuenta todos los rasgos y los rasgos comparables, de acuerdo a la proporción de estos últimos sobre los primeros (ver fórmula 7.14). Esta solución equilibrada es capaz de agrupar correctamente resultados de búsqueda escritos en distinto idioma, pero evita muchas agrupaciones incorrectas que sí se realizan cuando solamente se toman en cuenta los rasgos comparables. Por este motivo, ATMC obtiene mejoras significativas con respecto a ATC, aunque ambos usen la misma representación sin emplear recursos de traducción. Finalmente, ATMC obtiene resultados similares a ATC empleando traducción automática para los *nombres altamente multilingües*. Este dato corrobora que la propuesta presentada es adecuada para el escenario multilingüe.

A pesar de que ATMC presenta una solución eficiente para tratar el multilingüismo, cuenta con dos problemas. Por un lado, arrastra el problema de ATC debido a la introducción de vocabulario ruidoso que normalmente aparece en las páginas web. Este problema puede tratarse explorando diferentes técnicas de filtrado de términos más elaboradas. Por otro lado, ATMC solamente es útil a la hora de agrupar páginas escritas en distinto idioma cuando existe vocabulario común entre dichos idiomas. No obstante, algunos rasgos útiles a la hora de identificar a determinados individuos no tienen por qué estar escritos de la misma manera en distintos idiomas. Por ejemplo, las técnicas de alineación de multi-términos y NEs [Martínez et al., 1998] pueden ser útiles a la hora de identificar este tipo de rasgos. Por otro lado, Montalvo [2012] explora el uso de técnicas de identificación de cognados para tratar el *clustering* multilingüe de noticias. Ambos tipos de técnicas pueden enriquecer el conjunto de los rasgos comparables empleados por el algoritmo ATMC, de modo que pueda mejorarse la comparación entre resultados de búsqueda escritos en distintos idiomas.

8
Conclusions and future work

"Truth is much too complicated to allow anything but approximations."

— John von Neumann —

This chapter concludes this thesis. First, we briefly summarize the research conducted in this work. Next, we expose the main conclusions and contributions of the work. After right, we present an outlook on future directions of the research work in this thesis. Finally, we list the publications obtained during the course of this thesis.

8.1. Summary of the research included in this thesis

This thesis is focused on person name disambiguation in a scenario of searching on the Web. The problem can be described as follows: given a ranking of search results returned by a search engine when looking for a person name, the goal is to cluster the search results according to the individual they refer to. Thus, the challenge lies in estimating the number of different individuals that share the same query name, and grouping the web pages that talk about the same individual in the same group. Although Internet users usually look for people information on the Web, the most well-known search engines only provide some disambiguation tools when the person name is shared by a celebrity or historical figure. The interest of this problem by NLP, IR and TM communities is due to several reasons: (i) person names are an specially ambiguous kind of NEs; (ii) web pages are not restricted to a certain domain; and (iii) scenarios which involve Web search require efficient methods due to users expect quick responses.

Person name disambiguation on the Web has been addressed in the state-of-the-art as a clustering problem. The proposed systems are composed of two main phases: (1) web page representation, where the goal is to select suitable features from the web pages for this problem; and (2) applying a clustering algorithm to group web search results, so that each cluster contains all the web pages of a particular individual.

Regarding the representation of web pages, most systems of the state-of-the-art represent the search results by means of a combination of weighted features. The most

popular features are bag of words and named entities, but this representation is usually enriched with other kind of information as links, biographical data or features extracted from external resources as *WordNet*, *Wikipedia* or other web pages retrieved by additional queries related to the person name.

Regarding clustering algorithms, the most competitive systems have used the *Hierarchical Agglomerative Clustering* (HAC). However, HAC requires a similarity threshold in order to cut its returned dendrogram, which is usually obtained from training data, with its results being very sensitive according to the value of this parameter. Furthermore, this methodology needs both sufficient and representative training data in order to guaranty the results will be consistent for different data collections, which requires a huge human effort. In particular, some of these systems use a clustering strategy based on two phases: the first phase obtains cohesive clusters with a high precision value, while the second phase merge the initial clusters in order to improve the recall score.

This thesis studies the following open problems in person name disambiguation on the Web:

- **Robustness:** most disambiguation systems need training data to learn parameters which will be used during the clustering phase. However, Artiles [2009] concludes that this clustering strategy leads to obtain sensitive results with respect to the ambiguity degree of the person names.
- **Social media:** Berendsen [2015] concludes that the presence of web pages from social media platforms could have a negative impact in the results of the state-of-the-art methods, so this kind of web pages should be treated in a special way. Although it is common to find this kind of web pages when looking for a person name, there have not been different strategies widely studied to deal with them.
- **Multilingualism:** despite search engines are able to retrieve web pages written in several languages for the same query, the problem has not been addressed in a multilingual scenario. However, increasingly there are web pages written in different languages due to the popularization of the Internet in non-English speaking countries [Pimienta et al., 2009].

The main goal of this thesis is to define and analyze new strategies for person name disambiguation on the Web to solve the former issues paying attention to the computational cost due to the nature of this problem:

- First, this thesis presents two new clustering algorithms which do not need training data. Both algorithms are based on the use of *adaptive threshold functions*,

which compute a similarity threshold when comparing search results depending on their characteristics without the need of training data. The proposed algorithms outperform the results obtained by most of the systems of the state of the art and get similar results with respect the best ones.

- On the other hand, this thesis proposes two different heuristics methods to treat web pages from social media platforms. These heuristics assume that the web pages from the same social media platform could be merged because they contain links to profiles from different people with the same name and they also share common features which are not useful to distinguish different individuals. The proposed heuristics avoid many wrong groupings of social web pages with respect to not using them. In addition, they allow to merge social web pages from different platforms which leads to obtain a better grouping of this kind of search results.

- Finally, this thesis presents a method to deal with multilingualism. The proposed method does not require the use of translation resources which necessarily increases the cost of the disambiguation process. It is based on comparing search results written in different languages giving an special role to those features which are written the same way in both languages. This method obtains promising results using a data set composed by web pages written in several languages.

8.2. Conclusions

This section is estructured as follows. First, we review the conclusion of this thesis organized by chapters. Second, we present the main contributions of this work.

8.2.1. Conclusions detailed by chapter

The main conclusions of this thesis organized by chapter are:

- **Chapter 1** introduces the person name disambiguation problem on the Web scenario and its related issues that have not been addressed widely in the state-of-the-art. In addition, this chapter present our main hypothesis and the goal of this dissertation.

- **Chapter 2** reviews the state-of-the-art. Most systems have addressed person name disambiguation as a clustering problem divided in two main phases: (i) web page representation; and (ii) applying a clustering algorithm to group the web search results.

Regarding web page representation, we have divided the selected features in five classes according to their nature and how they are extracted: linguistic, non linguistic, web features, biographical information and external features. The state-of-the-art systems have widely used linguistic and non linguistic features, being NEs and words (or 1-grams) the most popular ones. Most systems have used the VSM model to represent the search results. Some of them use an hybrid representation combining VSM and graphs models. On the other hand, other systems use probabilistic models. The first ones have shown a better performance.

Regarding clustering algorithms, HAC has been used by most of the best systems and it has shown a better performance with respect to other algorithms as Quality Threshold, k-means, k-medoids or Fuzzy Ants. HAC requires a similarity threshold value in order to cut its returned dendrogram. The systems have obtained this threshold value by means of training data, using it later with all the person names of the test collections.

The role of social media in the problem has been addressed only by Berendsen [2015], who proposed a dual strategy based on grouping social web pages and non social web pages separately, merging after both kind of web pages.

On the other hand, the multilingualism has not been addressed in this problem due to all the data sets used to evaluate the systems contain search results written in the same language.

- **Chapter 3** describes the experimentation framework used in this thesis. First, we briefly explain the VSM model to represent web pages and we define several term weighting functions and similarity measures. Next, we describe the data sets considered in the experiments analyzing their main characteristics and differences. Finally, we explain the evaluation metrics and the statistical significance test used to compare the performance of the experiments carried out in this work.

- **Chapter 4** presents the first approach for person name disambiguation on the Web of this thesis. The proposed method is divided in two main steps: web page representation and application of a clustering algorithm.

  First, regarding web page representation, we present our hypotheses about what kind of features are suitable to capture relevant information about individuals. Particularly, we assume that n-grams composed by capitalized words are specially useful for that aim. We have presented several preliminary experiments to check the former assumption using two state-of-the-art clustering algorithms and several features, particularly n-grams, k-skip-n-grams and NEs.

  Next, we present a new clustering algorithm to group search results, called Unsupervised Person Name Disambiguation (UPND). This algorithm is based on

computing automatically a similarity threshold value when comparing different documents to avoid the need of training data. For this purpose, we have introduced in this thesis the concept of *adaptive threshold*, which can be seen as a mathematical function that computes a similarity threshold taking into account some characteristics of the compared documents.

Right after, we compare the results of our proposal with respect to the state-of-the art systems concluding that UPND gets competitive results. However, the main inconvenience of UPND is that the use of capitalized *n*-grams does not allow to represent a high number of search results, which cannot be compared with the rest of the documents and they are finally returned within singleton clusters. On the one hand, this explains why UPND obtains high precision values; and on the other hand, this shows that there is still room for improvement with UPND algorithm if we include also other kind of features.

- **Chapter 5** presents the second proposal for person name disambiguation on the Web. The proposed method is another clustering algorithm called *Adaptive Threshold Clustering* (*ATC*). This algorithm also uses adaptive thresholds in order to avoid the need of training data and uses several kind of features in order to solve the representation problem of UPND.

  ATC is divided in three phases, where the goal of the two first phases is to obtain cohesive initial clusters, while the third phase merges them to improve the recall values. This kind of clustering strategy has shown effective in this problem and it has been used by some of the best methods of the state-of-the-art.

  The first phase of ATC groups search results by means of their link structure. We have compared two different merging policies by links of search results: (i) checking if the search results are linked; and (ii) checking if the search results share some link. The first policy ensures high precision values but it does not have a positive impact in the F-measure result. On the other hand, the second policy leads to some mistakes due to links of popular web pages in the Internet, but obtains higher recall improvement without a drastic descent of the precision score and improves F-measure. The second phase of ATC applies UPND due to it obtains high precision values. Finally, the third phase of ATC applies a merging algorithm of the initial clusters. In this phase, the search results are represented by BoW which solves the main problem of UPND and the clusters are represented by means of centroids in order to capture the content of all their search results.

  ATC algorithm obtains better results than all the state-of-the-art methods which do not need training data and than most of the systems which need training data. In addition, ATC gets close results with respect to the best systems in the state-of-the-art in the WePS evaluation campaigns.

- **Chapter 6** studies the role of social media in the problem. This kind of web pages could lead to group incorrectly web pages when these are represented by means of their links and tokens. In particular, these kind of features have been used by the best systems of the state-of-the-art [Jiang et al., 2009; Yoshida et al., 2010; Liu et al., 2011].

  This chapter presents two new heuristics to treat social web pages in person name disambiguation. The first heuristic just avoid the comparison between social pages from the same social media platform based on the hypothesis that users usually have just one profile in each social network. The second heuristic is based on the idea that social media platforms use common vocabulary which leads to merge incorrectly profiles of different people. Then, this heuristic identifies common words of each social media platform and lately removes them. Both heuristics avoid incorrect groupings due to social web pages. Particularly, the first one is more effective and it does not increase the computational cost of the disambiguation process. In addition, this chapter extends this study to web pages from people search engines because they contain links to social media profiles of people with the same name. Thus, the presence of these web pages is negative when grouping the search results by means of links. This chapter presents an heuristic to deal with this kind of web pages that does not allow comparisons between them or with social search results in order to avoid incorrect groupings between social profiles of different individuals.

- **Chapter 7** addresses person name disambiguation on the Web in a multilingual scenario and presents the final proposal of person name disambiguation on the Web of this thesis. The proposed algorithm is called *Adaptive Threshold for Multilingual Clustering* (ATMC) and it can be described as an ATC generalization to the multilingual scenario which does not requiere any translation resource.

  First, we have analyzed the impact of using a machine translation tool to deal with multilingualism. We have seen that this resource obtains improvements for *highly multilingual person names*, characterized by a high proportion of their search results are written in different languages. However, the use of translation tools increases the processing time of the disambiguation process due to a new preprocessing step dedicated to translate the search results.

  On the other hand, ATMC deal with multilingualism taking into account that documents written in different languages are less similar than documents written in the same language. Thus, ATMC computes the similarity between two search results written in different languages as follows: (i) taking into account all their features; (ii) taking only into account those features written the same way in both languages; and (iii) balancing both similarities according to the proportion of the

second kind of features with respect to all the features. ATMC gets similar results with respect to ATC using a machine translation tool for *highly multilingual person names*, but it does not need any translation resource, so the processing time is not increased.

8.2.2. Summary of contributions

Next, we list the main contributions of this work:

- We have presented a typology of systems for person name disambiguation on the Web according to several aspects: the models used to represent the search results, the different kinds of features employed to capture the content of the search results and the clustering algorithms applied to group the search results. In addition, it has been discussed which models, features and clustering algorithms have shown a better performance for this problem.
- We have analyzed the suitability of several kinds of features in order to decide when different documents talk about the same individual. In particular, we have proposed a document representation based on capitalized n-grams.
- We have presented the adaptive threshold functions to compute automatically a similarity threshold when comparing different documents.
- We have presented a new clustering algorithm called Unsupervised Person Name Disambiguation (UPND). The proposed algorithm has several desirable properties: (i) it is a deterministic algorithm; (ii) it is able to estimate the number of clusters; and (iii) it does not require any parameter learned by training data. We have shown that UPND gets competitive results in several data sets of person name disambiguation on the Web. Recently, Toba et al. [2017] have used UPND in order to find information about former students of a university to update the information of an alumni database.
- We have analyzed the suitability of two different clustering policies of web pages by means of their link structure. We have seen that checking if two web pages are linked ensure high precision values but it does not suppose any advantages in terms of the F-measure results. On the other hand, taking the policy of checking if two web pages share any link leads to improve the results. However, this policy merges incorrectly web pages that contains links to popular web pages as search engines or online newspapers.
- We have presented the algorithm Adaptive Threshold Clustering (ATC) for person name disambiguation. ATC preserves the properties of UPND and it also solves the

main inconvenience of UPND because it enriches the web pages representation by means of their URLs, links and BoWs. The proposed algorithm gets better results than the unsupervised baselines and all the systems which do not need training data. In addition, ATC gets similar results with respect to the best systems of the state-of-the-art.

- We have corroborated that the presence of web pages from social media platforms (*social pages*) could have a negative impact in the results of person name disambiguation systems. We have shown that this is due to social pages from the same social platform contain links to several profiles of different people with the same name and they share words that are not useful in order to distinguish between several individuals. Most of the best systems of the state-of-the-art represent the search results with these features, so they could obtain lower results due to the presence of this kind of web pages. In addition, we have extended this study to vertical search engines specialized in people search on the Web due to they are usually focused in social media profiles.

- We have proposed three new heuristics to treat social pages in the problem. The proposed heuristics avoid incorrect groupings due to this kind of web pages and they allow to compare web pages from different social networks unlike the ONE IN ONE Social policy proposed in the state-of-the-art, so both take into account that each individual can have several profiles in different platforms. In addition, one heuristic deal with web pages from people search engines which contain links to social media profiles of different individuals with the same name.

- We have analyzed the impact of using a machine translation tool to address multilingualism in person name disambiguation on the Web. In addition, we have identified the characteristics of the person names such that the use of this resource is suitable.

- We have presented the algorithm Adaptive Threshold for Multilingual Clustering (ATMC) which does not need any translation resource. This method gives an important role to features written in the same way in different languages when comparing the documents. ATMC algorithm gets similar results with respect to translating the search results for *highly multilingual person names*, but it makes the disambiguation process lighter, which is desirable in a scenario where users expect a quick response.

8.3. Future Work

There are several lines of work in order to improve the proposed methods described in this thesis:

- **Search results representation:**
  - Most of the mistakes produced by the proposed algorithm are due to common vocabulary used in the Internet. The impact of this kind of features is specially negative in web pages from social media. We propose as future work to explore several filtering techniques in order to avoid this kind of noisy features.
  - We have studied suitable features in order to decide when two search results are related to the same individual. However, it could be useful to study suitable features in order to determine when two search results do not talk about the same person in order to avoid incorrect groupings. Combining both kind of features could improve the final clustering. For instance, some features as the middle names or temporal information could be good indicatives that two documents are not related to the same individual.
- **Clustering:**
  - We have seen that comparing different documents by means of an adaptive threshold function is suitable with respect to use a fixed similarity threshold value as most systems based on HAC do. A future line of work could be to analyze the suitability of different kinds of mathematic functions in order to define adaptive thresholds. Furthermore, it would be interesting to study techniques to infer automatically adaptive threshold functions just taking into account the data characteristics.
  - The clustering algorithms proposed in this thesis do not provided overlap clusters because they assume the *one person per document* policy. Although this assumption usually holds in person name disambiguation [Artiles, 2009], it is possible to find web pages which talk about several individuals with the same name. Then, a future work could be to propose a soft clustering version of our algorithms.
  - The complexity cost of the proposed algorithms is in $\mathcal{O}(N^2)$ where N is the number of search results. One option could be to propose a new version of the algorithm which initially divide the ranking of search results in groups of k web pages where $k < N$, so the algorithm can be applied in each group

simultaneously and finally, propose a final phase to merge the *clusters* of the different groups.

- **Application in different disambiguation problems:** the document representation used in this thesis is based on *n*-grams and capitalized words. Both types of features are not related to a particular kind of query or domain. Then, we would like to check the suitability of our algorithms in other related problems involving different kind of queries (e.g. organization names), or different kind of documents (e.g. author name disambiguation, person name disambiguation on news). In addition, other future line of work would consist in adapting our algorithms to other disambiguation scenarios. For instance, entity linking differs of the problem studied in this thesis because there are entities already disambiguated in a knowledge base.

- **Social media and popular web sites:** we have seen that treating in a special way social web pages has a positive impact in the results. This idea could be applied to other popular web sites of different nature. Then, a future line of work could be to establish what kind of web pages play an special role in the problem and propose methods to deal with them. For instance, Wikipedia could be very useful to disambiguate different individuals because its disambiguation pages provide information about different individuals with the same name. Then, clusters containing different Wikipedia entries would correspond to different individuals.

- **Multilingualism:** ATMC employs features composed by capitalized words, used in languages that use the latin alphabet. Then, a future line of work could be to adapt our algorithm for languages that use other alphabets, for instance, Russian or Chinese. On the other hand, the set of *comparable features* could be enriched by means of NEs alignment techniques or cognate identification methods. Thus, a future line of work would be to explore these approaches in order to improve the results in the multilingual scenario.

8.4. Publications

Right after, we present the list of publications in conferences and journals conducted during the course of this thesis.

- Delgado, Agustín D.; Martínez, Raquel; Fresno, Víctor; y Montalvo, Soto. A data driven approach for person name disambiguation in web search results. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers*, August 23-29, 2014, Dublin, Ireland, pages 301–310. URL `http://aclweb.org/anthology/C/C14/C14-1030.pdf`.

- Delgado, Agustín D.; Martínez, Raquel; Montalvo, Soto; y Fresno, Víctor. An unsupervised algorithm for person name disambiguation in the web. *Procesamiento del Lenguaje Natural*, 53:51–58, 2014. URL `http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5042`.
- Delgado, Agustín D.; Martínez, Raquel; Montalvo, Soto; y Fresno, Víctor. Person name disambiguation in the web using adaptive threshold clustering. *Journal of the Association for Information Science and Technology (JASIST)*. URL `https://doi.org/10.1002/asi.23810`.
- Delgado, Agustín D.; Martínez, Raquel; Montalvo, Soto; y Fresno, Víctor. Tratamiento de redes sociales en desambiguación de nombres de persona en la web. *Procesamiento del Lenguaje Natural*, 57:117–124, 2016. URL `http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5344`.
- Delgado, Agustín D.; Martínez, Raquel; Montalvo, Soto; y Fresno, Víctor. Person name disambiguation on the web in a multilingual context. Manuscript submitted for publication. *Information Sciences*.

Bibliografía

Aggarwal, Charu C. y Zhai, ChengXiang. *A Survey of Text Clustering Algorithms*, páginas 77–128. Springer US, Boston, MA, 2012. ISBN 978-1-4614-3223-4. URL http://dx.doi.org/10.1007/978-1-4614-3223-4_4. 21, 33, 52, 125

Aizawa, Akiko N. An information-theoretic perspective of tf-idf measures. *Information Processing & Management*, 39(1):45–65, 2003. URL http://dx.doi.org/10.1016/S0306-4573(02)00021-3. 56

Al-Kamha, Reema y Embley, David W. Grouping search-engine returned citations for person-name queries. En *Proceedings of the 6th Annual ACM International Workshop on Web Information and Data Management*, WIDM '04, páginas 96–103, New York, NY, USA, 2004. ACM. ISBN 1-58113-978-0. URL http://doi.acm.org/10.1145/1031453.1031472. 3, 16

Amigó, Enrique; Gonzalo, Julio; Artiles, Javier; y Verdejo, Felisa. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486, 2009. URL http://dx.doi.org/10.1007/s10791-008-9066-8. 72

Andrade, Miguel A. y Valencia, Alfonso. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics*, 14 (7):600–607, 1998. URL http://dx.doi.org/10.1093/bioinformatics/14.7.600. 58

Artiles, Javier. *Web People Search*. PhD thesis, E.T.S. Ingeniería Inform´tica, UNED, 2009. URL http://e\discretionary{-}{}{}spacio.uned.es/fez/eserv/tesisuned:IngInf\discretionary{-}{}{}Jartiles/Documento.pdf. 9, 23, 24, 28, 33, 34, 37, 39, 49, 57, 61, 64, 70, 88, 91, 94, 98, 198, 205

Artiles, Javier; Gonzalo, Julio; y Sekine, Satoshi. The semeval-2007 weps evaluation: Establishing a benchmark for the web people search task. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval'07, páginas 64–69, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621486. 3, 17, 18, 34, 64, 72, 91, 106

Artiles, Javier; Amigó, Enrique; y Gonzalo, Julio. The role of named entities in web people search. En *Proceedings of the 2009 Conference on Empirical Methods in Natural*

Language Processing: Volume 2, EMNLP '09, páginas 534–542, Stroudsburg, PA, USA, 2009a. Association for Computational Linguistics. ISBN 978-1-932432-62-6. URL http://dl.acm.org/citation.cfm?id=1699571.1699582. 20, 21, 23, 24, 25, 28, 79, 83, 88

Artiles, Javier; Gonzalo, Julio; y Sekine, Satoshi. WePS 2 evaluation campaign: overview of the web people search clustering task. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009b. URL http://nlp.uned.es/weps/weps2/papers/weps2-clustering-task-description.pdf. 3, 18, 25, 34, 37, 39, 41, 65, 72, 73, 95, 106, 144, 145

Artiles, Javier; Borthwick, Andrew; Gonzalo, Julio; Sekine, Satoshi; y Amigó, Enrique. WePS-3 evaluation campaign: Evaluation campaign: Overview of the web people search clustering and attribute extraction tasks. En *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010. URL http://nlp.cs.nyu.edu/pubs/papers/10-009.pdf. 3, 4, 18, 34, 66, 106, 156

Bagga, Amit y Baldwin, Breck. Entity-based cross-document coreferencing using the vector space model. En *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING'98, páginas 79–85, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics. URL http://dx.doi.org/10.3115/980451.980859. 3, 8, 16, 34, 72

Balog, Krisztian; Azzopardi, Leif; y de Rijke, Maarten. UVA: language modeling techniques for web people search. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 468–471, 2007. URL http://aclweb.org/anthology/S/S07/S07-1104.pdf. 31, 109, 135

Balog, Krisztian; He, Jiyin; Hofmann, Katja; Jijkoun, Valentin; Monz, Christof; Tsagkias, Manos; Weerkamp, Wouter; y de Rijke, Maarten. The university of amsterdam at WePS2. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/UVA.pdf. 20, 28, 36, 39, 45, 61, 88, 91, 110, 136, 144, 145, 155, 164

Baron, Alex y Freedman, Marjorie. Who is who and what is what: Experiments in cross-document co-reference. En *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, páginas 274–283, 2008. URL http://www.aclweb.org/anthology/D08-1029. 8

Bekkerman, Ron y McCallum, Andrew. Disambiguating web appearances of people in a social network. En *Proceedings of the 14th International Conference on World Wide Web*, WWW'05, páginas 463–470, New York, NY, USA, 2005. ACM. ISBN 1-59593-046-9. URL http://doi.acm.org/10.1145/1060745.1060813. 3, 16, 17, 32, 119, 140

Bentivogli, Luisa; Marchetti, Alessandro; y Pianta, Emanuele. *The News People Search Task at EVALITA 2011: Evaluating Cross-Document Coreference Resolution of Named Person Entities in Italian News*, páginas 126–134. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-35828-9. URL http://dx.doi.org/10.1007/978-3-642-35828-9_14. 8, 72

Berendsen, Richard. *Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation*. PhD thesis, Informatics Institute, University of Amsterdam, 2015. URL http://dare.uva.nl/document/2/165379. XX, 10, 24, 28, 34, 45, 47, 49, 50, 88, 91, 144, 145, 146, 149, 151, 152, 153, 155, 158, 163, 164, 165, 168, 198, 200

Berendsen, Richard; Kovachev, Bogomil; Nastou, Evangelia-Paraskevi; de Rijke, Maarten; y Weerkamp, Wouter. Result disambiguation in web people search. En *Advances in Information Retrieval: 34th European Conference on IR Research, ECIR 2012, Barcelona, Spain, April 1-5, 2012. Proceedings*, páginas 146–157, Berlin, Heidelberg, 2012. Springer. ISBN 978-3-642-28997-2. URL http://dx.doi.org/10.1007/978-3-642-28997-2_13. 41, 43, 63, 67, 72, 144, 158

Blei, David M.; Ng, Andrew Y.; y Jordan, Michael I. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. URL http://www.jmlr.org/papers/v3/blei03a.html. 29

Bollegala, Danushka; Matsuo, Yutaka; y Ishizuka, Mitsuru. Disambiguating personal names on the web using automatically extracted key phrases. En *ECAI 2006, 17th European Conference on Artificial Intelligence, August 29 - September 1, 2006, Riva del Garda, Italy, Including Prestigious Applications of Intelligent Systems (PAIS 2006), Proceedings*, páginas 553–557, 2006. 3, 16, 17, 63

Bosch, Antal; Bogers, Toine; y Kunder, Maurice. Estimating search engine index size variability: A 9-year longitudinal study. *Scientometrics*, 107(2):839–856, 2016. ISSN 0138-9130. URL http://dx.doi.org/10.1007/s11192-016-1863-z. 1

Brin, Sergey y Page, Lawrence. The anatomy of a large-scale hypertextual web search engine. En *Proceedings of the Seventh International Conference on World Wide Web 7*, WWW7, páginas 107–117, Amsterdam, The Netherlands, 1998. Elsevier Science Publishers B. V. URL http://dl.acm.org/citation.cfm?id=297805.297827. 1, 30

Caliński, Tadeusz y Harabasz, Jerzy. A dendrite method for cluster analysis. *Communications in Statistics-Simulation and Computation*, 3(1):1–27, 1974. 35

Carmel, David; Chang, Ming-Wei; Gabrilovich, Evgeniy; Hsu, Bo-June (Paul); y Wang, Kuansan. ERD'14: Entity recognition and disambiguation challenge. *SIGIR Forum*, 48(2):63–77, dec 2014. ISSN 0163-5840. URL http://doi.acm.org/10.1145/2701583.2701591. 7

Caropreso, María Fernanda; Matwin, Stan; y Sebastiani, Fabrizio. Text databases & document management. En *A Learner-independent Evaluation of the Usefulness of Statistical Phrases for Automated Text Categorization*, páginas 78–102. IGI Global, Hershey, PA, USA, 2001. ISBN 1-878289-93-4. URL http://dl.acm.org/citation.cfm?id=374247.374254. 79

Chen, Ying y Martin, James. CU-COMSEM: Exploring rich features for unsupervised web personal name disambiguation. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 125–128, Stroudsburg, PA, USA, 2007a. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621498. 24, 26, 28, 45, 46, 109, 135, 144, 164

Chen, Ying y Martin, James. Towards robust unsupervised personal name disambiguation. En *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic*, páginas 190–198, 2007b. URL http://www.aclweb.org/anthology/D07-1020. 43

Chen, Ying; Lee, Sophia Yat Mei; y Huang, Chu-Ren. PolyUHK: A robust information extraction system for web personal names. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/PolyUHK.pdf. 110, 111, 136

Chen, Ying; Yat Mei Lee, Sophia; y Huang, Chu-Ren. A robust web personal name information extraction system. *Expert Systems with Applications*, 39(3):2690–2699, February 2012. ISSN 0957-4174. URL http://dx.doi.org/10.1016/j.eswa.2011.08.125. 19, 24, 25, 30, 31, 140

Chinchor, Nancy. Muc-7 named entity task definition (version 3.5). En *Proceedings of the 7th Message Understanding Conference*, 1997. URL http://www-nlpir.nist.gov/related_projects/muc/proceedings/ne_task.html. 16

Cigarrán, Juan Manuel. *Organización de resultados de búsqueda mediante análisis formal de conceptos*. PhD thesis, E.T.S. Ingeniería Informática, UNED, 2008. URL http://e-spacio.uned.es/fez/eserv/tesisuned:IngInf-Jcigarran/Documento.pdf. 33

Coll Ardanuy, Mariona; van den Bos, Maarten; y Sporleder, Caroline. You shall know people by the company they keep: Person name disambiguation for social network construction. En *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, páginas 63–73, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://anthology.aclweb.org/W16-2107. 8

Cummins, Ronan. A standard document score for information retrieval. En *International Conference on the Theory of Information Retrieval, ICTIR '13, Copenhagen, Denmark, September 29 - October 02, 2013*, page 24, 2013. URL http://doi.acm.org/10.1145/2499178.2499183. 58

Day, William H. E. y Edelsbrunner, Herbert. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1(1):7–24, 1984. ISSN 1432-1343. URL http://dx.doi.org/10.1007/BF01890115. 37

Deerwester, Scott C.; Dumais, Susan T.; Landauer, Thomas K.; Furnas, George W.; y Harshman, Richard A. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990. URL http://dx.doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9. 29

Defays, Daniel. An efficient algorithm for a complete link method. *The Computer Journal*, 20(4):364–366, 1977. URL http://dx.doi.org/10.1093/comjnl/20.4.364. 37

del Valle-Agudo, David; de Pablo-S´nchez, C´sar; y Vicente-Díez, María Teresa. UC3M_13: Disambiguation of person names based on the composition of simple bags of typed terms. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 362–365, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621553. 109, 135

Doddington, George R.; Mitchell, Alexis; Przybocki, Mark A.; Ramshaw, Lance A.; Strassel, Stephanie; y Weischedel, Ralph M. The automatic content extraction (ACE) program - tasks, data, and evaluation. En *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004, May 26-28, 2004, Lisbon, Portugal*, 2004. URL http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf. 16

Dornescu, Iustin; Orasan, Constantin; y Lesnikova, Tatiana. Cross-document coreference for weps. En *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010. URL http://clef2010.clef-initiative.eu/resources/proceedings/clef2010labs_submission_103_new.pdf. 9, 26, 27, 28, 36, 40, 61, 80, 111, 137

Duque, Andrés. *Word Sense Disambiguation in Multilingual Contexts*. PhD thesis, E.T.S. Ingeniería Informática, UNED, 2017. URL http://e-spacio.uned.es/fez/eserv/tesisuned:IngInf-Aduque/DUQE_FERNANDEZ_ANDRES_Tesis.pdf. 170

Ellman, Jeremy y Emery, Gary. UNN-WePS: Web person search using co-present names and lexical chains. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 402–405, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621563. 109, 135

Elmacioglu, Ergin; Tan, Yee Fan; Yan, Su; Kan, Min-Yen; y Lee, Dongwon. PSNUS: web people name disambiguation by simple clustering with rich features. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 268–271, 2007. URL http://aclweb.org/anthology/S/S07/S07-1058.pdf. 24, 26, 88, 109, 135

Ferguson, Thomas S. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973. URL http://www.jstor.org/stable/2958008. 31

Ferreira, Anderson A.; Gonçalves, Marcos André; y Laender, Alberto H.F. A brief survey of automatic methods for author name disambiguation. *SIGMOD Rec.*, 41(2):15–26, Agosto 2012. ISSN 0163-5808. URL http://doi.acm.org/10.1145/2350036.2350040. 8

Ferrés, Daniel y Rodríguez, Horacio. TALP at weps-3 2010. En *CLEF 2010 LABs and Workshops, Notebook Papers, 22-23 September 2010, Padua, Italy*, 2010. URL http://ceur-ws.org/Vol-1176/CLEF2010wn-WePS-FerresEt2010.pdf. 19, 26, 111, 137

Finkel, Jenny Rose; Grenager, Trond; y Manning, Christopher. Incorporating non-local information into information extraction systems by gibbs sampling. En *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, páginas 363–370, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. URL https://doi.org/10.3115/1219840.1219885. 23, 80, 83

Fleischman, Michael Ben y Hovy, Eduard. Multi-document person name resolution. En *In 42nd Annual Meeting of the Association for Computational Linguistics (ACL), Reference Resolution Workshop*, páginas 1–8, Stroudsburg, PA, USA, July 2004. Association for Computational Linguistics. URL http://aclweb.org/anthology/W04-0701. 8

Fresno, Víctor. *Representación Autocontenida de documentos HTML: una propuesta basada en combinaciones heurísticas de criterios*. PhD thesis, Universidad Rey Juan Carlos (URJC), 2006. URL http://www.escet.urjc.es/~vfresno/tesis_VFresno.pdf. 29

Frey, Brendan J. y Dueck, Delbert. Clustering by passing messages between data points. *Science*, 2007. ISSN 0036-8075. doi: 10.1126/science.1136800. URL http://www.psi.toronto.edu/affinitypropagation/FreyDueckScience07.pdf. 62, 63, 83

Fujiwara, Yasuhiro; Irie, Go; y Kitahara, Tomoe. Fast algorithm for affinity propagation. En *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, páginas 2238–2243, 2011. URL http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-373. 63

Gale, William A.; Church, Kenneth W.; y Yarowsky, David. One sense per discourse. En *Proceedings of the Workshop on Speech and Natural Language*, HLT '91, páginas 233–

237, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics. ISBN 1-55860-272-0. URL http://dx.doi.org/10.3115/1075527.1075579. 34, 98

Gaussier, Éric; Goutte, Cyril; Popat, Kris; y Chen, Francine. *A Hierarchical Model for Clustering and Categorising Documents*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. ISBN 978-3-540-45886-9. URL http://dx.doi.org/10.1007/3-540-45886-7_16. 33

Geiß, Johanna y Gertz, Michael. With a little help from my neighbors: Person name linking using the wikipedia social network. En *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW'16 Companion, páginas 985–990, Republic and Canton of Geneva, Switzerland, 2016. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-4144-8. URL http://dx.doi.org/10.1145/2872518.2891109. 9

Girolami, Mark A. y Kabán, Ata. On an equivalence between PLSI and LDA. En *SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 28 - August 1, 2003, Toronto, Canada*, páginas 433–434, 2003. URL http://doi.acm.org/10.1145/860435.860537. 30

Gong, Jun y Oard, Douglas. Determine the entity number in hierarchical clustering for web personal name disambiguation. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. 24, 110, 136

González, José Carlos; Maté, Pablo; Vadillo, Laura; Sotomayor, Rocío; y Carrera, Álvaro. Learning by doing: A baseline approach to the clustering of web people search results. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/UPM.pdf. 25, 110, 136

Gooi, Chung Heong y Allan, James. Cross-document coreference on a large scale corpus. En *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2004, Boston, Massachusetts, USA, May 2-7, 2004*, páginas 9–16, 2004. URL http://aclweb.org/anthology/N/N04/N04-1002.pdf. 8, 16

Grishman, Ralph y Sundheim, Beth. Message understanding conference-6: A brief history. En *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, páginas 466–471, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics. URL http://dx.doi.org/10.3115/992628.992709. 16

Grütze, Toni; Kasneci, Gjergji; Zuo, Zhe; y Naumann, Felix. Bootstrapping wikipedia to answer ambiguous person name queries. En *Workshops Proceedings of the 30th International Conference on Data Engineering Workshops, ICDE 2014, Chicago, IL, USA,*

March 31 - April 4, 2014, páginas 56–61, 2014. doi: 10.1109/ICDEW.2014.6818303. URL http://dx.doi.org/10.1109/ICDEW.2014.6818303. 8

Guthrie, David; Allison, Ben; Liu, Wei; Guthrie, Louise; y Wilks, Yorick. A closer look at skip-gram modelling. En *Proceedings of the 5thInternational Conference on Language Resources and Evaluation (LREC 2006). Genoa, Italy.*, páginas 1222–1225, 2006. URL http://www.lrec-conf.org/proceedings/lrec2006/pdf/357_pdf.pdf. 25, 82, 88

Han, Xianpei y Zhao, Jun. CASIANED: People attribute extraction based on information extraction. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/CL-CASIANED.pdf. 26, 27, 110, 136

Han, Xianpei y Zhao, Jun. Structural semantic relatedness: A knowledge-based method to named entity disambiguation. En *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, páginas 50–59, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1858681.1858687. 32, 34

He, Bin; Patel, Mitesh; Zhang, Zhen; y Chang, Kevin Chen-Chuan. Accessing the deep web. *Communications of the ACM*, 50(5):94–101, may 2007. ISSN 0001-0782. doi: 10.1145/1230819.1241670. URL http://doi.acm.org/10.1145/1230819.1241670. 2

He, Zhengyan; Wang, Houfeng; y Li, Sujian. The task 2 of CIPS-SIGHAN 2012 named entity recognition and disambiguation in chinese bakeoff. En *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, páginas 108–114, Stroudsburg, PA, USA, December 2012. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W12-6321. 8

Heyer, Laurie J.; Kruglyak, Semyon; y Yooseph, Shibu. Exploring Expression Data: Identification and Analysis of Coexpressed Genes. *Genome Research*, 9(11):1106–1115, 1999. 39

Heyl, Andrea y Neumann, Günter. DFKI2: an information extraction based approach to people disambiguation. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 137–140, 2007. URL http://aclweb.org/anthology/S/S07/S07-1027.pdf. 35, 94, 109, 135

Hiemstra, Djoerd. A probabilistic justification for using tf x idf term weighting in information retrieval. *International Journal on Digital Libraries*, 3(2):131–139, 2000. URL http://dx.doi.org/10.1007/s007999900025. 56

Hill, D. R. A vector clustering technique. *Mechanized Information Storage, Retrieval and Dissemination*, 1968. 39

Hoffart, Johannes; Yosef, Mohamed Amir; Bordino, Ilaria; Fürstenau, Hagen; Pinkal, Manfred; Spaniol, Marc; Taneva, Bilyana; Thater, Stefan; y Weikum, Gerhard. Robust disambiguation of named entities in text. En *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, páginas 782–792, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145521. 7

Hofmann, Thomas. Probabilistic latent semantic indexing. En *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, páginas 50–57, New York, NY, USA, 1999. ACM. ISBN 1-58113-096-1. URL http://doi.acm.org/10.1145/312624.312649. 29, 33

Hrbacek, Karel y Jech, Thomas J. *Introduction to set theory*. Monographs and textbooks in pure and applied mathematics. M. Dekker, New York, 1999. ISBN 0-585-24341-7. URL http://opac.inria.fr/record=b1097588. 100

Huang, Anna. Similarity measures for text document clustering. En *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, páginas 49–56, 2008. URL http://www.academia.edu/download/32952068/pg049_Similarity_Measures_for_Text_Document_Clustering.pdf. 59

Huang, Hsun-Hui y Kuo, Yau-Hwang. Cross-lingual document representation and semantic similarity measure: A fuzzy set and rough set based approach. *IEEE Transactions on Fuzzy Systems*, 18(6):1098–1111, Diciembre 2010. ISSN 1063-6706. URL http://dx.doi.org/10.1109/TFUZZ.2010.2065811. 29

Ikeda, Masaki; Ono, Shingo; Sato, Issei; Yoshida, Minoru; y Nakagawa, Hiroshi. Person name disambiguation on the web by twostage clustering. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. 23, 24, 30, 34, 37, 110, 136

Iria, José; Xia, Lei; y Zhang, Ziqi. WIT: web people search disambiguation using random walks. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 480–483, 2007. URL http://aclweb.org/anthology/S/S07/S07-1107.pdf. 32, 35, 40, 109, 119, 135

Jaccard, Paul. Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:241 – 272, 1901. 59, 60

Jiang, Lili; Wang, Jianyong; An, Ning; Wang, Shengyuan; Zhan, Jian; y Li, Lian. GRAPE: A graph-based framework for disambiguating people appearances in web search. En *ICDM 2009, The Ninth IEEE International Conference on Data Mining, Miami, Florida,*

USA, 6-9 December 2009, páginas 199–208, 2009. URL http://dx.doi.org/10.1109/ICDM.2009.25. 21, 27, 32, 40, 45, 47, 93, 109, 110, 111, 116, 118, 135, 136, 137, 139, 140, 148, 159, 202

Jones, Karen Spärck. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 01 1972. ISSN 0022-0418. doi: 10.1108/eb026526. 56, 127

Kalmar, Paul y Blume, Matthias. FICO: Web person disambiguation via weighted similarity of entity contexts. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 149–152, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621504. 109, 135

Kalmar, Paul y Freitag, Dayne. Features for web person disambiguation. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/FICO.pdf. 110, 136

Kaufman, Leonard y Rousseeuw, Peter J. *Clustering by Means of Medoids*. Reports of the Faculty of Mathematics and Informatics. Faculty of Mathematics and Informatics, 1987. URL https://books.google.es/books?id=HK-4GwAACAAJ. 39

Kozareva, Zornitsa y Ravi, Sujith. Unsupervised name ambiguity resolution using a generative model. En *Proceedings of the First Workshop on Unsupervised Learning in NLP*, EMNLP '11, páginas 105–112, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-13-8. URL http://dl.acm.org/citation.cfm?id=2140458.2140471. 31, 32, 43

Kozareva, Zornitsa; Vázquez, Sonia; y Montoyo, Andrés. UA-ZSA: web page clustering on the basis of name disambiguation. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 338–341, 2007. URL http://aclweb.org/anthology/S/S07/S07-1073.pdf. 20, 26, 31, 39, 109, 119, 135

Lan, Man; Zhang, Yu Zhe; Lu, Yue; Su, Jian; y Tan, Chew Lim. Which who are they? people attribute extraction and disambiguation in web search results. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/ECNU.pdf. 19, 27, 39, 109, 110, 136

Lana-Serrano, Sara; Villena-Román, Julio; y González, José Carlos. DAEDALUS at webps-3 2010: k-medoids clustering using a cost function minimization. En *CLEF 2010 LABs and Workshops, Notebook Papers, 22-23 September 2010, Padua, Italy*, 2010. URL http://ceur-ws.org/Vol-1176/CLEF2010wn-WePS-Lana-SerranoEt2010.pdf. 39, 88, 109, 111, 137

Lancaster, Frederick Wilfrid y Gallup, Emily. *Information retrieval on-line*. Melville Publishing Co, Los Angeles, CA, USA, 1973. 29

Lefever, Els; Hoste, Véronique; y Fayruzov, Timur. Aug: A combined classification and clustering approach for web people disambiguation. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 105–108, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621493. 26, 27, 109, 135

Lefever, Els; Fayruzov, Timur; Hoste, Véronique; y de Cock, Martine. Fuzzy ants clustering for web people search. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/AUG.pdf. 25, 26, 40, 41, 94, 110, 136

Liu, Zhengzhong; Lu, Qin; y Xu, Jian. High performance clustering for web person name disambiguation using topic capturing. En *Proceedings of The First International Workshop on Entity-Oriented Search (EOS)*, páginas 1–6, New York, NY, USA, 2011. ACM. URL http://research.microsoft.com/en-us/um/beijing/events/eos2011/9.pdf. 19, 20, 24, 26, 28, 30, 34, 45, 47, 109, 110, 135, 136, 144, 155, 164, 202

Long, Chong y Shi, Lei. Web person name disambiguation by relevance weighting of extended feature sets. En *CLEF 2010 LABs and Workshops, Notebook Papers, 22-23 September 2010, Padua, Italy*, 2010. URL http://ceur-ws.org/Vol-1176/CLEF2010wn-WePS-LongEt2010.pdf. 9, 24, 27, 28, 29, 30, 38, 45, 46, 94, 111, 112, 137, 138, 144, 155, 164

Luhn, Hans Peter. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):309–317, Octubre 1957. ISSN 0018-8646. URL http://dx.doi.org/10.1147/rd.14.0309. 55, 56

MacQueen, J. Some methods for classification and analysis of multivariate observations. En *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, páginas 281–297, 1967. 39

Mann, Gideon S. *Multi-Document Statistical Fact Extraction and Fusion*. PhD thesis, Johns Hopkins University, Baltimore, MD, USA, 2006. URL https://www.cs.jhu.edu/~gsm/publications/mann06thesis.pdf. 64, 90

Mann, Gideon S. y Yarowsky, David. Unsupervised personal name disambiguation. En *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CoNLL'03, páginas 33–40, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics. URL http://dx.doi.org/10.3115/1119176.1119181. 3, 16, 17, 26, 31, 43, 63, 169

Manning, Christopher D.; Raghavan, Prabhakar; y Schütze, Hinrich. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008. ISBN 0521865719, 9780521865715. 31, 35, 37, 40, 125, 126

Martínez, Raquel; Abaitua, Joseba; y Casillas, Arantza. Aligning tagged bitexts. En *Proceedings of the Sixth Workshop on Very Large Corpora, COLING-ACL'98*, páginas 102–109, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics. URL http://www.aclweb.org/website/old_anthology/W/W98/W98-1112.pdf. 195

Martínez-Romo, Juan y Araujo, Lourdes. Web people search disambiguation using language model techniques. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/UNED.pdf. 20, 27, 31, 110, 136

Mikolov, Tomas; Sutskever, Ilya; Chen, Kai; Corrado, Gregory S.; y Dean, Jeffrey. Distributed representations of words and phrases and their compositionality. En *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, páginas 3111–3119, 2013. URL http://dl.acm.org/citation.cfm?id=2999792.2999959. 25

Miller, George A. WordNet: A lexical database for english. *Commun. ACM*, 38(11): 39–41, Noviembre 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL http://doi.acm.org/10.1145/219717.219748. 27

Momeni, Fakhri y Mayr, Philipp. Using co-authorship networks for author name disambiguation. En *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries*, JCDL '16, páginas 261–262, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4229-2. doi: 10.1145/2910896.2925461. URL http://doi.acm.org/10.1145/2910896.2925461. 9

Monmarché, Nicolas. *Algorithmes de fourmis artificielles : applications à la classification et à l'optimisation. (Artificial ant based algorithms applied to clustering and optimization problems)*. PhD thesis, François Rabelais University, Tours, France, 2000. URL https://tel.archives-ouvertes.fr/tel-00005186. 40

Montalvo, Soto. *Estudio y nuevas estrategias en el uso de las Entidades Nombradas en el Clustering Bilingüe de noticias*. PhD thesis, Universidad Rey Juan Carlos (URJC), 2012. URL https://eciencia.urjc.es/bitstream/handle/10115/12075/sotoThesis2012.pdf. 8, 170, 195

Montalvo, Soto; Martínez, Raquel; Fresno, Víctor; y Capilla, Rafael. Multilingual information access on the web. *IEEE Computer*, 48(7):73–75, 2015a. 7, 10, 170

Montalvo, Soto; Martínez, Raquel; Fresno, Víctor; y Delgado, Agustín D. Exploiting named entities for bilingual news clustering. *JASIST*, 66(2):363–376, 2015b. URL http://dx.doi.org/10.1002/asi.23175. 74

Montalvo, Soto; Martínez, Raquel; Campillos, Leonardo; Delgado, Agustín D.; Fresno, Víctor; y Verdejo, Felisa. MC4WEPS: a multilingual corpus for web people search disambiguation. *Language Resources and Evaluation*, páginas 1–28, 2016. ISSN 1574-0218. doi: 10.1007/s10579-016-9365-4. URL http://dx.doi.org/10.1007/s10579-016-9365-4. 63, 68, 69, 72, 158, 179

Monz, Christof y Weerkamp, Wouter. A comparison of retrieval-based hierarchical clustering approaches to person name disambiguation. En *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, páginas 650–651, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-483-6. URL http://doi.acm.org/10.1145/1571941.1572060. 20, 24, 27, 28, 31, 56

Nagy, István T. Person attribute extraction from the textual parts of web pages. *Acta Cybern.*, 20(3):419–439, Agosto 2012. ISSN 0324-721X. URL http://dx.doi.org/10.14232/actacyb.20.3.2012.4. 21, 26, 28, 111, 137, 140

Nastase, Vivi; Mihalcea, Rada; y Radev, Dragomir R. A survey of graphs in natural language processing. *Natural Language Engineering*, 21(5):665–698, 2015. URL http://journals.cambridge.org/article_S1351324915000340. 30

Navigli, Roberto. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2):10:1–10:69, feb 2009. ISSN 0360-0300. URL http://doi.acm.org/10.1145/1459352.1459355. 16

Nithya, N. S.; Duraiswamy, K.; y Gomathy, P. A survey on clustering techniques in medical diagnosis. *International Journal of Computer Science Trends and Technology (IJCST)*, 1(2):17–23, 2013. 33

Nugent, Rebecca y Meila, Marina. *An Overview of Clustering Applied to Molecular Biology*. Humana Press, Totowa, NJ, 2010. ISBN 978-1-60761-580-4. URL http://dx.doi.org/10.1007/978-1-60761-580-4_12. 33

Nuray-Turan, Rabia; Kalashnikov, Dmitri V.; y Mehrotra, Sharad. Exploiting web querying for web people search. *ACM Transactions on Database Systems*, 37(1):7:1–7:41, March 2012. ISSN 0362-5915. doi: 10.1145/2109196.2109203. URL http://doi.acm.org/10.1145/2109196.2109203. 19, 20, 23, 25, 26, 27, 28, 29, 32, 40, 45, 46, 47, 79, 81, 88

Ono, Shingo; Sato, Issei; Yoshida, Minoru; y Nakagawa, Hiroshi. *Person Name Disambiguation in Web Pages Using Social Network, Compound Words and Latent Topics*, pá-

ginas 260–271. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-68125-0. doi: 10.1007/978-3-540-68125-0_24. URL http://dx.doi.org/10.1007/978-3-540-68125-0_24. 24, 31

Pedersen, Ted; Purandare, Amruta; y Kulkarni, Anagha. Name discrimination by clustering similar contexts. En *Computational Linguistics and Intelligent Text Processing, 6th International Conference, CICLing 2005, Mexico City, Mexico, February 13-19, 2005, Proceedings*, páginas 226–237, 2005. URL http://dx.doi.org/10.1007/978-3-540-30586-6_24. 63

Pimienta, Daniel; Prado, Daniel; y 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. URL http://unesdoc.unesco.org/images/0018/001870/187016e.pdf. 7, 43, 50, 140, 170, 198

Pinto, David; Tovar, Mireya; Vilariño, Darnes; Díaz, Héctor; y Jiménez-Salazar, Héctor. An unsupervised approach based on fingerprinting to the web people search task. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/BUAP.pdf. 110, 136

Popescu, Octavian y Magnini, Bernardo. IRST-BP: web people search using name entities. En *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval@ACL 2007, Prague, Czech Republic, June 23-24, 2007*, páginas 195–198, 2007. URL http://aclweb.org/anthology/S/S07/S07-1041.pdf. 23, 25, 28, 45, 79, 88, 109, 135

Porter, Martin F. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980. doi: 10.1108/eb046814. URL http://dx.doi.org/10.1108/eb046814. 20, 52, 83

Rao, Delip; Garera, Nikesh; y Yarowsky, David. JHU1: An unsupervised approach to person name disambiguation using web snippets. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 199–202, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621516. 24, 25, 26, 28, 29, 39, 109, 135

Robertson, Stephen. Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation*, 60(5):503–520, 2004. URL http://dx.doi.org/10.1108/00220410410560582. 56

Romano, Lorenza; Buza, Krisztian; Giuliano, Claudio; y Schmidt-Thieme, Lars. XMedia: Web people search by clustering with machinely learned similarity measures. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/XMedia.pdf. 24, 31, 39, 110, 136

Rozovskaya, Alla y Roth, Dan. Grammatical error correction: Machine translation and classifiers. En *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume1: Long Papers*, 2016. URL http://aclweb.org/anthology/P/P16/P16-1208.pdf. 183

Saggion, Horacio. SHEF: Semantic tagging and summarization techniques applied to cross-document coreference. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 292–295, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621537. 56, 88, 109, 135

Saggion, Horacio. Experiments on semantic-based clustering for cross-document coreference. En *Proceedings of the Third Joint International Conference on Natural Language Processing*, páginas 149–156, Hyderabad, India, January 2008. AFNLP, AFNLP. URL http://aclweb.org/anthology-new/I/I08/I08-1020.pdf. 28, 83

Salton, Gerald; Wong, A.; y Yang, Chung-Shu. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, nov 1975. ISSN 0001-0782. doi: 10.1145/361219.361220. URL http://doi.acm.org/10.1145/361219.361220. 53

Salton, Gerard y Buckley, Chris. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988. URL http://dx.doi.org/10.1016/0306-4573(88)90021-0. 31, 56, 127

Salton, Gerard; Singhal, Amit; Mitra, Mandar; y Buckley, Chris. Automatic text structuring and summarization. *Inf. Process. Manage.*, 33(2):193–207, 1997. URL http://dx.doi.org/10.1016/S0306-4573(96)00062-3. 29

Sang, Erik F. Tjong Kim. Introduction to the conll-2002 shared task: Language-independent named entity recognition. En *Proceedings of the 6th Conference on Natural Language Learning, CoNLL 2002, Held in cooperation with COLING 2002, Taipei, Taiwan, 2002*, 2002. URL http://aclweb.org/anthology/W/W02/W02-2024.pdf. 16

Sang, Erik F. Tjong Kim y de Meulder, Fien. Introduction to the conll-2003 shared task: Language-independent named entity recognition. En *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, páginas 142–147, 2003. URL http://aclweb.org/anthology/W/W03/W03-0419.pdf. 16

Schockaert, Steven; Cock, Martine De; Cornelis, Chris; y Kerre, Etienne E. Clustering web search results using fuzzy ants. *International Journal of Intelligent Systems*, 22(5):455–474, 2007. URL http://dx.doi.org/10.1002/int.20209. 40

Sedding, Julian y Kazakov, Dimitar. Wordnet-based text document clustering. En *Proceedings of the 3rd Workshop on RObust Methods in Analysis of Natural Language Data*, ROMAND '04, páginas 104–113, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621445.1621458. 18

Sekine, Satoshi. Extended named entity ontology with attribute information. En *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*, 2008. URL http://www.lrec-conf.org/proceedings/lrec2008/summaries/21.html. 23

Sekine, Satoshi; Sudo, Kiyoshi; y Nobata, Chikashi. Extended named entity hierarchy. En *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, Las Palmas, Canary Islands - Spain, May 2002. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2002/pdf/120.pdf. 16

Shannon, Claude E. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, July 1948. ISSN 0005-8580. doi: 10.1002/j.1538-7305.1948.tb01338.x. 56

Shen, Wei; Wang, Jianyong; y Han, Jiawei. Entity linking with a knowledge base: Issues, techniques, and solutions. *Transactions on Knowledge & Data Engineering*, 27(2):443–460, 2015. doi: 10.1109/TKDE.2014.2327028. URL http://www.computer.org/csdl/trans/tk/2015/02/06823700-abs.html. 9

Sibson, R. SLINK: an optimally efficient algorithm for the single-link cluster method. *Comput. J.*, 16(1):30–34, 1973. URL http://dx.doi.org/10.1093/comjnl/16.1.30. 37

Smalheiser, Neil R. y Torvik, Vetle I. Author name disambiguation. *ARIST*, 43(1):1–43, 2009. doi: 10.1002/aris.2009.1440430113. URL http://dx.doi.org/10.1002/aris.2009.1440430113. 9

Smirnova, Elena; Avrachenkov, Konstantin; y Trousse, Brigitte. Using web graph structure for person name disambiguation. En *CLEF 2010 LABs and Workshops, Notebook Papers, 22-23 September 2010, Padua, Italy*, 2010. URL http://ceur-ws.org/Vol-1176/CLEF2010wn-WePS-SmirnovaEt2010.pdf. 26, 32, 40, 45, 46, 47, 56, 111, 137

Song, Fei; Cohen, Robin; y Lin, Song. Web people search based on locality and relative similarity measures. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. 25, 27, 28, 110, 136

Steinhaus, Hugo. Sur la division des corps matériels en parties. *Bulletin de l'Academie Polonesa des Sciences Cl. III. 4*, páginas 801–804, 1956. 38

Strehl, Alexander; Ghosh, Joydeep; y Mooney, Raymond. Impact of similarity measures on web-page clustering. En *Workshop on artificial intelligence for web search (AAAI 2000)*, páginas 58–64. American Association for Artificial Intelligence, 2000. URL https://vvvvw.aaai.org/Papers/Workshops/2000/WS-00-01/WS00-01-011.pdf. 58, 59

Sugiyama, Kazunari y Okumura, Manabu. TITPI: Web people search task using semi-supervised clustering approach. En *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, páginas 318–321, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621543. 109, 135

Toba, Hapnes; Wijaya, Evelyn A.; Wijanto, Maresha C.; y Karnalim, Oscar. Enhanced unsupervised person name disambiguation to support alumni tracer study. *Global Journal of Engineering Education (GJEE)*, 19(1):42–48, 2017. 203

van Rijsbergen, Cornelis J. Foundation of Evaluation. *Journal of Documentation*, 30(4): 365–373, 1974. 73

van Rijsbergen, Cornelis J. *Information Retrieval*. Butterworth, 1979. ISBN 0-408-70929-4. 33

Venkateshan, Priya. Clustering web people search results using fuzzy ant-based clustering. En *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009. URL http://nlp.uned.es/weps/weps2/papers/PRIYAVEN.pdf. 25, 40, 88, 109, 110, 136

Vidden, Chad; Vriens, Marco; y Chen, Song. Comparing clustering methods for market segmentation: A simulation study. *Applied Marketing Analytics*, 2(3):225–238, 2016. 33

Wan, Xiaojun; Gao, Jianfeng; Li, Mu; y Ding, Binggong. Person resolution in person search results: Webhawk. En *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*, páginas 163–170, 2005. URL http://doi.acm.org/10.1145/1099554.1099585. 3, 16, 17, 64

Weerkamp, Wouter; Berendsen, Richard; Kovachev, Bogomil; Meij, Edgar; Balog, Krisztian; y de Rijke, Maarten. People searching for people: analysis of a people search engine log. En *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29, 2011*, páginas 45–54, 2011. URL http://doi.acm.org/10.1145/2009916.2009927. 67, 70, 159

Wick, Michael; Singh, Sameer; y McCallum, Andrew. A discriminative hierarchical model for fast coreference at large scale. En *Proceedings of the 50th Annual Meeting of the*

Association for Computational Linguistics: Long Papers - Volume 1, ACL '12, páginas 379–388, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=2390524.2390578. 32, 45

Wilcoxon, Frank. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6): 80–83, December 1945. ISSN 00994987. doi: 10.2307/3001968. URL http://dx.doi.org/10.2307/3001968. 74

Winchester, Dan y Lee, Mark. Using proper names to cluster documents. En *In Acquiring (and Using) Linguistic (and World) Knowledge for Information Access: Papers from the spring Symposium (Technical Report SS-02-09), Menlo Park*, páginas 3–8, 2002. 3, 8, 16

Xu, Jian; Lu, Qin; y Liu, Zhengzhong. Aggregating skip bigrams into key phrase-based vector space model for web person disambiguation. En *11th Conference on Natural Language Processing, KONVENS 2012, Empirical Methods in Natural Language Processing, Vienna, Austria, September 19-21, 2012*, páginas 108–117. ÖGAI, 2012. URL http://www.oegai.at/konvens2012/proceedings/16_xu12o/. 25

Xu, Jian; Lu, Qin; Li, Minglei; y Li, Wenjie. Web person disambiguation using hierarchical co-reference model. En *Computational Linguistics and Intelligent Text Processing - 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part I*, páginas 279–291, 2015. doi: 10.1007/978-3-319-18111-0_22. URL http://dx.doi.org/10.1007/978-3-319-18111-0_22. 9, 24, 25, 26, 28, 32, 38, 44, 45, 46, 47, 49, 94, 109, 110, 111, 112, 119, 135, 136, 144, 148, 159

Xu, Yu-Meng; Wang, Chang-Dong; y Lai, Jian-Huang. Weighted multi-view clustering with feature selection. *Pattern Recognition*, 53:25–35, 2016. doi: 10.1016/j.patcog.2015.12.007. URL http://dx.doi.org/10.1016/j.patcog.2015.12.007. 21

Yoshida, Minoru; Ikeda, Masaki; Ono, Shingo; Sato, Issei; y Nakagawa, Hiroshi. Person name disambiguation by bootstrapping. En *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, páginas 10–17, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0153-4. doi: 10.1145/1835449.1835454. URL http://doi.acm.org/10.1145/1835449.1835454. 25, 26, 34, 45, 46, 47, 93, 110, 111, 116, 118, 119, 136, 137, 139, 140, 148, 155, 159, 202

Zhao, Wayne Xin; Chen, Rishan; Fan, Kai; Yan, Hongfei; y Li, Xiaoming. A novel burst-based text representation model for scalable event detection. En *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 2: Short Papers*, páginas 43–47, 2012. URL http://www.aclweb.org/anthology/P12-2009. 74